



VR-Link 3.4

DIS Class Documentation

This file documents VR-Link classes. It is based on the header files and includes many of the comments from them. It provides an alternative to looking through the individual header files when you need information about constructors or other aspects of a class's functioning. We hope you find this format helpful.

Classes are listed alphabetically and in a class hierarchy. The hypertext links will help you quickly jump to related information.

- [Table of Contents \(alphabetical listing\)](#)
- [Hierarchical List](#)

Notes

- There are two separate versions of VR-Link class documentation. One is for DIS and the other is for HLA.
- If you find a difference between the information in this file and the original header file shipped with your version of VR-Link, rely on the header file.
- The top line in each class description lists the header file in which it is defined. However, the capitalization of the header file is incorrect. The actual header files used mixed capitalization, not all lowercase.
- Some links point to header files. If you click one of these links, you will get a message indicating that the file could not be found. The file name will be included in the message, so if you want to, you can look at the file in the VR-Link Reference Manual or the delivered files.

MÄK Technologies, Inc.
185 Alewife Brook Parkway
Cambridge, MA 02138
Phone: 617.876.8085
Fax: 617.876.9208
www.mak.com
support@mak.com

Copyright 1999 MÄK Technologies, Inc. All Rights Reserved
Document ID: VRL-3.4-5D-990428

VR-Link 3.4 DIS Class Documentation

[Hierarchy of classes](#)

Table of Contents

- [Packages](#)
- [Classes](#)
- [Functions](#)
- [Variables](#)
- [Macros](#)
- [Enums, Unions, Structs](#)
- [Typedefs](#)

Packages

- [Services*](#) //////////////////////////////////

Classes

- [DtAbsNotifyObserver](#) class *DtAbsNotifyObserver*: *DtAbsNotifyObserver* is an abstract base class
- [DtAbsoluteVCPdu](#)
- [DtAckPredOpt](#) options for *DtAcknowledgePredicate*
- [DtAcknowledgePdu](#)
- [DtAcknowledgeRPdu](#) class *DtAcknowledgeRPdu*: Instances of *DtAcknowledgeRPdu* are used to send reliable acknowledge PDUs
- [DtActionRequestPdu](#)
- [DtActionRequestRPdu](#)
- [DtActionResponsePdu](#)
- [DtActionResponseRPdu](#)
- [DtAggregatePublisher](#) class *DtAggregatePublisher*: Instances of *DtAggregatePublisher* are used to take care of the details associated with publishing the state of an aggregate
- [DtAggregateStateDecoder](#) class *DtAggregateStateDecoder*: Instances of *DtAggregateStateDecoder* are used to take state information from an aggregate state PDU and place it into an aggregate state repository
- [DtAggregateStateEncoder](#) class *DtAggregateStateEncoder*: Instances of *DtAggregateStateEncoder* are used to take state information from an aggregate state repository and place it into an aggregate state PDU
- [DtAggregateStatePdu](#)
- [DtAggregateStateRepository](#) class *DtAggregateStateRepository*: Instances of *DtAggregateStateRepository* are used to maintain the state of an aggregate
- [DtAntennaPatternRecord](#) class *DtAntennaPatternRecord*: Instances of *DtAntennaPatternRecord* are
- [DtApproximator](#) class *DtApproximator*: *DtApproximator* is an abstract base class

- [DtArealObjectStatePdu](#)
- [DtArtPart](#) *One Articulated Part*
- [DtArtPartList](#) *class DtArtPartList: Instances of DtArtPartList are used to manage articulated parts*
- [DtArtPartThresholder](#) *class DtArtPartThresholder: DtArtPartThresholder is used to store art-part-related thresholds, and to perform some art part thresholding*
- [DtAttachedPartList](#) *class DtAttachedPartList: Instances of DtAttachedPartList are lists of attached parts*
- [DtBaseEntityStateRepository](#) *class DtBaseEntityStateRepository: Instances of DtBaseEntityStateRepository are used to maintain basic state information needed by all kinds of entities*
- [DtBaseExerciseConn](#)
- [DtBaseHashKey](#) *class DtBaseHashKey: DtBaseHashKey is an abstract base class for hashkeys*
- [DtBaseMsg](#)
- [DtBaseReflectedAggregate](#)
- [DtBaseReflectedEntity](#)
- [DtBasicFixedWingAircraftRecord](#) *class DtBasicFixedWingAircraftRecord: Instances of DtBasicFixedWingAircraftRecord are used to contain the entity state and logistics information about an individual entity within a group of entities*
- [DtBasicGroundCombatSoldierRecord](#) *class DtBasicGroundCombatSoldierRecord: Instances of DtBasicGroundCombatSoldierRecord are used to contain the entity state and logistics information about an individual entity within a group of entities*
- [DtBasicGroundCombatVehicleRecord](#) *class DtBasicGroundCombatVehicleRecord: Instances of DtBasicGroundCombatVehicleRecord are used to contain the entity state and logistics information about an individual entity within a group of entities*
- [DtBasicRotorWingAircraftRecord](#) *class DtBasicRotorWingAircraftRecord: Instances of DtBasicRotorWingAircraftRecord are used to contain the entity state and logistics information about an individual entity within a group of entities*
- [DtBeamAntennaPatternRecord](#) *class DtBeamAntennaPatternRecord: Instances of DtBeamAntennaPatternRecord are beam antenna pattern records used in conjunction with DtRadioTransmitterStateRepository (TSR)*
- [DtBurstDescriptor](#)
- [DtCartesianView](#) *class DtCartesianView: Instances of DtCartesianView are used to create a Cartesian coordinate view of the geocentric coordinates in an Entity State Repository*
- [DtClientSocket](#)
- [DtCollisionElasticPdu](#)
- [DtCollisionPdu](#)
- [DtCommentPdu](#)
- [DtCommentRPdu](#)
- [DtCompassVCPdu](#)
- [DtCoordTransform](#)
- [DtCoordTranslation](#)
- [DtCreateEntityPdu](#)
- [DtCreateEntityRPdu](#)
- [DtCsFilterTest](#)
- [DtDataPdu](#)
- [DtDataQueryPdu](#)

- [DtDataQueryRPdu](#)
- [DtDataRPdu](#)
- [DtDatumMap](#)
- [DtDatumMapEntry](#)
- [DtDatumRecordSet](#)
- [DtDcm](#)
- [DtDeadReckoner](#) class *DtDeadReckoner*: Instances of *DtDeadReckoner* are used to provide approximated position and orientation information for entities using various dead reckoning algorithms
- [DtDesignatorPdu](#)
- [DtDesignatorRepository](#) class *DtDesignatorRepository*: Instances of *DtDesignatorRepository* are
- [DtDetonationPdu](#)
- [DtDirectedRotRate](#) rotation rate given by single rate and axis of rotation
- [DtDisPduFactory](#) class *DtDisPduFactory*: Instances of *DtDisPduFactory* are used to create DIS PDUs
- [DtDmeGlobalFcns](#)
- [DtElectromagneticEmissionPdu](#)
- [DtEmbeddedSystemRepository](#) class *DtEmbeddedSystemRepository*: Instances of *DtEmbeddedSystemRepository* are
- [DtEmissionPdu](#)
- [DtEmitterBeamRepository](#) class *DtEmitterBeamRepository*: Instances of *DtEmitterBeamRepository* are
- [DtEmitterSystemRepository](#) class *DtEmitterSystemRepository*: Instances of *DtEmitterSystemRepository* are
- [DtEnhancedFixedWingAircraftRecord](#) class *DtEnhancedFixedWingAircraftRecord*: Instances of *DtEnhancedFixedWingAircraftRecord* are used to contain the entity state and logistics information about an individual entity within a group of entities
- [DtEnhancedGroundCombatSoldierRecord](#) class *DtEnhancedGroundCombatSoldierRecord*: Instances of *DtEnhancedGroundCombatSoldierRecord* are used to contain the entity state and logistics information about an individual entity within a group of entities
- [DtEnhancedGroundCombatVehicleRecord](#) class *DtEnhancedGroundCombatVehicleRecord*: Instances of *DtEnhancedGroundCombatVehicleRecord* are used to contain the entity state and logistics information about an individual entity within a group of entities
- [DtEnhancedRotorWingAircraftRecord](#) class *DtEnhancedRotorWingAircraftRecord*: Instances of *DtEnhancedRotorWingAircraftRecord* are used to contain the entity state and logistics information about an individual entity within a group of entities
- [DtEntityIdentifier](#)
- [DtEntityPublisher](#)
- [DtEntityStateDecoder](#) class *DtEntityStateDecoder*: Instances of *DtEntityStateDecoder* are objects capable of decoding *EntityState* PDUs into a *DtEntityStateRepository*
- [DtEntityStateEncoder](#)
- [DtEntityStatePdu](#)
- [DtEntityStateRepository](#) class *DtEntityStateRepository*: Instances of *DtEntityStateRepository* are used to maintain entity state
- [DtEntityStateUpdatePdu](#)
- [DtEntityType](#)
- [DtEnvironmentTypeRecord](#) class *DtEnvironmentTypeRecord*: Instances of *DtEnvironmentTypeRecord* are used

by the *DtGriddedDataPdu* for specifying environmental type

- [DtEnvironmentalProcessPdu](#)
- [DtEventId](#)
- [DtEventReportPdu](#)
- [DtEventReptRPdu](#)
- [DtExerciseConn](#)
- [DtFedReader](#) class *DtFedReader*: Instances of *DtFedReader* are used to read a FED file, and fill in a *DtFedTarget* with the information read
- [DtFedTarget](#) class *DtFedTarget*: *DtFedTarget* is an abstract class representing some representation of FED file data that can be filled in by a *DtFedReader*
- [DtFilename](#) class *DtFilename*: Instances of *DtFilename* are *DtStrings* which convert filenames to the native format
- [DtFilterTest](#)
- [DtFirePdu](#)
- [DtFundamentalParameterData](#)
- [DtFundamentalParameterData](#)
- [DtGeodeticCoord](#) class *DtGeodeticCoord*: Instances of *DtGeodeticCoord* are used to explicitly represent Geodetic coordinates
- [DtGlobalIdHashlist](#) class *DtGlobalIdHashlist*: Instances of *DtGlobalIdHashlist* are hashlists, where you can store associations between *DtGlobalObjectDesignators* and arbitrary pieces of data
- [DtGlobalObjectDesignatorList](#)
- [DtGridAxisRecordFixed](#) class *DtGridAxisRecordFixed*: *DtGridAxisRecordFixed* is the fixed grid axis record used by the gridded data PDU class (*DtGridDataPdu*)
- [DtGridAxisRecordVariable](#) class *DtGridAxisRecordVariable*: Instances of *DtGridAxisRecordVariable* are used to contain the information in the grid axis variable spacing records used by the gridded data PDU
- [DtGridDataRecord](#) class *DtGridDataRecord*: *DtGridDataRecord* is an abstract base class for the grid data record classes which are used by the gridded data PDU class (*DtGridDataPdu*)
- [DtGridDataRecordType0](#) class *DtGridDataRecordType0*: Instances of *DtGridDataRecordType0* are used to contain the information in the grid data records (of representation type 0) used by the gridded data PDU class
- [DtGridDataRecordType1](#) class *DtGridDataRecordType1*: Instances of *DtGridDataRecordType1* are used to contain the information in the grid data records (of representation type 1) used by the gridded data PDU class
- [DtGridDataRecordType2](#) class *DtGridDataRecordType2*: Instances of *DtGridDataRecordType2* are used to contain the information in the grid data records (of representation type 2) used by the gridded data PDU class
- [DtGriddedDataPdu](#)
- [DtGroundLogisticsVehicleRecord](#) class *DtGroundLogisticsVehicleRecord*: Instances of *DtGroundLogisticsVehicleRecord* are used to contain the entity state and logistics information about an individual entity within a group of entities
- [DtGroupSocket](#)
- [DtGroupVCPdu](#)
- [DtHashItem](#) class *DtHashItem*: Instances of *DtHashItem* are used internally to a hashlist to associate a pointer to an item in a list with a key
- [DtHashlist](#)
- [DtIffAtcNav aidsBeamData](#) class *DtIffAtcNav aidsBeamData*: *DtIffAtcNav aidsBeamData* is a class which contains the antenna data for the IFF/ATC/NAVAIDS PDU
- [DtIffAtcNav aidsLayerHeader](#) class *DtIffAtcNav aidsLayerHeader*: *DtIffAtcNav aidsLayerHeader* is a class which

contains the layer header information for a layer of the IFF/ATC/NAVAIDS PDU

- [DtIffAtcNav aidsOperational](#) class *DtIffAtcNav aidsOperational*: *DtIffAtcNav aidsOperational* is a class which contains the fundamental operational data for the IFF/ATC/NAVAIDS PDU
- [DtIffAtcNav aidsParameter](#) class *DtIffAtcNav aidsParameter*: *DtIffAtcNav aidsParameter* is a class which contains the parameter data for the IFF/ATC/NAVAIDS PDU
- [DtIffAtcNav aidsPdu](#)
- [DtIffAtcNav aidsSecondOperational](#) class *DtIffAtcNav aidsSecondOperational*: *DtIffAtcNav aidsSecondOperational* is a class which contains the secondary operational data for a layer of the IFF/ATC/NAVAIDS PDU
- [DtIffAtcNav aidsSystemID](#) class *DtIffAtcNav aidsSystemID*: *DtIffAtcNav aidsSystemID* is a class which contains information identifying the emitting system in the IFF/ATC/NAVAIDS PDU
- [DtIntHashKey](#) class *DtIntHashKey*: Instances of *DtIntHashKey* are used as a hash key made from an integer
- [DtInteraction](#) class *DtInteraction*: Instances of *DtInteraction* are used for representing DIS events
- [DtIntercomControlPdu](#)
- [DtIntercomEntityDestRec](#) class *DtIntercomEntityDestRec*: Instances of *DtIntercomEntityDestRec* are used to contain the information in the entity destination parameter records used by the intercom control PDU class
- [DtIntercomGroupAsgmtRec](#) class *DtIntercomGroupAsgmtRec*: Instances of *DtIntercomGroupAsgmtRec* are used to contain the information in the group assignment parameter record used by the intercom control PDU class
- [DtIntercomGroupDestRec](#) class *DtIntercomGroupDestRec*: Instances of *DtIntercomGroupDestRec* are used to contain the information in the group destination parameter record used by the intercom control PDU class
- [DtIntercomParmRec](#) class *DtIntercomParmRec*: *DtIntercomParmRec* is an abstract base class for the parameter record classes which are used by the intercom control PDU (*DtIntercomControlPdu*)
- [DtIntercomSignalPdu](#)
- [DtIntrusiveList](#) An intrusive list is a list that assumes that the elements that you add are pointers to instances of classes derived from *DtListedObject*
- [DtIsGroupOfPdu](#)
- [DtIsPartOfPdu](#)
- [DtLaserPdu](#)
- [DtLeAppearancePdu](#)
- [DtLeArticulatedPartsPdu](#)
- [DtLeDeadReckoningParameters](#) class *DtLeDeadReckoningParameters*: *DtLeDeadReckoningParameters* is a class which contains the entity dead reckoning parameter information for the live entity PDUs
- [DtLeDetonationPdu](#)
- [DtLeEntityID](#) class *DtLeEntityID*: *DtLeEntityID* is a class which contains the entity identification information for the live entity PDUs
- [DtLeEntityLinearVelocity](#) class *DtLeEntityLinearVelocity*: *DtLeEntityLinearVelocity* is a class which contains the entity linear velocity information for the live entity PDUs
- [DtLeEntityLocationEntityCoordinates](#) class *DtLeEntityLocationEntityCoordinates*: *DtLeEntityLocationEntityCoordinates* is a class which contains the entity location information (in entity coordinates) for the live entity PDUs
- [DtLeEntityLocationWorldCoordinates](#) class *DtLeEntityLocationWorldCoordinates*: *DtLeEntityLocationWorldCoordinates* is a class which contains the entity location information for the live entity PDUs
- [DtLeEntityOrientation](#) class *DtLeEntityOrientation*: *DtLeEntityOrientation* is a class which contains the entity orientation information for the live entity PDUs
- [DtLeFirePdu](#)

- [DtLeOrientationError](#) class *DtLeOrientationError*: *DtLeOrientationError* is a class which contains the entity orientation error information for the live entity PDUs
- [DtLePositionError](#) class *DtLePositionError*: *DtLePositionError* is a class which contains the entity location information for the live entity PDUs
- [DtLeTspiPdu](#)
- [DtLgrControlPdu](#) class *DtLgrControlPdu*: Instances of *DtLgrControlPdu* are a family of message used to control a remote MaK-Logger over the network
- [DtLgrEndPdu](#) class *DtLgrEndPdu*: Instances of *DtLgrEndPdu* are to tell remote logger to skip to end of file during playback
- [DtLgrEofActionPdu](#) class *DtLgrEofActionPdu*: Instances of *DtLgrEofActionPdu* are to tell remote logger that it should pause, wrap, or stop at the end of playback
- [DtLgrHeartbeatPdu](#) class *DtLgrHeartbeatPdu*: Instances of *DtLgrHeartbeatPdu* are to tell remote logger to send packet for each playback entity at least every HEARTBEAT seconds
- [DtLgrPausePdu](#) class *DtLgrPausePdu*: Instances of *DtLgrPausePdu* are to tell remote logger to pause
- [DtLgrPbFilePdu](#) class *DtLgrPbFilePdu*: Instances of *DtLgrPbFilePdu* are used for switching to play-mode and defining which file to read from when sending traffic into the simulation
- [DtLgrPlayActionPdu](#) class *DtLgrPlayActionPdu*: Instances of *DtLgrPlayActionPdu* are to tell remote logger what action to take initially when set to playback mode
- [DtLgrPlayPdu](#) class *DtLgrPlayPdu*: Instances of *DtLgrPlayPdu* are used to tell remote logger to begin recording
- [DtLgrQuitPdu](#) class *DtLgrQuitPdu*: Instances of *DtLgrQuitPdu* are messages telling a remote logger to quit
- [DtLgrRecFilePdu](#) class *DtLgrRecFilePdu*: Instances of *DtLgrRecFilePdu* are used to switching remote logger to recording-mode and to set the file into which to store captured simulation traffic
- [DtLgrRecordActionPdu](#) class *DtLgrRecordActionPdu*: Instances of *DtLgrRecordActionPdu* are to tell remote logger to stop or begin recording when entering record-mode
- [DtLgrRecordPdu](#) class *DtLgrRecordPdu*: Instances of *DtLgrRecordPdu* are to tell remote logger to begin recording
- [DtLgrSetTimePdu](#) class *DtLgrSetTimePdu*: Instances of *DtLgrSetTimePdu* are to tell remote logger to go to set position to HH:MM:SS
- [DtLgrSkipTimePdu](#) class *DtLgrSkipTimePdu*: Instances of *DtLgrSetTimePdu* are to tell remote logger to forward or backward HH:MM:SS
- [DtLgrStartPdu](#) class *DtLgrStartPdu*: Instances of *DtLgrStartPdu* are used to tell remote logger to start recording
- [DtLgrStopPdu](#) class *DtLgrStopPdu*: Instances of *DtLgrStopPdu* are to tell remote logger to stop
- [DtLgrTimeScalePdu](#) class *DtLgrTimeScalePdu*: Instances of *DtLgrTimeScalePdu* are to tell remote logger to slow down or speed up playback
- [DtLgrTimeoutPdu](#) class *DtLgrTimeoutPdu*: Instances of *DtLgrTimeoutPdu* are to tell remote logger how long to generate interim packets for entity before assuming no more packets in file
- [DtLinearObjectStatePdu](#)
- [DtLinearSegmentParameter](#) class *DtLinearSegmentParameter*: *DtLinearSegmentParameter* is a class which contains the linear segment parameters for the linear object state PDU
- [DtList](#) A *DtList* is a list of elements of arbitrary type
- [DtListItem](#) *DtListItem* is an envelope class used by *DtList*
- [DtListedObj](#) This is a base class from which elements of a *DtIntrusiveList* must be derived
- [DtMessagePdu](#)
- [DtMimicTrackVCPdu](#)
- [DtMimicVCPdu](#)

- [DtMinefieldDataPdu](#)
- [DtMinefieldPerimeterRecord](#) class *DtMinefieldPerimeterRecord*: *DtMinefieldPerimeterRecord* is a class which contains coordinates for the minefield perimeter points
- [DtMinefieldQueryPdu](#)
- [DtMinefieldResponseNackPdu](#)
- [DtMinefieldStatePdu](#)
- [DtModelessVCPdu](#)
- [DtModulationType](#)
- [DtNet32Vector](#)
- [DtNet64Vector](#)
- [DtNetBurstDescriptor](#)
- [DtNetEntityIdentifier](#)
- [DtNetEntityMarking](#)
- [DtNetEntityType](#)
- [DtNetEulerAngles](#)
- [DtNetEventId](#)
- [DtNetFloat32](#) Network-order 32-bit IEEE-754 floating-point quantity
- [DtNetFloat64](#) Network-order 64-bit IEEE-754 floating-point quantity
- [DtNetInt16](#) Network-order signed 16-bit quantity
- [DtNetInt32](#) Network-order signed 32-bit quantity
- [DtNetRadioEntityType](#)
- [DtNetSimulationAddress](#)
- [DtNetSocket](#)
- [DtNetTimeStamp](#) class *DtNetTimeStamp*: Instances of *DtNetTimeStamp* are network representations of a DIS time stamp
- [DtNetU16](#) Network-order unsigned 16-bit quantity
- [DtNetU32](#) Network-order unsigned 32-bit quantity
- [DtNullFederateAmbassador](#)
- [DtObjectIdHashKey](#) class *DtObjectIdHashKey*: Instances of *DtObjectIdHashKey* are hash keys made from *DtObjectIds*
- [DtObjectIdHashlist](#) class *DtObjectIdHashlist*: Instances of *DtObjectIdHashlist* are hashlists, where you can store associations between *DtObjectIds* and arbitrary pieces of data
- [DtObjectType](#) class *DtObjectType*: *DtObjectType* is a class which contains information identifying the type of a synthetic environment object
- [DtOrbitVCPdu](#)
- [DtPdu](#)
- [DtPduCallbackInfo](#) class *DtPduCallbackInfo*: Instances of *DtPduCallbackInfo* are used by *DtPduCallbackManager* to keep track of PDU callback functions
- [DtPduCallbackManager](#) class *DtPduCallbackManager*: Instances of *DtPduCallbackManager* are used to manage callbacks on the PDU kinds for the exercise connection
- [DtPduDatumRecordSet](#) An instance of this class is a *DtDatumRecordSet* that uses a part of a *DtPduWithData* as

its network representation

- [DtPduFactory](#)
- [DtPduWithData](#)
- [DtPointObjectStatePdu](#)
- [DtPropulsionSystem](#) class *DtPropulsionSystem*: *DtPropulsionSystem* is a class which contains the propulsion system data for the supplemental emissions/entity state PDU
- [DtRadioEntityType](#)
- [DtRadioReceiverRepository](#) class *DtRadioReceiverRepository*: Instances of *DtRadioReceiverRepository* are
- [DtRadioTransmitterRepository](#) class *DtRadioTransmitterRepository*: Instances of *DtRadioTransmitterRepository* are
- [DtReceiverPdu](#)
- [DtRecordPdu](#)
- [DtRecordQueryPdu](#)
- [DtRecordSet](#) class *DtRecordSet*: Instances of *DtRecordSet* are used to contain the record sets used by the PDU classes which inherit from the record set PDU base class
- [DtRecordSetPdu](#) class *DtRecordSetPdu*: *DtRecordSetPdu* is an abstract base class used by PDUs which require the record set functionality
- [DtReflectedAggregate](#) class *DtReflectedAggregate*: Instances of this class provide a protocol independent interface to a reflected aggregate in a DIS environment
- [DtReflectedAggregateList](#)
- [DtReflectedEmitterSystem](#) class *DtReflectedEmitterSystem*: Instances of *DtReflectedEmitterSystem* represent *EmitterSystem* object simulated by remote applications
- [DtReflectedEmitterSystemList](#)
- [DtReflectedEntity](#) class *DtReflectedEntity*: Instances of this class provide a protocol independent interface to a reflected entity in a DIS environment
- [DtReflectedEntityList](#)
- [DtReflectedObject](#) class *DtReflectedObject*: Instances of *DtReflectedObject* are used to represent a simulation entity reflected across a network (i
- [DtReflectedObjectCallbackInfo](#) class *DtReflectedObjectCallbackInfo*: Instances of *DtReflectedObjectCallbackInfo* are used to contain the function and user data associated with a *DtReflectedObject* callback,
- [DtReflectedObjectList](#) class *DtReflectedObjectList*: Instances of *DtReflectedObjectList* are list of instances of classes derived from *DtReflectedObject*
- [DtRemoveEntityPdu](#)
- [DtRemoveEntityRPdu](#)
- [DtRepairCompletePdu](#)
- [DtRepairResponsePdu](#)
- [DtResupplyCancelPdu](#)
- [DtResupplyOfferPdu](#)
- [DtResupplyReceivedPdu](#)
- [DtScript](#) A *DtScript* is a *DtList* of *StealthInstructions*
- [DtSelectParam](#)
- [DtServiceRequestPdu](#)

- [DtSetDataPdu](#)
- [DtSetDataRPdu](#)
- [DtSetRecordPdu](#)
- [DtSignalPdu](#)
- [DtSilentAggregateList](#) class *DtSilentAggregateList*: *DtSilentAggregateList* is a class which contains the information needed to describe a list of silent aggregates for the aggregate state PDU
- [DtSilentEntityList](#) class *DtSilentEntityList*: *DtSilentEntityList* is a class which contains the information needed to describe a list of silent entities for the aggregate state PDU
- [DtSimanPdu](#)
- [DtSmoothOriWithDr](#) class *DtSmoothOriWithDr*: Instances of *DtSmoothOriWithDr* are
- [DtSmoothOrient](#) ----- ORIENTATION -----
- [DtSmoothPos](#) ----- POSITION -----
- [DtSmoothPosWithDr](#) class *DtSmoothPosWithDr*: Instances of *DtSmoothPosWithDr* are
- [DtSmoother](#) class *DtSmoother*: Instances of *DtSmoother* are used to provide approximated position and orientation information for entities
- [DtSocket](#)
- [DtSocketPairSocket](#)
- [DtStartResumePdu](#)
- [DtStartResumeRPdu](#)
- [DtStateMsg](#) class *DtStateMsg*: Instances of *DtStateMsg* are used to represent state data communicated via the network in a DIS (as opposed to HLA) environment
- [DtStateRepository](#) class *DtStateRepository*: *DtStateRepository* is an abstract base class for maintaining state for an object
- [DtStopFreezePdu](#)
- [DtStopFreezeRPdu](#)
- [DtString](#) class *DtString*: Instances of *DtString* are
- [DtStringHashKey](#) class *DtStringHashKey*: Instances of *DtStringHashKey* are used as a hash key made from a string
- [DtSuppEmisEntStatePdu](#)
- [DtSystemHashKey](#) class *DtSystemHashKey*: Instances of *DtSystemHashKey* are used as hash keys for an emitter system
- [DtTaitBryan](#)
- [DtTcpForwarder](#)
- [DtTcpSocket](#)
- [DtTcpSocketMgr](#)
- [DtTetherTrackVCPdu](#)
- [DtTetherVCPdu](#)
- [DtThresholder](#) class *DtThresholder*: Instances of *DtThresholder* store thresholding information used to determine what attributes of an entity need to be updated
- [DtTopoView](#) class *DtTopoView*: Instances of *DtTopoView* are used to create a Topographic coordinate view of the geocentric coordinates in an Entity State Repository
- [DtTrackJamDesignator](#) class *DtTrackJamDesignator*: Instances of *DtTrackJamDesignator* are

- [DtTrackJamDesignatorList](#)
- [DtTrackJamTarget](#)
- [DtTrackJamTarget](#)
- [DtTrackVCPdu](#)
- [DtTransferControlRequestPdu](#)
- [DtTransmitterPdu](#)
- [DtUnderwaterAcousticAddtnlPassiveAct](#) class *DtUnderwaterAcousticAddtnlPassiveAct*:
DtUnderwaterAcousticAddtnlPassiveAct is a class which contains the additional passive activity information for the underwater acoustics PDU
- [DtUnderwaterAcousticBeam](#) class *DtUnderwaterAcousticBeam*: *DtUnderwaterAcousticBeam* is a class which contains the beam fundamental data parameters for the underwater acoustics PDU
- [DtUnderwaterAcousticEmitterSystem](#) class *DtUnderwaterAcousticEmitterSystem*:
DtUnderwaterAcousticEmitterSystem is a class which contains information describing the acoustic emitter systems in an underwater acoustics PDU
- [DtUnderwaterAcousticPdu](#)
- [DtUnderwaterAcousticShaftSpeed](#) class *DtUnderwaterAcousticShaftSpeed*: *DtUnderwaterAcousticShaftSpeed* is a class which contains the shaft speed information for the underwater acoustics PDU
- [DtUnknownPdu](#)
- [DtUnknownVCPdu](#)
- [DtUtmCoord](#) class *DtUtmCoord*: Instances of *DtUtmCoord* are used to represent UTM coordinates
- [DtUtmView](#) class *DtUtmView*: Instances of *DtUtmView* are used to create a UTM coordinate view of the geocentric coordinates in an Entity State Repository
- [DtVarDatumSpec](#) This class has public data members for backwards compatibility with pre-VR-Link3
- [DtVariableDatumRecord](#) class *DtVariableDatumRecord*: Instances of *DtVariableDatumRecord* are used to hold a single variable datum record and to provide access to its network representation
- [DtVector](#) class *DtVector*: No virtual functions, because we want the size of a *DtVector* to 3 * 8 bytes - so that you can make the same assumptions about it that we used to make about its size and layout
- [DtVectoringNozzleSystem](#) class *DtVectoringNozzleSystem*: *DtVectoringNozzleSystem* is a class which contains the vectoring nozzle system data for the supplemental emissions/entity state PDU
- [DtViewControlPdu](#)

Functions

- [\(*DtViewControlInteractionCb\)](#)
- [\(*DtAcknowledgeInteractionCb\)](#)
- [\(*DtAcknowledgeRPduCb\)](#) *DtAcknowledgeRPdu* callback function signature
- [\(*DtActionRequestInteractionCb\)](#)
- [\(*DtActionRequestRPduCb\)](#) *DtActionRequestRPdu* callback function signature
- [\(*DtActionResponseInteractionCb\)](#)
- [\(*DtActionResponseRPduCb\)](#) *DtActionResponseRPdu* callback function signature
- [\(*DtAggregateCallbackFunction\)](#)
- [\(*DtAggregateStatePduCb\)](#) *DtAggregateStatePdu* callback function signature
- [\(*DtArealObjectStatePduCb\)](#) *DtArealObjectStatePdu* callback function signature

- [\(*DtCollisionElasticPduCb\)](#) *DtCollisionElasticPdu callback function signature*
- [\(*DtCollisionInteractionCb\)](#)
- [\(*DtCommentPduCb\)](#)
- [\(*DtCommentRPduCb\)](#) *DtCommentRPdu callback function signature*
- [\(*DtCreateEntityPduCb\)](#)
- [\(*DtCreateEntityRPduCb\)](#) *DtCreateEntityRPdu callback function signature*
- [\(*DtDataPduCb\)](#)
- [\(*DtDataQueryPduCb\)](#)
- [\(*DtDataQueryRPduCb\)](#) *DtDataQueryRPdu callback function signature*
- [\(*DtDataRPduCb\)](#) *DtDataRPdu callback function signature*
- [\(*DtDesignatorPduCb\)](#)
- [\(*DtDetonationInteractionCb\)](#)
- [\(*DtElectromagneticEmissionPduCb\)](#)
- [\(*DtEmissionPduCb\)](#)
- [\(*DtEmitterSystemCallbackFunction\)](#)
- [\(*DtEntityCallbackFunction\)](#)
- [\(*DtEntityStatePduCb\)](#)
- [\(*DtEntityStateUpdatePduCb\)](#) *DtEntityStateUpdatePdu callback function signature*
- [\(*DtEnvironmentalProcessPduCb\)](#) *DtEnvironmentalProcessPdu callback function signature*
- [\(*DtEventReportPduCb\)](#)
- [\(*DtEventReptRPduCb\)](#) *DtEventReptRPdu callback function signature*
- [\(*DtFireInteractionCb\)](#)
- [\(*DtGlobalGetFcn\)](#)
- [\(*DtGlobalSetFcn\)](#)
- [\(*DtGriddedDataPduCb\)](#) *DtGriddedDataPdu callback function signature*
- [\(*DtIffAtcNav aidsPduCb\)](#) *DtIffAtcNav aidsPdu callback function signature*
- [\(*DtIntercomControlPduCb\)](#) *DtIntercomControlPdu callback function signature*
- [\(*DtIntercomSignalPduCb\)](#) *DtIntercomSignalPdu callback function signature*
- [\(*DtIsGroupOfPduCb\)](#) *DtIsGroupOfPdu callback function signature*
- [\(*DtIsPartOfPduCb\)](#) *DtIsPartOfPdu callback function signature*
- [\(*DtLaserPduCb\)](#)
- [\(*DtLeAppearancePduCb\)](#) *DtLeAppearancePdu callback function signature*
- [\(*DtLeArticulatedPartsPduCb\)](#) *DtLeArticulatedPartsPdu callback function signature*
- [\(*DtLeDetonationPduCb\)](#) *DtLeDetonationPdu callback function signature*
- [\(*DtLeFirePduCb\)](#) *DtLeFirePdu callback function signature*
- [\(*DtLeTspiPduCb\)](#) *DtLeTspiPdu callback function signature*
- [\(*DtLgrControlInteractionCb\)](#)
- [\(*DtLinearObjectStatePduCb\)](#) *DtLinearObjectStatePdu callback function signature*

- [\(*DtMessagePduCb\)](#)
- [\(*DtMinefieldDataPduCb\)](#) *DtMinefieldDataPdu callback function signature*
- [\(*DtMinefieldQueryPduCb\)](#) *DtMinefieldQueryPdu callback function signature*
- [\(*DtMinefieldResponseNackPduCb\)](#) *DtMinefieldResponseNackPdu callback function signature*
- [\(*DtMinefieldStatePduCb\)](#) *DtMinefieldStatePdu callback function signature*
- [\(*DtPduCallbackFcn\)](#) *PDU callback function signature*
- [\(*DtPduCreator\)](#) *Pdu creation function signature*
- [\(*DtPointObjectStatePduCb\)](#) *DtPointObjectStatePdu callback function signature*
- [\(*DtPostDrainCb\)](#) *class DtBaseExerciseConn: DtBaseExerciseConn is an abstract base class*
- [\(*DtPredicate\)](#)
- [\(*DtReceiverPduCb\)](#)
- [\(*DtRecordPduCb\)](#) *DtRecordPdu callback function signature*
- [\(*DtRecordQueryPduCb\)](#) *DtRecordQueryPdu callback function signature*
- [\(*DtReflectedObjectCallbackFunction\)](#) *Callback function that takes a DtReflectedObject and user data*
- [\(*DtRemoveEntityPduCb\)](#)
- [\(*DtRemoveEntityRPduCb\)](#) *DtRemoveEntityRPdu callback function signature*
- [\(*DtRepairCompletePduCb\)](#)
- [\(*DtRepairResponsePduCb\)](#)
- [\(*DtResupplyCancelPduCb\)](#)
- [\(*DtResupplyOfferPduCb\)](#)
- [\(*DtResupplyReceivedPduCb\)](#)
- [\(*DtServiceRequestPduCb\)](#)
- [\(*DtSetDataPduCb\)](#)
- [\(*DtSetDataRPduCb\)](#) *DtSetDataRPdu callback function signature*
- [\(*DtSetRecordPduCb\)](#) *DtSetRecordPdu callback function signature*
- [\(*DtSignalInteractionCb\)](#)
- [\(*DtStartResumePduCb\)](#)
- [\(*DtStartResumeRPduCb\)](#)
- [\(*DtStopFreezePduCb\)](#)
- [\(*DtStopFreezeRPduCb\)](#) *DtStopFreezeRPdu callback function signature*
- [\(*DtSuppEmisEntStatePduCb\)](#) *DtSuppEmisEntStatePdu callback function signature*
- [\(*DtTransferControlRequestPduCb\)](#) *DtTransferControlRequestPdu callback function signature*
- [\(*DtTransmitterPduCb\)](#)
- [\(*DtUnderwaterAcousticPduCb\)](#) *DtUnderwaterAcousticPdu callback function signature*
- [\(*DtUnknownPduCb\)](#)
- [APA](#)
- [D_E_Beamwidth](#)
- [DtAbort](#) *DtAbort() exits the program*

- [DtAbsNotifyObserver](#) *copy constructor*
- [DtAbsRealTime](#) *Return current wallclock time - seconds since Jan 1, 1970*
- [DtAbsTime2ClockTime](#)
- [DtAbsoluteVCPdu](#)
- [DtAckPredOpt](#)
- [DtAcknowledgePdu](#)
- [DtAcknowledgePredicate](#) *for use with Connection::DtReadUntil*
- [DtAcknowledgePredicate](#) *Return 1 if to, from, and ackFlag match between pkt and arg and positive response*
- [DtAcknowledgeRPdu](#)
- [DtActionRequestPdu](#)
- [DtActionRequestRPdu](#) *Constructors*
- [DtActionResponsePdu](#)
- [DtActionResponseRPdu](#) *Constructors*
- [DtAddDatumId2TypeMapping](#)
- [DtAddEnumString](#)
- [DtAggregatePublisher](#) *copy constructor*
- [DtAggregateStateDecoder](#) *default constructor*
- [DtAggregateStateEncoder](#) *default constructor*
- [DtAggregateStatePdu](#)
- [DtAggregateStateRepository](#) *default constructor*
- [DtAntennaPatternRecord](#) *copy constructor*
- [DtApproximator](#) *default constructor*
- [DtArealObjectStatePdu](#) *Constructors*
- [DtArtPart](#) *default constructor*
- [DtArtPartList](#) *copy constructor*
- [DtArtPartThresholder](#) *copy constructor*
- [DtAttachedPartList](#) *copy constructor*
- [DtBaseEntityStateRepository](#) *copy constructor*
- [DtBaseExerciseConn](#) *default constructor*
- [DtBaseHashKey](#) *copy constructor*
- [DtBaseMsg](#) *Constructor Object is not fully constructed until initMsg is called*
- [DtBaseReflectedAggregate](#) *constructor*
- [DtBaseReflectedEntity](#) *copy constructor*
- [DtBasicFixedWingAircraftRecord](#)
- [DtBasicGroundCombatSoldierRecord](#) *Constructors*
- [DtBasicGroundCombatVehicleRecord](#) *Constructors*
- [DtBasicRotorWingAircraftRecord](#)
- [DtBeamAntennaPatternRecord](#) *Initialize from DtAntennaPatternRecord*

- [DtBecomeNonBlocking](#)
- [DtBecomeNonBlocking](#)
- [DtBecomeNonBlocking](#)
- [DtBecomeNonBlocking](#)
- [DtBecomeNonBlocking](#) *designate the socket as non-blocking*
- [DtBodyToMat](#)
- [DtBodyToRef_to_Euler](#) *Given the transformation matrix that converts from a body frame to a reference frame (or visa-versa), produce the Euler angles between the body and the reference frames*
- [DtBurstDescriptor](#)
- [DtCartesianView](#) *copy constructor*
- [DtClientSocket](#)
- [DtClockTime2AbsTime](#)
- [DtClose](#)
- [DtCollisionElasticPdu](#)
- [DtCollisionPdu](#)
- [DtCommentPdu](#)
- [DtCommentRPdu](#) *Constructors*
- [DtCompassVCPdu](#)
- [DtCoordTransform](#)
- [DtCoordTranslation](#)
- [DtCreateCross](#)
- [DtCreateEntityPdu](#)
- [DtCreateEntityRPdu](#)
- [DtCsFilterTest](#)
- [DtCurrentMapDatum](#)
- [DtDataPdu](#)
- [DtDataQueryPdu](#)
- [DtDataQueryRPdu](#) *Constructors*
- [DtDataRPdu](#)
- [DtDatumId2Type](#)
- [DtDatumMap](#)
- [DtDatumMapEntry](#)
- [DtDatumRecordSet](#) *Constructor*
- [DtDcmAdd](#)
- [DtDcmAdjugate](#)
- [DtDcmCheck](#)
- [DtDcmCopy](#) *make orthonormal, pick most important*
- [DtDcmDcmMul](#) *result = m1 - m2*
- [DtDcmDcmRotAngle](#) *Direction Cosine Matrix routines*

- [DtDcmDcmRotAxis](#)
- [DtDcmDcmRotTrace](#) *Returns the trace of the rotation matrix, R, between between matrices A and B*
- [DtDcmDeterminant](#)
- [DtDcm](#)
- [DtDcmDump](#)
- [DtDcmFixMatrix](#) *Is this matrix orthonormal?*
- [DtDcmForm](#)
- [DtDcmFromQuaternion](#)
- [DtDcmIdentInit](#) *Initialize to all zeros*
- [DtDcmInit](#)
- [DtDcmInverse](#)
- [DtDcmLevelInit](#)
- [DtDcmRotInit](#) *Rotation is counterclockwise when viewed a positive axis*
- [DtDcmRotInit2](#)
- [DtDcmScale](#)
- [DtDcmSub](#) *result = m1 + m2*
- [DtDcmToElr](#)
- [DtDcmTranspose](#) *result = m1 x m2*
- [DtDcmTrigInit](#)
- *\$ only pos rots?*
- [DtDcmVecMul](#) *Combined vector/matrix routines*
- [DtDeadReckoner](#) *copy constructor*
- [DtDeg2DmsString](#)
- [DtDesignatorPdu](#)
- [DtDesignatorRepository](#) *copy constructor*
- [DtDetonationPdu](#)
- [DtDirectRotRate](#) *directed rotation rates*
- [DtDirectedRotRate](#)
- [DtDisPduFactory](#) *default constructor Adds creators for the various DIS pdus*
- [DtDistSqr](#) $||p1-p2||_{2}^2$
- [DtDistance](#) *additional Routines*
- [DtDmeGlobalFcns](#)
- [DtDumpBoolean](#)
- [DtDumpEnum](#)
- [DtDumpFloat](#)
- [DtDumpHexBytes](#)
- [DtDumpHexWord](#)
- [DtDumpInteger](#)
- [DtDumpString](#)

- [DtEXTEND_ATTR_ENUM](#)
- [DtEXTEND_ATTR_VAL_ENUM](#)
- [DtEXTEND_BIT_SHIFT_ENUM](#)
- [DtElectromagneticEmissionPdu](#)
- [DtElrCopy](#)
- [DtElrElrCat](#)
- [DtElrForm](#)
- [DtElrIdent](#)
- [DtElrToMat](#)
- [DtElrTranspose](#)
- [DtEmbeddedSystemRepository](#) *default constructor*
- [DtEmissionPdu](#)
- [DtEmitterBeamRepository](#) *copy constructor*
- [DtEmitterSystemRepository](#) *default constructor*
- [DtEnhancedFixedWingAircraftRecord](#) *Constructors*
- [DtEnhancedGroundCombatSoldierRecord](#)
- [DtEnhancedGroundCombatVehicleRecord](#)
- [DtEnhancedRotorWingAircraftRecord](#) *Constructors*
- [DtEntityIdentifier](#)
- [DtEntityPublisher](#)
- [DtEntityStateDecoder](#) *copy constructor*
- [DtEntityStateEncoder](#) *Constructor*
- [DtEntityStatePdu](#)
- [DtEntityStateRepository](#) *copy constructor*
- [DtEntityStateUpdatePdu](#)
- [DtEntityType](#)
- [DtEnuToNedDcm](#)
- [DtEnumString](#)
- [DtEnumString](#)
- [DtEnumString](#)
- [DtEnvironmentTypeRecord](#) *default constructor*
- [DtEnvironmentalProcessPdu](#) *Constructors*
- [DtEulerToEuler](#) *Given Euler angles with respect to reference frame "in", and the transformation matrix between frame "in" and frame öut", produce the Euler angles with repect to reference frame öut"*
- [DtEuler_to_BodyToRef](#) *Given Euler angles between a body and a reference frame, produce the transformation matrix that converts from the body frame to the reference frame (or visa-versa)*
- [DtEuler_to_RefToBody](#)
- [DtEventId](#)
- [DtEventReportPdu](#)

- [DtEventReptRPdu](#)
- [DtExerciseConn](#) ----- INLINES -----
- [DtFatalError](#) *These functions print messages using stdio depending on the notify level*
- [DtFatalNetPerror](#)
- [DtFatalPerror](#)
- [DtFedReader](#) *copy constructor*
- [DtFedTarget](#) *Constructor*
- [DtFilename](#) *constructor*
- [DtFilterTest](#)
- [DtFirePdu](#)
- [DtFirstHostInetAddr](#)
- **DtGeocToGeod :**
 - in [duplicate](#) *Conversions between geocentric and geodetic: Use of these functions is discouraged in favor of DtGeodeticCoord member functions which perform conversions*
 - in [DtGeodeticCoord](#)
- [DtGeocToTopoTransform](#) *This function can initialize a DtCoordTransform so that it can provide transformations between the geocentric coordinate system, and a particular topographic coordinate system defined by latitude and longitude*
- **DtGeocToUtm :**
 - in [duplicate](#) *Use of these functions is discouraged in favor of DtUtmCoord member functions which perform conversions*
 - in [DtUtmCoord](#)
- [DtGeodToGeoc](#)
- **DtGeodToUtm :**
 - in [duplicate](#) *Use of these functions is discouraged in favor of DtUtmCoord member functions which perform conversions*
 - in [DtUtmCoord](#)
- [DtGeodeticCoord](#) *constructor*
- [DtGetApproximateElapsedRealTime](#) *Returns the value of DtGetElapsedRealTime at the time that DtSetSimTime was last invoked*
- [DtGetBroadcastMask](#)
- [DtGetElapsedRealTime](#) *Return real seconds elapsed since DtTimeInit was called*
- [DtGetSimTime](#) *Returns the last value passed to DtSetSimTime - the current value of VR-Link simulation time*
- [DtGlobalIdHashlist](#) *default constructor*
- [DtGlobalObjectDesignatorList](#) *default constructor*
- [DtGridAxisRecordFixed](#) *Constructor for derived classes*
- [DtGridAxisRecordVariable](#) *Constructors*
- [DtGridDataRecord](#) *Constructors (protected because this is an abstract base class)*
- [DtGridDataRecordType0](#)
- [DtGridDataRecordType1](#)
- [DtGridDataRecordType2](#)

- [DtGridDeclinationDcm](#)
- [DtGriddedDataPdu](#)
- [DtGroundLogisticsVehicleRecord](#) Constructors
- [DtGroupSocket](#) Constructor - uses *addr* for listening
- [DtGroupVCPdu](#)
- [DtGtPacketDequeue](#)
- [DtGtPacketEnqueue](#)
- [DtGtRecv](#)
- [DtGtSend](#) No, but Ethernet does, and that's the only thing the GT100 will ever work with
- [DtHashItem](#) default constructor
- [DtHashlist](#) Copy constructor - always uses an internal list, so if you want your copy to use an external list, do not use the copy constructor
- [DtIffAtcNavAidsBeamData](#)
- [DtIffAtcNavAidsLayerHeader](#)
- [DtIffAtcNavAidsOperational](#) Constructors
- [DtIffAtcNavAidsParameter](#) Constructors
- [DtIffAtcNavAidsPdu](#) Constructors
- [DtIffAtcNavAidsSecondOperational](#) Constructors
- [DtIffAtcNavAidsSystemID](#)
- [DtInetAddrOfDevice](#)
- [DtInetAddrString](#) converts *ulong* to internet dot notation (returns internal static buffer)
- [DtInetBroadcastOfDevice](#)
- [DtInfo](#)
- [DtInitNetSocket](#)
- [DtIntHashKey](#) default constructor
- [DtInteraction](#) copy constructor
- [DtIntercomControlPdu](#)
- [DtIntercomEntityDestRec](#) Constructors
- [DtIntercomGroupAsgmtRec](#) Constructors
- [DtIntercomGroupDestRec](#) Constructors
- [DtIntercomParmRec](#)
- [DtIntercomSignalPdu](#)
- [DtIsGroupOfPdu](#)
- [DtIsInList](#) Returns -1 if *num* is not in *lst*
- [DtIsMulticastAddr](#)
- [DtIsPartOfPdu](#)
- [DtIsSimTimeSet](#) Abort if *DtSetSimTime* was never called
- [DtLaserPdu](#)
- [DtLatLon_to_GeocToTopo](#) The matrix produced by *DtLatLon_to_GeocToTopo* can be used to rotate a geocentric

vector to a topographic vector like this: `DtDcmVecMul(geocToTopo, Vgeoc, Vtopo);`

- [DtLatLon to TopoToGeoc](#) From latitude and longitude, we can produce the rotation matrices that rotate between geocentric and topographic-down orientations
- [DtLeAppearancePdu](#)
- [DtLeArticulatedPartsPdu](#)
- [DtLeDeadReckoningParameters](#)
- [DtLeDetonationPdu](#)
- [DtLeEntityID](#) Constructors
- [DtLeEntityLinearVelocity](#) Constructors
- [DtLeEntityLocationEntityCoordinates](#) Constructors
- [DtLeEntityLocationWorldCoordinates](#) Constructors
- [DtLeEntityOrientation](#) Constructors
- [DtLeFirePdu](#) Constructors
- [DtLeOrientationError](#)
- [DtLePositionError](#) Constructors
- [DtLeTspiPdu](#)
- [DtLgrControlPdu](#) default constructor
- [DtLgrEndPdu](#) construct and prepare for send
- [DtLgrEofActionPdu](#) default constructor
- [DtLgrHeartbeatPdu](#) construct from network representation
- [DtLgrPausePdu](#) construct from network representation
- [DtLgrPbFilePdu](#) copy constructor
- [DtLgrPlayActionPdu](#) construct and prepare for send
- [DtLgrPlayPdu](#) default constructor
- [DtLgrQuitPdu](#) default constructor
- [DtLgrRecFilePdu](#) construct and prepare for send
- [DtLgrRecordActionPdu](#) copy constructor
- [DtLgrRecordPdu](#) construct and prepare for send
- [DtLgrSetTimePdu](#) default constructor
- [DtLgrSkipTimePdu](#) default constructor
- [DtLgrStartPdu](#) construct and prepare for send
- [DtLgrStopPdu](#) default constructor
- [DtLgrTimeScalePdu](#) copy constructor
- [DtLgrTimeoutPdu](#) construct from network representation
- [DtLimit](#)
- [DtLinearObjectStatePdu](#)
- [DtLinearSegmentParameter](#)
- [DtList](#) Copy constructor
- [DtListItem](#) Constructor - called only by `DtList`

- [DtListedObj](#) *Constructor*
- [DtM](#)
- [DtMM_Mult](#)
- [DtM_Init](#)
- [DtMakeRotationDcm](#)
- [DtMax](#)
- [DtMessagePdu](#)
- [DtMimicTrackVCPdu](#)
- [DtMimicVCPdu](#)
- [DtMin](#)
- [DtMinefieldDataPdu](#)
- [DtMinefieldPerimeterRecord](#)
- [DtMinefieldQueryPdu](#)
- [DtMinefieldResponseNackPdu](#) *Constructors*
- [DtMinefieldStatePdu](#)
- [DtModPer](#) *For x with period ABS(incl-excl), return DtModPerLo(x, incl, excl) if incl < excl return DtModPerHi(x, excl, incl) if incl > excl*
- [DtModPerHi](#) *For x with period (max-min), return min < x <= max*
- [DtModPerLo](#) *For x with period (max-min), return min <= x < max*
- [DtModPerZero](#) *For x with period per, return 0 <= x < per*
- [DtModelessVCPdu](#)
- [DtModulationType](#)
- [DtNedToEnuDcm](#)
- [DtNet32Vector](#)
- [DtNet64Vector](#)
- [DtNetEulerAngles](#)
- [DtNetFloat32](#)
- [DtNetFloat64](#)
- [DtNetInt16](#)
- [DtNetInt32](#) *Least Significant Word*
- [DtNetMaskOfDevice](#)
- [DtNetSocket](#) *No default for flags since 3rd arg is needed to disambiguate*
- [DtNetTimeStamp](#) *copy constructor*
- [DtNetU16](#) *Least Significant Byte*
- [DtNetU32](#)
- [DtNewDfltSocket](#)
- [DtNewViewControlPdu](#)
- [DtNullFederateAmbassador](#)
- [DtObjectIdHashKey](#) *copy constructor*

- [DtObjectIdHashlist](#) *default constructor*
- [DtObjectType](#)
- [DtOrbitVCPdu](#)
- [DtPduCallbackInfo](#) *copy constructor*
- [DtPduCallbackManager](#) *default constructor*
- [DtPduDatumRecordSet](#) *Constructor*
- [DtPdu](#) *Copy constructor*
- [DtPduFactory](#) *copy constructor*
- [DtPduWithData](#) *Constructor*
- [DtPointObjectStatePdu](#) *Constructors*
- [DtPollInputLine](#) *polls line-buffered standard input returns NULL if no input returns internal buffer of line if have input exits on ^D*
- [DtPropulsionSystem](#) *Constructors*
- [DtQuatQuatMul](#)
- [DtQuaternionDump](#) *void DtQuatScale(DtQuaternion *qresult, DtQuaternion *q, DtReal delta_t);*
- [DtQuaternionFromTaitBryan](#)
- [DtROUND_TO_MULT](#) *Rounds num up to a multiple of mult*
- [DtRadioEntityType](#)
- [DtRadioReceiverRepository](#) *default constructor*
- [DtRadioTransmitterRepository](#) *copy constructor*
- [DtReal](#)
- [DtReceiverPdu](#)
- [DtRecordPdu](#) *Constructors*
- [DtRecordQueryPdu](#) *Constructors*
- [DtRecordSet](#)
- [DtRecordSetPdu](#)
- [DtRecv](#) *this is slower as it does a memcpy()*
- [DtRecv](#)
- [DtRecv](#)
- [DtRecv](#)
- [DtRecv](#)
- [DtRefToBody_to_Euler](#)
- [DtReflectedAggregate](#) *copy constructor*
- [DtReflectedAggregateList](#) *Constructor*
- [DtReflectedEmitterSystem](#) *Constructor*
- [DtReflectedEmitterSystemList](#) *Constructor*
- [DtReflectedEntity](#) *Constructor*
- [DtReflectedEntityList](#) *Copy constructor is protected because it is not fully implemented*
- [DtReflectedObjectCallbackInfo](#) *copy constructor*

- [DtReflectedObject](#) *constructor*
- [DtReflectedObjectList](#) *copy constructor*
- [DtRemoveEntityPdu](#)
- [DtRemoveEntityRPdu](#)
- [DtRepairCompletePdu](#)
- [DtRepairResponsePdu](#)
- [DtReportPktsRcvd](#)
- [DtReportPktsRcvd](#)
- [DtResupplyCancelPdu](#)
- [DtResupplyOfferPdu](#)
- [DtResupplyReceivedPdu](#)
- [DtRotate](#)
- [DtScript](#)
- [DtSelect](#)
- [DtSelectParam](#)
- [DtSend](#) *Virtual function override - send to all remote sites*
- [DtSend](#)
- [DtServiceRequestPdu](#)
- [DtSetDataPdu](#)
- [DtSetDataRPdu](#)
- [DtSetRecordPdu](#) *Constructors*
- [DtSetSimTime](#) *Set the current value of VR-Link simulation time*
- [DtSetSym](#)
- [DtSignalPdu](#)
- [DtSilentAggregateList](#)
- [DtSilentEntityList](#) *Constructors*
- [DtSleep](#)
- [DtSmoothOriWithDr](#) *copy constructor*
- [DtSmoothOrient](#) *Constructors*
- [DtSmoothPos](#)
- [DtSmoothPosWithDr](#) *copy constructor*
- [DtSmoother](#) *copy constructor*
- [DtSocket](#)
- [DtSocketPairSocket](#)
- [DtStartResumePdu](#)
- [DtStartResumeRPdu](#) *Constructors*
- [DtStateMsg](#) *default constructor*
- [DtStateRepository](#) *default constructor*

- [DtStopFreezePdu](#)
- [DtStopFreezeRPdu](#)
- [DtStrdup](#) *Strdup wrapper*
- [DtString](#)
- [DtStringHashKey](#) *copy constructor*
- [DtStringToInetAddr](#)
- [DtSum](#)
- [DtSuppEmisEntStatePdu](#)
- [DtSystemHashKey](#) *copy constructor*
- [DtTaitBryan](#)
- [DtTbToMat](#)
- [DtTcpForwarder](#)
- [DtTcpSocket](#) *Client Constructor*
- [DtTcpSocketMgr](#)
- [DtTetherTrackVCPdu](#)
- [DtTetherVCPdu](#)
- [DtThresholder](#) *copy constructor*
- [DtTimeInit](#) *Initialize all VR-Link time functionality This must be called before any other time-related functions*
- [DtTimeNow](#) *Obsolete*
- [DtToLower](#) *Convert str to all lowercase*
- [DtToUpper](#) *Convert str to all uppercase*
- [DtTopoView](#) *constructor*
- [DtTrackJamDesignator](#) *copy constructor*
- [DtTrackJamDesignatorList](#) *copy constructor*
- [DtTrackVCPdu](#)
- [DtTransferControlRequestPdu](#) *Constructors*
- [DtTransmitterPdu](#)
- [DtTypeMatchPattern](#)
- [DtUnderwaterAcousticAddtnlPassiveAct](#)
- [DtUnderwaterAcousticBeam](#)
- [DtUnderwaterAcousticEmitterSystem](#) *Constructors*
- [DtUnderwaterAcousticPdu](#)
- [DtUnderwaterAcousticShaftSpeed](#) *Constructors*
- [DtUnknownPdu](#)
- [DtUnknownVCPdu](#)
- [DtUseMapDatum](#) *Set and Get current map datum NOTE: map datum memory passed to DtUseMapDatum() must remain valid until next change of datum*
- [DtUtmCoord](#) *default constructor*
- [DtUtmGridDeclination](#)

- [DtUtmInit](#) Sets reference point for all DtUtmCoords Vrlink presently supports only one UTM reference at a time, but the reference point may be reinitialized at any point
- [DtUtmToGeoc](#)
- [DtUtmToGeod](#)
- [DtUtmView](#) constructor
- [DtVV_Product](#)
- [DtVariableDatumRecord](#) copy constructor
- [DtVecAdd](#) result = -vec
- [DtVecBoxProd](#) sweep v1 to v2
- [DtVecCheck](#)
- [DtVecCopy](#) Initialize to all zeros
- [DtVecCosProd](#) result = v1 - v2
- [DtVecCrossProd](#)
- [DtVecDcmMul](#)
- [DtVecDotProd](#) cos of angle between vecs
- [DtVecDump](#)
- [DtVecElrTransform](#)
- [DtVecInfNorm](#)
- [DtVecInit](#)
- [DtVecNeg](#)
- [DtVecNormalize](#)
- [DtVecScale](#) Is this vector normalized?
- [DtVecSub](#)
- [DtVector](#)
- [DtVectoringNozzleSystem](#) Constructors
- [DtVerbose](#)
- [DtWarn](#)
- [DtWarnNetPerror](#)
- [DtWarnPerror](#)
- [EM_AppearanceOffset](#)
- [ERP](#)
- [ERP](#)
- [IR_AppearanceOffset](#)
- [PDU_SequenceNumber](#)
- [PRF](#)
- [PgRF](#)
- [PrivateListenToMulticastGroup](#)
- [TDL_Type](#)
- [X_Offset](#)

- [X_Offset](#)
- [X_Offset](#)
- [X_Offset](#)
- [X_Offset](#)
- [Y_Offset](#)
- [Y_Offset](#)
- [Y_Offset](#)
- [Y_Offset](#)
- [Y_Offset](#)
- [Z_Offset](#)
- [Z_Offset](#)
- [Z_Offset](#)
- [Z_Offset](#)
- [Z_Offset](#)
- [absolute](#) *Returns DtTrue if the filename is an absolute filename, or a relative filename, respectively*
- [acceleration](#)
- [acceleration](#)
- [acceleration](#)
- [acceleration](#)
- [acceleration](#)
- [acceleration](#)
- [acceleration](#)
- [accept](#)
- [acceptPdu](#) *Determines if a PDU should be processed or not*
- [acknowledgeFlag](#)
- [acknowledgeFlag](#)
- [acousticID](#)
- [acousticName](#)
- [acousticSigRepIndex](#)
- [action](#) *Set/Get value for what to do when Logger reaches EOF*
- [action](#)
- [action](#)
- [actionId](#)
- [actionId](#)
- [activeEmissionParmIndex](#)
- [addAfter](#) *Add element to the list just after the indicated item*
- [addAggregateAdditionCallback](#) *Aggregate added callback registration/deregistration*
- [addAggregateRemovalCallback](#) *Aggregate removed callback registration/deregistration*

- [addCallback](#) *Add (register)/Remove (deregister) function to be called when pdu is received*
 - [addCallback](#) *Add (register)/Remove (deregister) function to be called when pdu is received*
 - [addCallback](#) *Add (register)/Remove (deregister) function to be called when pdu is received*
 - [addCallback](#) *Add (register)/Remove (deregister) function to be called when pdu is received*
 - [addCallback](#) *Add (register)/Remove (deregister) function to be called when pdu is received*
 - [addCallback](#) *DIS -- install/remove handler to call when receiving a LC packet*
 - [addCallback](#) *Add (register)/Remove (deregister) function to be called when pdu is received*
 - [addCallback](#)
 - [addCallback](#) *Add (register)/Remove (deregister) function to be called when pdu is received*
 - [addCallback](#) *Add (register)/Remove (deregister) function to be called when pdu is received*
-

- [addCallback](#)
- [addCreator](#) *Add a pdu creation function to the factory*
- [addDis203Creators](#) *Adds creators for all DIS 2*
- [addDis204Creators](#) *Adds creators for all DIS 2*
- [addDis214Creators](#) *Adds creators for all DIS 2*
- [add](#) *Add an association between key and data to the list*
- [add](#) *Add something to the list*
- [add](#) *Add element to the end of the list*
- [add](#) *Add an association between key and data to the list*
- [addEmitterAdditionCallback](#) *Emitter added callback registration/deregistration*
- [addEmitterRemovalCallback](#) *Emitter removed callback registration/deregistration*
- [addEntityAdditionCallback](#) *Entity added callback registration/deregistration*
- [addEntityRemovalCallback](#) *Entity removed callback registration/deregistration*
- [addInstruction](#)
- [addInterClass](#) *Define a class*
- [addInterestInMcastAddr](#) *Add/Subtract interest in a multicast address*
- [addInterestInPduKind](#) *Adds/subtracts interest in a PDU kind*
- [addObjClass](#) *Define a class*
- [addObjectAdditionCallback](#) *Object added callback registration/deregistration*
- [addObjectRemovalCallback](#) *Object removed callback registration/deregistration*
- [addParam](#) *Add the specified parameter type to the part*
- [addParameter](#) *Define a parameter of a class*
- [addPduCallback](#) *Register/deregister callbacks by PDU kind*
- [addPduCallback](#) *Add/Remove a function to be called when a particular PDU kind is received*
- [addPostDrainCallback](#) *add/remove callbacks to be executed from within drainInput right before it returns*
- [addPostUpdateCallback](#) *Entity state callback registration/deregistration*
- [addPostUpdateCallback](#) *emitterSystem state callback registration/deregistration*
- [addPostUpdateCallback](#) *Entity state callback registration/deregistration*
- [addRead](#)
- [addRemoteSite](#) *Add/remove a remote site to send to*
- [addToEnd](#) *Add element to the end of the list*
- **addToList :**
 - in [DtIntrusiveList](#) *Override of virtual function*
 - in [DtList](#) *Add item to the list after after*
- [addToStart](#) *Add element to the beginning of the list*
- [addressInPacket](#)
- [addressInPacket](#)
- [afterburnerOn](#)

- [illegible]

- [appearanceAttribute](#)
- [appearanceAttribute](#)
- [appearanceAttribute](#)
- [appearanceAttribute](#)
- [appearanceAttribute](#)
- [appearanceAttribute](#)
- [appearance](#)
- [appearance](#)
- [appearance](#)
- [appearance](#)
- [appearance](#)
- [appearance](#)
- [appearance](#)
- [appearance](#)
- [appearance](#)
- [appearance](#)
- [appearanceField1](#) *Get the first appearance flag field (this field is automatically maintained by the PDU as other PDU fields are set)*
- [appearanceField1Flag](#) *Get the value for a specific appearance flag from the first appearance field*
- [appearanceField1Offset](#) *Returns the offset (in bytes) of a field in the variable data region*
- [appearanceField2Offset](#)
- [applicableModes](#)
- [application](#)
- [applicationId](#)
- [approxBodyToGeoc](#) *Get the approximated data body to geocentric coordinate matrix*
- [approxBodyToGeoc](#)
- [approxBodyToGeoc](#) *Get the approximated data body to geocentric coordinate matrix*
- [approxEuler](#) *Get approximated orientation (euler angles) at the specified time*
- [approxEuler](#) *Get the approximated (i*
- [approxEuler](#) *Get approximated orientation (euler angles) at the specified time*
- [approxLocation](#) *Get the approximated location at the specified time*
- [approxLocation](#)
- [approxLocation](#) *Get the approximated location at the specified time*
- [approxVal](#) *Get the approximated value*
- [approxVelocity](#) *Get the approximated velocity at the specified time*
- [approxVelocity](#)
- [approxVelocity](#) *Get the approximated velocity at the specified time*

- [approximating](#)
- [approximator](#)
- [artNotAttached](#) *Returns true if param is an articulated parameter record, false if its an attached part record*
- [artParamIsRate](#) *Get whether the parameter type is actually a rate*
- [artParmOffset](#) *Returns the offset (in bytes) of a field in the variable data region*
- [artParmOffset](#)
- [artPartFromPartNum](#) *Get the articulated part from the part number*
- [artPartFromPartType](#) *Get the first articulated part with the part type*
- [artPartList](#) *returns a DtArtPartList containing the parts described in this PDU*
- [artPartList](#) *get the articulated parts list*
- [artPartRecord](#)
- [artPartRecord](#)
- [artPartSpecs](#) *return the list of DtArtPartSpecs describing the parts in the list*
- [artPartSpecs](#)
- [artPartSpecs](#)
- [artPartThreshold](#) *Get the thresholding value (functional, translational, or rotational) for the parameter type*
- [artPartThresholder](#)
- [artPartsExceedAggThreshold](#) *Get whether or not an aggregate threshold (such as position, translation, rotation, etc*
- [artPartsExceedIndivThreshold](#) *Get whether or not an individual parameter threshold has been exceeded*
- [artPartsNeedUpdating](#) *Returns true if art part info is out of date, false otherwise*
- [asr](#) *Synonym for aggregateStateRep()*
- [attParmOffset](#)
- [attParmOffset](#)
- [attPartList](#)
- [attPartsNeedUpdating](#) *Returns true if art part info is out of date, false otherwise*
- **attachId :**
 - in [DtCompassVCPdu](#)
 - in [DtMimicTrackVCPdu](#)
 - in [DtMimicVCPdu](#)
 - in [DtOrbitVCPdu](#)
 - in [DtTetherTrackVCPdu](#)
 - in [DtTetherVCPdu](#)
- [attachedPartChange](#)
- [attachedPartRecord](#)
- [attachedPartStations](#)
- [attackerId](#)
- [attackerId](#)
- [attributelsNotOwned](#)

- [attributeOwnedByRTI](#)
- [attributeOwnershipAcquisitionNotification](#)
- [attributeOwnershipDivestitureNotification](#)
- [attributeOwnershipUnavailable](#)
- [attributesInScope](#)
- [attributesOutOfScope](#)
- [audioAppearanceOffset](#)
- [axisType](#) *Get the grid axis record type from a network structure*
- [azimuthBeamwidth](#) *Get the azimuth beam bandwidth*
- [azimuthCenter](#)
- [azimuthCenter](#)
- [azimuthError](#)
- [azimuthRate](#)
- [azimuthSweep](#)
- [azimuthSweep](#)
- [azimuthalBeamwidth](#)
- [bandwidth](#)
- [baseAddPostUpdateCallback](#) *Object state callback registration/deregistration*
- [baseLoadNetRecord](#)
- [baseLoadNetRecord](#)
- [baseRemovePostUpdateCallback](#)
- [baseUnloadNetRecord](#) *Load/unload data from a packet record*
- [baseUnloadNetRecord](#) *Load/unload data from a packet record*
- [beamCenterAzimuth](#)
- [beamCenterD_E](#)
- [beamData](#)
- [beamDataLength](#)
- [beamDataRecordOffset](#) *Returns a buffer offset to the given beam fundamental data parameter of a particular emitter system*
- [beamDataWords](#)
- [beamDataWords](#)
- [beamFunction](#)
- [beamFunction](#)
- [beamFunction](#)
- [beamID_Number](#)
- [beamId](#)
- [beamId](#)
- [beamId](#)
- [beamId](#)

- [beamList](#) *Returns DtList of DtEmitterBeamStateRepositories*
- [beamParameterIndex](#)
- [beamParameterIndex](#)
- [beamParameterIndex](#)
- [beamSRById](#) *Return a pointer to the first BeamSR with specified beamID*
- [beamType](#)
- [bodyHeading](#)
- [bodyOrientation](#)
- [bodyToGeoc](#) *Get the body to geocentric coordinate matrix*
- [bodyToLocal](#) *Get the body to local coordinate matrix*
- [bodyToLocal](#) *Get the body to local coordinate matrix*
- [bucketFromHashKey](#) *Determine the bucket from the hashkey*
- [bucketItemLookup](#) *Lookup an item in a bucket*
- [burst](#) *NOT USED IN HLA*
- [burst](#) *NOT USED IN HLA*
- [burstLength](#)
- [burstMunitionType](#)
- [burstMunitionType](#)
- [burstOffset1](#)
- [burstOffset1](#)
- [burstOffset2](#)
- [burstOffset2](#)
- [burstOffset3](#)
- [burstOffset3](#)
- [byteLengthOfNetworkRecord](#) *Get the length of the network representation of a record*
- [byteLengthOfNetworkRecord](#) *Get the length of the network representation of a record*
- [bzero](#)
- [callbackFunction](#)
- [camouflageType](#)
- [canRead](#)
- [capabilities](#)
- [capabilities](#)
- [capabilitiesOffset](#)
- [caseCompare](#) *Case-insensitive string compare*
- [category](#)
- [category](#)
- [category](#)
- [changeOptionsBit](#)

- [changeOptions](#)
- [changeSize](#) *Changes the size of the message*
- [changeSize](#) *Changes the size of the datumRecSet, by asking my PDU to change size, then reinitializing to use the right piece of the PDU's buffer*
- [checkGoAway](#)
- [classOfEnv](#)
- [cleanup](#)
- [clearAll](#) *zeros out all bytes in the network representation*
- [clearData](#) *zeros out all bytes after the PDU header*
- [clear](#) *removes all parts and parameters from list Non-virtual since called from ctor/dtor*
- [clear](#)
- [clear](#) *deletes the string, and sets the myString pointer to NULL non-virtual because only called by destructor, and non-virtual members myString is set to NULL, so it should be allocated some memory if operators are to be used on the string*
- [clearField1Flags](#) *Clear a flag and all following flags in appearance field 1*
- [clearFireFlags](#) *Clear the flag and all following flags in the fire flags field*
- [clearList](#)
- [clearTspiFlags](#) *Clear the flag and all following flags in the TSPI field*
- [clone](#) *return an empty object of the same type as this*
- [clone](#) *Returns a new'ed clone of the instance*
- [clone](#) *Create a copy of this object*
- [clone](#) *return an empty object of the same type as this*
- [clone](#) *make a copy of this hash key*
- [clone](#) *Returns a clone of the instance*
- [clone](#) *Create a copy of this object*
- [clone](#) *return an empty object of the same type as this*
- [clone](#) *return an empty object of the same type as this*
- [clone](#) *return an empty object of the same type as this*
- [clone](#) *return an empty object of the same type as this*
- [clone](#) *return an empty object of the same type as this*
- [clone](#) *make a copy of this hash key*
- [clone](#) *make a copy of this hash key*
- [clone](#) *return an empty object of the same type as this*
- [clone](#) *return an empty object of the same type as this*
- [clone](#) *Create a copy of this object*
- [clone](#) *return an empty object of the same type as this*
- [clone](#) *make a copy of this hash key*
- [clone](#) *make a copy of this hash key*
- [cmdType](#)
- [codeName](#)

- [codeName](#)
- [codeName](#)
- [coefficientOfRestitution](#)
- [collidId](#)
- [collidingEntityID](#)
- [collisionInterResultXX](#)
- [collisionInterResultXY](#)
- [collisionInterResultXZ](#)
- [collisionInterResultYY](#)
- [collisionInterResultYZ](#)
- [collisionInterResultZZ](#)
- [collisionType](#)
- [combine](#) *Constructs a new filename by combining the path with filename, which is an absolute or relative path*
- [commClass](#)
- [commType](#)
- [computeHashValue](#) *compute the hashed value for this key*
- [computeHashValue](#) *compute the hashed value for this key*
- [computeHashValue](#) *compute the hashed value for this key*
- [computeHashValue](#) *compute the hashed value for this key*
- [computeHashValue](#) *compute the hashed value for this key*
- [computeParameter](#) *Get the parameter type from the part and parameter value*
- [computePartType](#) *Get the part type from the part and parameter value*
- [computeStatus](#)
- [computeStatus](#)
- [computedBeamDataLength](#)
- [computedBeamDataLength](#)
- [computedSystemDataLength](#)
- [computedSystemDataLength](#)
- [confirmAttributeOwnershipAcquisitionCancellation](#)
- [connectionEstablished](#) *Determines if the tcp socket has connected successfully*
- [constDatumValPtr](#) *Get a const pointer to the datum value inside of the record*
- [constNetStruct](#) *Cast the buffer to a const DtNetVarDatum struct*
- [constantGrid](#)
- [contactVelocity](#)
- [controlType](#)
- [coordSystem](#)
- **coordTrans :**
 - in [DtCoordTransform](#)
 - in [DtCoordTranslation](#)

- [coordinateOffsetXi](#)
- [coordinateScaleXi](#)
- [copyList](#) *Copy the list*
- [count](#)
- [count](#) *Get number of objects*
- [count](#) *returns the number of occurrences of the character*
- [country](#)
- [country](#)
- [create](#)
- [create](#) *Create a DtAcknowledgeRPdu*
- [create](#)
- [create](#) *Create a DtActionRequestRPdu*
- [create](#)
- [create](#) *Create a DtActionResponseRPdu*
- [create](#) *Create a DtAggregateStatePdu*
- [create](#)
- [create](#) *Create a DtAreaObjectStatePdu*
- [create](#) *Create a DtCollisionElasticPdu*
- [create](#)
- [create](#)
- [create](#) *Create a DtCommentRPdu*
- [create](#)
- [create](#) *Create a DtCreateEntityRPdu*
- [create](#)
- [create](#)
- [create](#) *Create a DtDataQueryRPdu*
- [create](#) *Create a DtDataRPdu*
- [create](#)
- [create](#)
- [create](#)
- [create](#)
- [create](#)
- [create](#)
- [create](#)
- [create](#)
- [create](#)
- [create](#)
- [create](#) *Create a DtEntityStateUpdatePdu*

- [create](#) *Create a DtEnvironmentalProcessPdu*
- [create](#)
- [create](#) *Create a DtEventReptRPdu*
- [create](#)
- [create](#) *Create a DtGriddedDataPdu*
- [create](#) *Create a DtIffAtcNav aidsPdu*
- [create](#) *Create a DtIntercomControlPdu*
- [create](#) *Create a DtIntercomSignalPdu*
- [create](#) *Create a DtIsGroupOfPdu*
- [create](#) *Create a DtIsPartOfPdu*
- [create](#)
- [create](#) *Create a DtLeAppearancePdu*
- [create](#) *Create a DtLeArticulatedPartsPdu*
- [create](#) *Create a DtLeDetonationPdu*
- [create](#) *Create a DtLeFirePdu*
- [create](#) *Create a DtLeTspiPdu*
- [create](#) *Create a PDU representation from a network packet*
- [create](#) *Create a DtLinearObjectStatePdu*
- [create](#)
- [create](#) *Create a DtMinefieldDataPdu*
- [create](#) *Create a DtMinefieldQueryPdu*
- [create](#) *Create a DtMinefieldResponseNackPdu*
- [create](#) *Create a DtMinefieldStatePdu*
- [create](#) *Create a DtPointObjectStatePdu*
- [create](#)
- [create](#)
- [create](#)
- [create](#) *Create a DtRecordPdu*
- [create](#) *Create a DtRecordQueryPdu*
- [create](#)
- [create](#) *Create a DtRemoveEntityRPdu*
- **create :**
 - in [DtRepairCompletePdu](#)
 - in [DtRepairResponsePdu](#)
- **create :**
 - in [DtResupplyCancelPdu](#)
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [create](#)

- [create](#)
- [create](#) *Create a DtSetDataRPdu*
- [create](#) *Create a DtSetRecordPdu*
- [create](#)
- [create](#)
- [create](#) *Create a DtStartResumeRPdu*
- [create](#)
- [create](#) *Create a DtStopFreezeRPdu*
- [create](#) *Create a DtSuppEmisEntStatePdu*
- [create](#) *Create a DtTransferControlRequestPdu*
- [create](#)
- [create](#) *Create a DtUnderwaterAcousticPdu*
- [create](#)
- [create](#)
- [createPdu](#) *Create the specified pdu*
- [cryptoKeyId](#)
- [cryptoKeyId](#)
- [cryptoMode](#)
- [cryptoSystem](#)
- [cryptoSystem](#)
- [currentHeaderMatchesFirstInBundle](#)
- [currentShaftRPM](#)
- [currentTimeForStamping](#) *Return time to send with outgoing messages (default is sim time if absolute timestamping, real time otherwise)*
- [dRAgorithm](#)
- [damageState](#)
- [dataByte](#)
- [dataByte](#)
- [dataByte](#)
- [data](#) *Returns the data element contained in the DtListItem envelope*
- [dataFilterBit](#)
- [dataFilter](#)
- [dataFilter](#)
- [dataLeftInCurrentBundle](#)
- [dataLength](#)
- [dataSection](#)
- [dataSection](#) *Returns the data portion of the PDU*
- [dataSection](#)
- [dataSection](#) *Returns the data portion of the net structure*

- [dataSection](#)
- [dataSection](#) Returns the data portion of the pdu
- [dataSection](#)
- [dataSection](#) Returns the data portion of the net structure
- [dataSection](#)
- [dataSection](#) Returns the data portion of the net structure
- [dataSection](#) Returns a pointer to the data section of the PDU - the part that starts with the numberOfFixedFields field
- [dataSection](#)
- [dataSection](#) Returns the data portion of the net structure
- [dataValue](#)
- [dataValue](#)
- [datumBitLength](#)
- [datumByteLength](#) Get the datum length in bytes (padded to an 8 byte boundary)
- [datumByteSize](#) Compute the total datum size in bytes from the datum bit length
- [datumID](#)
- [datumID](#)
- [datumId](#) Returns the datum param found in the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFixed or DtVar
- [datumId](#) Returns the datum param found in the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFIXED or DtVAR
- [datumRecSet](#) Returns our datum record set helper object
- [datumRecordOffset](#) Returns an offset from the beginning of variable length data to a particular variable datum record
- [datumSizeBits](#) Returns the number of bits of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFixed or DtVar
- [datumSizeBits](#) Returns the number of bits of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFIXED or DtVAR
- [datumSizeBytes](#) Returns the number of bytes of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFixed or DtVar
- [datumSizeBytes](#) Returns the number of bytes of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFIXED or DtVAR
- [datumValByteArray](#)
- [datumValByteArray](#)
- [datumValByteArray](#)
- [datumValFloat32](#) ***** INSPECTING DATUM VALUES BY INDEX ***** If you know the data type of the value contained in a particular record, use one of the following to return it
- [datumValFloat32](#)
- [datumValFloat32](#) ***** INSPECTING DATUM VALUES BY PARAM ID ***** If you know the data type of the value contained in a particular record, use one of the following to return it
- [datumValFloat32](#)
- [datumValFloat64](#)

- [datumValFloat64](#)
- [datumValFloat64](#)
- [datumValFloat64](#)
- [datumValInt16](#)
- [datumValInt32](#)
- [datumValInt32](#)
- [datumValInt32](#)
- [datumValInt32](#)
- [datumValPtr](#) Returns a pointer to the index'th fixed or variable record in the network representation, depending on whether fixedOrVar is DtFixed or DtVar
- [datumValPtr](#)
- [datumValPtr](#) Returns a pointer to the index'th fixed or variable record in the network representation, depending on whether fixedOrVar is DtFIXED or DtVAR
- [datumValPtr](#) Get a pointer to the datum value inside of the record
- [datumValU16](#)
- [datumValU32](#)
- [datumValU8](#)
- [datumValUnsigned32](#)
- [datumValUnsigned32](#)
- [datumValUnsigned32](#)
- [deadReckonBodyToGeoc](#) If the last dead reckoned body to geocentri coordinate matrix is not valid for the specified time, recalculate it and same the validity time
- [deadReckonEuler](#) If the last dead reckoned orientation (euler angles) is not valid for the specified time, recalculate it and same the validity time
- [deadReckonLoc](#) If the last dead reckoned position is not valid for the specified time, recalculate it and same the validity time
- [deadReckonOrientation](#) Calculate the orientation matrix
- [deadReckonPosition](#) Calculate the dead reckon position information
- [deadReckonedBodyToGeoc](#) Get the dead-reckoned body to geocentric coordinate matrix
- [deadReckonedEuler](#)
- [deadReckonedLocation](#) Get the dead-reckoned location at the specified time
- [deadReckonedVelocity](#) Get the dead-reckoned velocity at the specified time
- [deadReckoningAlgorithm](#)
- [deadReckoningAlgorithm](#)
- [deadReckoningParametersOffset](#)
- [decodeArtParts](#) decode the articulated parts portion of a PDU into an EntityStateRepository
- [decodeAttachedParts](#) decode the attached parts portion of a PDU into an EntityStateRepository
- [decode](#) Decode an aggregate state PDU and update a state repository
- [decode](#) decode a PDU into an EntityStateRepository
- [decoder](#)

- [decoder](#) Returns a pointer to the Decoder being used to decode state messages into our ASR
- [decoder](#) Returns a pointer to the Decoder being used to decode state messages into our ESR
- [delRead](#)
- [deleteBytes](#) Deletes numBytes bytes at the indicated offset in the net rep
- [deleteBytes](#) Deletes numBytes bytes at the indicated addr within the net rep, by asking my PDU to insert bytes, then reinitializing to use the right piece of the PDU's buffer
- [deleteIfNoExternal](#) Delete the internal buffer if there is no external buffer
- [deltaX](#)
- [deltaY](#)
- [deltaZ](#)
- [density](#)
- [designatedEntityId](#)
- [designatedObject](#)
- [designatingEntityId](#)
- [designatorCode](#)
- [designatorCode](#)
- [designatorId](#)
- [designatorPower](#)
- [designatorSpotLocation](#)
- [designatorSpotRelative](#)
- [designatorSpotWorld](#)
- [designatorWavelength](#)
- [destDeviceID](#)
- [destEntityID](#)
- [destEntityID](#)
- [destEquipID](#)
- [destGroupBitField](#)
- [destLineID](#)
- [destLineID](#)
- [destLineStateCmd](#)
- [destLineStateCmd](#)
- [destPriority](#)
- [destPriority](#)
- [destinationAddr](#) Returns the current destination address for the PDU
- [detail](#)
- [detailedModulationType](#)
- [deton2FlagFieldOffset](#)
- [detonFlag](#) Get the value for a specific flag from the first flags field
- [detonFlagFieldOffset](#) Returns the offset (in bytes) of a field from variable data region

- [detonFlagsField](#) *Get flags field 1 (this field is automatically maintained by the PDU as other PDU fields are set)*
- [detonationResult](#)
- [deviceId](#)
- [dfltAction](#) *The default action is to dead-reckon*
- **dfltAction :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [dfltAction](#) *The default action is to dead-reckon*
- [dfltAggDimensionThreshold](#)
- [dfltFractionalThreshold](#)
- [dfltPacketGood](#)
- [dfltRotationThreshold](#)
- [dfltRotationThreshold](#)
- [dfltRotationThreshold](#)
- [dfltRotationTraceThreshold](#)
- **dfltSmoothPeriod :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [dfltSmoothPeriod](#)
- [dfltThreshMode](#)
- [dfltTimeThreshold](#)
- [dfltTimeThreshold](#)
- [dfltTranslationThreshold](#)
- [dfltTranslationThreshold](#)
- [dfltTranslationThreshold](#)
- [dimensions](#)
- [dimensions](#)
- [disableFiltering](#)
- [disableLayer2](#)
- [disableTest](#)
- [disconnectFromServer](#)
- [discoverObjectInstance](#) //////////////////////////////////
- [doTimeouts](#) *Remove any remote objects which have not been updated during the timeout period*
- [doTimeoutsPostDrainCb](#) *calls doTimeouts*
- [domain](#)
- [domain](#)
- [domainFinalXi](#)
- [domainInitialXi](#)
- [domainNumPointsXi](#)

- [drainInput](#) *Read and process messages from the network until either no more messages, or for timeout seconds, whichever happens first*
- [drainInput](#) *Normal way of processing input from the network*
- [drainNotifySocket](#)
- [dropMulticastGroup](#)
- [dropMulticastGroup](#)
- [dropMulticastGroup](#)
- [east](#)
- [electricFieldX](#) *Get electric field X*
- [electricFieldZ](#) *Get electric field Z*
- [elevationBeamwidth](#) *Get the elevation beam width*
- [elevationCenter](#)
- [elevationCenter](#)
- [elevationError](#)
- [elevationSweep](#)
- [elevationSweep](#)
- [emEmissionCb](#) *callback for the exercise connection*
- [emitterBeamAddr](#)
- [emitterBeamAddr](#)
- [emitterFunction](#)
- [emitterId](#)
- [emitterId](#)
- [emitterId](#)
- [emitterName](#)
- [emitterNumber](#)
- [emitterSysRep](#) *Get the emitter system's state repository*
- [emitterSystemAddr](#)
- [emitterSystemAddr](#)
- [emitterSystemRecordOffset](#) *Returns a buffer offset to the given acoustic emitter system*
- [emittingEntityID](#)
- [emittingEntityID](#)
- [emittingEntityId](#)
- [emittingEntityId](#)
- [emittingSystemId](#)
- [empty](#) *Remove all keys and data from the list*
- [emptyList](#) *Empty the list*
- [emptyLists](#) *Empty the object list and the timeout list*
- [enableFiltering](#) *Enable/disable PDU filtering*
- [enableLayer2](#) *Enable/Disable layer 2*

- [encoder](#)
- [encodingClass](#)
- [encodingClass](#)
- [encodingType](#)
- [encodingType](#)
- [engineRPM](#)
- [engineSmokeOn](#)
- [entities](#)
- [entityAdded](#) *Called after an entity is added and before addition callbacks are processed*
- [entityCoords](#)
- [entity](#)
- [entityID](#)
- [entityID](#)
- [entityID](#)
- [entityID](#)
- [entityID](#)
- [entityID](#)
- [entityID](#)
- [entityID](#)
- [entityID_Offset](#) *Returns an offset from the beginning of variable length data to a particular entity ID*
- [entityId](#)
- [entityId](#)
- [entityId](#)
- [entityId](#)
- [entityId](#) *Returns the entity identifier*
- [entityId](#)
- [entityId](#)
- [entityLocation](#)
- [entityMarkingOffset](#)
- [entityNum](#)
- [entitySpeed](#)
- [entitySpeed](#)
- [entitySpeed](#)
- [entityStateCb](#) *callback for the exercise connection*
- [entityStateRep](#) *Get the entity's state repository*
- [entityStateRep](#)
- [entityStateRep](#) *Returns a pointer to the EntityStateRepository - through which you can inspect current state information*

- **entityToAdd :**
 - in [DtGroupVCPdu](#)
 - in [DtModelessVCPdu](#)
- **entityToRemove :**
 - in [DtGroupVCPdu](#)
 - in [DtModelessVCPdu](#)
- [entityTypeAtStation](#) *Returns the entityType at a particular station*
- [entityType](#) *Returns the entityType for a part, by index*
- [entityType](#)
- [entityType](#)
- [entityType](#)
- [entityType](#)
- [entityTypeOffset](#)
- [envSimAppID](#)
- [environmentType](#)
- [environmentType](#)
- [environmentalProcessID](#)
- [environmentalStatusBit](#)
- [environmentalStatus](#)
- [equivalent](#) *Returns true if we contain the exact same information as other*
- [esr](#)
- [esr](#) *Synonym for emitterSysRep()*
- [esr](#) *Synonym for entityStateRep()*
- [eulerTrans](#)
- [evalTest](#) *No-op*
- [evalTest](#)
- [eventID](#)
- [eventID](#)
- [eventID](#)
- [eventID_Event](#)
- [eventID_Offset1](#)
- [eventID_Offset1](#)
- [eventID_Offset2](#)
- [eventID_Offset2](#)
- [eventId](#)
- [eventId](#)
- [eventId](#)
- [eventId](#)
- [eventId](#)

- [eventId](#)
- [eventId](#)
- [eventNum](#)
- [eventType](#)
- [eventType](#)
- [eventType](#)
- [eventType](#)
- [executePostDrainCbs](#) *should be called by derived class' drainInputs*
- [exerciseConn](#) *Get our exercise connection*
- [exerciseConn](#) *Get our exercise connection*
- [exerciseConn](#) *Get my exerciseConn*
- [exerciseld](#)
- [exerciseld](#)
- [exerciseld](#)
- [exerciseld](#)
- [extendTest](#)
- [extendTest](#)
- [extra](#)
- [federationNotRestored](#)
- [federationNotSaved](#)
- [federationRestoreBegun](#)
- [federationRestored](#)
- [federationSaved](#)
- [federationSynchronized](#)
- [feedSmoothing](#) *provide inputs to the actual smoothing algorithm*
- [fieldNumber](#)
- [fieldOffset](#)
- [fieldScale](#)
- [filename](#)
- [filename](#)
- [fillDataPdu](#)
- [fillNetParams](#) *fills in netParams based on current values*
- [fillOutArtAndAttParts](#) *Copies current articulated and attached part info from the ESR to the PDU*
- [fillOut](#) *fill out a state message which reflects a state repository*
- [fillOut](#) *Copy current ESR data to the pdu*
- [fillSpecList](#)
- [finalPdulsSetAction](#) *Called when an entity sends its final PDU*
- [findItem](#)

- [findNextPdu](#)
- [findNextValidParamType](#) Find the next valid parameter type
- [findOwnId](#) returns an id of (-1,-1,-1)
- [findOwnId](#)
- [findString](#) Returns 1 after reading a line containing str Returns 0 if hit EOF without finding such a line
- [fireFlag](#) Get the value for a specific flag from the flags field
- [fireFlagFieldOffset](#) Returns the offset (in bytes) of a field from variable data region
- [fireFlagsField](#) Get flags field (this field is automatically maintained by the PDU as other PDU fields are set)
- [fireMissionIndex](#)
- [firePowerDisabled](#)
- [firingEntityID](#)
- [firingEntityID](#)
- [first](#) Return first and last item on the list
- [first](#) get the first and last elements of the aggregate list
- [first](#) get the first and last elements of the entity list
- [first](#) get the first and last elements of the entity list
- [firstObj](#) Get the first/last object in the list
- [firstValidParamType](#) Get the first valid (added) parameter type
- [fixedDatumId](#)
- [fixedDatumId](#)
- [fixedDatumId](#) Returns the datum param found in the index'th fixed datum record
- [fixedDatumId](#) Returns the datum param found in the index'th fixed datum record
- [flamesPresent](#)
- [flush](#)
- [forceId](#)
- [forceId](#)
- [forceId](#)
- [forceId](#)
- [forceId](#)
- [forceId](#)
- [forceId_Offset](#)
- [forceId](#)
- [forceId](#)
- [forceId](#)
- [forceUpdate](#) Force a state update to be sent during the next call to tick, even if nothing would normally cause a send
- [formation](#)
- [formation](#)
- [fractionalThreshold](#)
- [frequencyBandwidth](#)

- [frequency](#)
- [frequency](#)
- [frequency](#) *Get either the upper or lower word of the 64 bit frequency*
- [frequency](#)
- [frequencyHopInUse](#)
- [frequencyRange](#)
- [frozenBehavior](#)
- [frozenBehavior](#)
- [frozen](#)
- [frozen](#)
- [fuelStatus](#)
- [fuelStatus](#)
- [fuelStatus](#)
- [func](#) *Get the callback function*
- [function](#)
- [fuseType](#)
- [fuseType](#)
- [fusingRecordOffset](#) *Returns a buffer offset to the fusing records*
- [gFindOwnId](#)
- [gFindOwnId](#)
- [gGuiseSameAsType](#)
- [geocToUtmMatrix](#) *Get the matrix which transforms form geocentric to UTM*
- [geocentric](#) *Get geocentric coordinate equivalent (equivalent to geocentric)*

- [getArtPartParameters](#)
- [getAttPartParameters](#)
- [getAttPartParameters](#)
- [getAttachedType](#)
- [getAudioAppearance](#)
- [getBasicFixedWingAircraftRecord](#)
- [getBasicGroundCombatSoldierRecord](#)
- [getBasicGroundCombatVehicleRecord](#)
- [getBasicRotorWingAircraftRecord](#)
- [getBeam](#)
- [getBurstFuse](#)
- [getBurstFuse](#)
- [getBurstQuantity](#)
- [getBurstQuantity](#)
- [getBurstRate](#)
- [getBurstRate](#)
- [getBurstWarhead](#)
- [getBurstWarhead](#)
- [getCapabilities](#)
- [getCapability](#)
- [getDataByte](#)
- [getDataBytes](#)
- [getData](#)
- [getData](#)
- [getData](#)
- [getDataFilterField](#)
- [getDataFloat32](#)
- [getDataFloat64](#)
- [getDataInt16](#)
- [getDataInt32](#)
- [getDataU16](#)
- [getDataU32](#)
- [getDatumBitLength](#)
- [getDatumID](#)
- [getDatumValByteArray](#)
- [getDatumValByteArray](#)
- [getDatumValFloat32](#)
- [getDatumValFloat64](#)

- [getDatumValInt16](#)
- [getDatumValInt32](#)
- [getDatumValPtr](#) *Get a pointer to datum value*
- [getDatumValU16](#)
- [getDatumValU32](#)
- [getDatumValU8](#)
- [getDeadReckoningParameters](#)
- [getDeton2Flag](#) *Get the value for a specific flag from the second flags field*
- [getDeton2FlagsField](#) *Get flags field 2 (this field is automatically maintained by the PDU as other PDU fields are set)*
- [getEM_Appearance](#)
- [getEmitterSystem](#)
- [getEnhancedFixedWingAircraftRecord](#)
- [getEnhancedGroundCombatSoldierRecord](#)
- [getEnhancedGroundCombatVehicleRecord](#)
- [getEnhancedRotorWingAircraftRecord](#)
- [getEntityID](#)
- [getEntityID](#)
- [getEntityTypeAtStation](#)
- [getEntityType](#)
- [getEntityType](#)
- [getEventID_Application](#)
- [getEventID_Application](#)
- [getEventID_Event](#)
- [getEventID_Site](#)
- [getEventID_Site](#)
- [getForceID](#)
- [getGeocentric](#) *Get geocentric coordinate equivalent (convert geodetic to geocentric)*
- [getGeocentric](#) *Get geocentric coordinate equivalent (convert UTM to geocentric)*
- [getGeodetic](#) *Get geodetic coordinate equivalent (convert UTM to geodetic)*
- [getGridAxisRecord](#)
- [getGridDataRecord](#)
- [getGroundBurialDepthOffset](#)
- [getGroundLogisticsVehicleRecord](#)
- [getIR_Appearance](#)
- [getIndex](#) *Looks through all fixed and variable records for one containing param as the parameter ID*
- [getIndex](#) *Looks through all fixed and variable records for one containing param as the parameter ID*
- [getLinearSegmentParameter](#)
- [getLocation](#)

- [getLocationEntity](#)
- [getLocationWorld](#)
- [getMarkingCharacterSet](#)
- [getMarkingText](#)
- [getMarkingText](#)
- [getMarkingText](#)
- [getMeasuredSpeed](#)
- [getMineEmplacementTime](#)
- [getMineEntityID](#)
- [getMineLocation](#)
- [getMineOrientation](#)
- [getMineType](#)
- [getMinefieldAlgae](#)
- [getMunitionID_Application](#)
- [getMunitionID_Application](#)
- [getMunitionID_Entity](#)
- [getMunitionID_Entity](#)
- [getMunitionID_Site](#)
- [getMunitionID_Site](#)
- [getNetAntennaParams](#)
- [getNetModulationParams](#)
- [getNozzleSystem](#)
- [getNumTripWires](#)
- [getNumVertices](#)
- **getOffset :**
 - in [DtCoordTransform](#)
 - in [DtCoordTranslation](#)
- [getOrientation](#)
- [getOrientation](#)
- [getOrientation](#)
- [getOrientationError](#)
- [getPDU_SequenceNumber](#)
- [getPaintScheme](#)
- [getParmData](#)
- [getParmRecord](#)
- [getPerimeterPoint](#)
- [getPerimeterPoint](#)
- [getPointLocation](#)
- [getPositionError](#)

- [getPrimaryFusing](#)
- [getPropulsionSystem](#)
- [getPsMsg](#)
- [getRecordID](#)
- [getRecordSet](#)
- [getRecordType](#) *Get the record type for parameter record parmIndex*
- [getReflectance](#)
- [getScalarDetectionCoefficient](#)
- [getSecondaryFusing](#)
- [getSensorTypeCategory](#)
- [getSensorTypeCategory](#) *Get the sensor type category for a record*
- [getSensorTypeSubcategory](#)
- [getSensorTypeSubcategory](#)
- [getShaftSpeed](#)
- [getSilentAggregateList](#)
- [getSilentAggregateList](#)
- [getSilentEntityList](#)
- [getSilentEntityList](#)
- [getSnowBurialDepthOffset](#)
- [getStationNumber](#)
- [getSubaggregateID](#)
- [getSubaggregateID](#)
- [getSystemSpecificData](#)
- [getTargetEntityID](#)
- [getTargetEntityID](#)
- [getThermalContrast](#)
- [getTranslations](#)
- [getVal](#)
- [getVariableDatumRecord](#)
- [getVariableDatumRecord](#)
- [getVelocity](#)
- [getVertex](#)
- [getVisualAppearanceAttribute](#)
- [getVisualAppearance](#)
- [getWaterBurialDepthOffset](#)
- **getdata :**
 - in [DtDatumMapEntry](#)
 - in [DtDmeGlobalFcns](#)
- [getopt](#)

- [globalId](#) *Get the global object designator - same as id for DIS*
- [globalId](#) *Get the global object designator - same as id for DIS*
- [globalId](#) *Get the aggregate's object's global object designator - for DIS, it's the same as its DtEntityIdentifier*
- [globalId](#) *Get the entity's object's global object designator - for DIS, it's the same as its DtEntityIdentifier*
- [gridAxisRecordLength](#) *Returns the length of a given grid axis record*
- [gridAxisRecordOffset](#) *Returns a buffer offset to the given grid axis record*
- [gridAxisRecordType](#) *Get the grid axes record type for a particular record*
- [gridDataRecordLength](#) *Returns the length of a given grid data record*
- [gridDataRecordOffset](#) *Returns a buffer offset to the given grid data record*
- [gridDataRecordType](#) *Get the grid data record type for a particular record*
- [gridDeclination](#) *Get the heading that northing is off of north*
- [groundBurialDepthRecordOffset](#) *Returns a buffer offset to the ground burial depth records*
- [groundMaintenanceStatus](#)
- [groupBitField](#)
- [groupCmd](#)
- [groupEntityID](#)
- [guessTimeValid](#) *Since timestamp only represents number of seconds past the hour, guessTimeValid chooses the hour that would make the timestamp closest to reference time, and returns a time that is timestamp seconds past that hour*
- [guise](#)
- [guise](#)
- [guiseSameAsType](#) *Returns a special DtEntityType that indicates to the DtEntityPublisher constructor that the entity type should also be used as the guise*
- [gunElevation](#)
- [gunElevation](#)
- [gunElevationRate](#)
- [gunElevationRate](#)
- [hasAmmunitionSupplyCap](#)
- [hasFuelSupplyCap](#)
- [hasRecoveryCap](#)
- [hasRepairCap](#)
- [hashKey](#) *get the key for the hashed item*
- [hashKey](#) *Get a hashkey for this object*
- [hashKey](#) *Get a hashkey for this object*
- [hashKey](#) *Get a hashkey for this object*
- [hashKey](#) *Get a hashkey for this object*
- [hashKeyFromStateMessage](#) *Get a hash key from a state message*
- [hashKeyFromStateMessage](#) *Get a hash key from a state message*
- [hashKeyFromStateMessage](#) *Get a hash key from a state message*
- [hashKeyFromStateMessage](#) *Get a hash key from a state message*

- [hashValue](#) *get the hashed value for this key*
- [hatchState](#)
- [headAzimuth](#)
- [headElevation](#)
- [headElevationRate](#)
- [headScanRate](#)
- **heading :**
 - in [DtGroupVCPdu](#)
 - in [DtOrbitVCPdu](#)
- **headingOffset :**
 - in [DtCompassVCPdu](#)
 - in [DtGroupVCPdu](#)
 - in [DtOrbitVCPdu](#)
- [heartbeat](#)
- [highDensity](#)
- [horDeflection](#)
- [horizontalDeviation](#)
- [horizontalDeviation](#)
- [horizontalError](#)
- [host](#)
- [host](#)
- [hostId](#)
- [i](#) *Get a (1,0,0) vector*
- [id](#) *Get the entity's identifier*
- [id](#) *Get the aggregate's identifier*
- [id](#) *Get the entity's identifier*
- [id](#) *Get the entity's identifier*
- [id](#)
- [id](#) *Get the aggregate's id*
- [id](#) *Get the entity's id*
- [idString](#) *Get a string identification of this object*
- [idString](#) *Get a string identification of this object*
- [idString](#) *Get a string identification of this object*
- [idString](#) *Get a string identification of this object*
- [identity](#)
- [ignoreFlags](#)
- [immobilized](#)
- **indexInBounds :**
 - in [DtResupplyOfferPdu](#)

- in [DtResupplyReceivedPdu](#)
- [indexInBounds](#)
- [informAttributeOwnership](#)
- [informationLayers](#)
- [infraRedSigReplIndex](#)
- [init0](#)
- [initArtAndAttachedParts](#)
- [initArtParts](#)
- [initArtParts](#)
- [initArtParts](#) *initialize articulated parts*
- [initDataPdu](#)
- [initDataSection](#)
- [initDataSection](#) *Initialize the data section of the PDU to reflect the fixed parameters and variable parameters and sizes indicated*
- [initDatumIds](#)
- [initDatumIds](#) *Initialize the fixed and variable fields*
- [init](#) *not virtual since called from constructor*
- [init](#) *The init functions don't set values for the parameters, they merely set up the part and parameter types*
- [init](#) *Stations is a zero-terminated array of stations*
- [init](#)
- [init](#)
- [init](#) *Initialize to reflect the fixed parameters and variable parameters and sizes indicated*
- [init](#)
- [init](#) *not virtual since called from constructor*
- [init](#)
- [init](#)
- [init](#)
- **init :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [initHeader](#) *Initializes PDU header*
- [initMsg](#)
- [initPdu](#) *Must be called after default ctor before PDU is completely constructed*
- [init_iodev](#)
- [init_read](#)
- [init_write](#)
- [initialIndexXi](#)
- [initiateFederateRestore](#)
- [initiateFederateSave](#)

- **input :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [inputSource](#)
- [inputSource](#)
- [insertBytes](#) *Inserts numBytes zeroed-out bytes at the indicated address*
- [insertBytes](#) *Inserts numBytes zeroed-out bytes at the indicated addr within the net rep, by asking my PDU to insert bytes, then reinitializing to use the right piece of the PDU's buffer*
- [install](#)
- [interleafFactor](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*

- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *identity for use by toolkit*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the PDU kind associated with the derived PDU class*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#)
- **internalGetPduKind :**
 - in [DtRepairCompletePdu](#)
 - in [DtRepairResponsePdu](#)
- **internalGetPduKind :**
 - in [DtResupplyCancelPdu](#)
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*

- [internalGetPduKind](#)
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#) *Returns the internal PDU kind*
- [internalGetPduKind](#)
- [internalGetPduKind](#)
- [internalSetMarkingText](#) *Set the marking text*
- [ipAddr](#)
- [ipAddr](#)
- [isBundling](#)
- [isConcealed](#)
- [isInternal](#)
- [isMember](#) *Returns DtTrue if the give BSR is in this ESR's list of BSRs*
- [isOk](#)
- [isOk](#)
- [isOk](#)
- [isOk](#)
- [isOk](#)
- [isRotational](#) *Returns 1 if the dead reckoning algorithm is a rotational one, 0 otherwise*
- [isUnbundling](#)
- [issueld](#) *INSPECTOR FUNCTIONS ******
- [issuingEntityID](#)
- [itemLookup](#) *Lookup something in the list, return DtListItem*
- [itemLookup](#) *Returns the DtListItem envelope object containing the indicated data element*
- [j](#) *Get a (0,1,0) vector*
- [jammingModeSequence](#)
- [jammingModeSequence](#)
- [jammingModeSequence](#)
- [k](#) *Get a (0,0,1) vector*
- [keys](#) *Returns a DtList of DtHashKey*'s representing the set of keys with which data is currently associated*
- [kind](#) *int entityKind() const {return(DtEntityKind);}*
- [kind](#)
- [kind](#)
- [kindToFamily](#) *Maps PDU kind to ProtocolFamily*
- [lasedEntityId](#)
- [lasedSpotRelative](#)

- [lasedSpotWorld](#)
- [laserCode](#)
- [lasingEntityId](#)
- [last](#)
- [last](#)
- [last](#)
- [last](#)
- [lastObj](#)
- [lastSetAcceleration](#) *Return the value last passed to setAcceleration, regardless of whether we have an approximator*
- [lastSetLocation](#)
- [lastSetOrientation](#)
- [lastSetRotationalVelocity](#)
- [lastSetVal](#)
- [lastSetVelocity](#)
- [lastSimTimeUpdated](#) *Returns the value of simulation time at the time an update was last received for the entity*
- [lastSourceInetAddr](#) *int isUnbundling(); is not currently implemented*
- [lastSourceInetAddr](#)
- [lastSourceInetAddr](#)
- [lat](#) *inline functions*
- [latitude](#)
- [launcherFlashPresent](#)
- [launcherRaised](#)
- [layer2Header](#) *Get layer header for layer 2 This record is set automatically by the PDU class*
- [layerInfo](#)
- [layerLength](#)
- [layerNumber](#)
- [layerPresent](#)
- [layerPresent](#) *Query the layers field of the fundamental operational data record*
- [length](#) *Get size of message*
- [length](#) *return the length of the string*
- [lgrCommand](#) *get message-type for proper casting to a subclass*
- [lifeformState](#)
- [lightState](#)
- [linearAcceleration](#)
- [linearAcceleration](#)
- [list](#) *Get a DtList containing all of the void* data that you have listed, regardless of keys*
- [list](#) *Returns a DtList of void**s which are the data that you have added to the hashlist (independent of hash keys)*
- [list](#) *Get a DtList containing all of the void* data that you have listed, regardless of keys*

- [illegible]

- [loadNetRecord](#)
- [loadNetRecord](#)
- [loadNetRecord](#)
- [loadNetRecord](#)
- [loadNetRecord](#)
- [loadNetRecord](#)
- [loadNetRecord](#)
- [loadNetRecord](#)
- [loadNetRecord](#)
- [loadNetRecordSet](#)
- [localId](#) *Get the entity's localId - same as id for DIS*
- [localId](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [location](#)
- [locationEntityOffset](#)
- [locationOfImpact](#)
- [locationOffset](#)
- [locationOffset](#)
- [locationWorldOffset](#)
- [lon](#)
- [longitude](#)
- [lookup](#)
- [lookup](#) *Lookup the data associated with key*
- [lookup](#) *Lookup something in the list, return data pointer*

- [lookup](#) *Lookup the data associated with key*
- [lookup](#) *lookup an aggregate from its object ID*
- [lookup](#) *lookup an emitter from its object ID and system ID*
- [lookup](#) *lookup an entity from its object ID*
- [majorModType](#)
- [majorModulationType](#)
- [makeEntityId](#) *Returns an entity ID consisting of the DtExerciseConn's site and host numbers, and entityNum as the entity number*
- [makeResponseDataPdu](#)
- [markingCharSet](#)
- [markingCharSet](#)
- [markingCharacterSet](#)
- [markingCharacterSet](#)
- [markingText](#)
- [markingText](#)
- [markingText](#)
- [mass](#)
- [mass](#)
- [masterChannelID](#)
- [masterDeviceID](#)
- [masterEntityID](#)
- **masterSmoothEnable :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [masterSmoothEnable](#)
- [match](#) == *returns 1 if they are equal, 0 if they are not equal*
- [match](#) *Get whether one callback info equals another*
- **matchPattern :**
 - in [DtEntityIdentifier](#)
 - in [DtEntityType](#)
- [measuredSpeedOffset](#)
- [mineEmplacementTimeRecordOffset](#) *Returns a buffer offset to the mine emplacement time records*
- [mineEntityID_RecordOffset](#) *Returns a buffer offset to the mine entity ID records*
- [mineLocationRecordOffset](#) *Returns a buffer offset to the mine location records*
- [mineOrientationRecordOffset](#) *Returns a buffer offset to the mine orientation records*
- [mineType](#)
- [mineTypeRecordOffset](#) *Returns a buffer offset to a mine type record*
- [minefieldID](#)
- [minefieldID](#)

- [minefieldID](#)
- [minefieldID](#)
- [minefieldProtocolMode](#)
- [minefieldSequenceNumber](#)
- [minefieldType](#)
- [modelType](#)
- [modifications](#)
- [modifications](#)
- [modifications](#)
- [modificationsField](#)
- [modificationsField](#)
- [modificationsField](#)
- [modifier](#)
- [modulationParamsAddr](#)
- [modulationParamsAddr](#)
- [modulationParamsLength](#)
- [modulationParamsLength](#)
- [modulationSystemType](#)
- [modulationType](#)
- [movementSpeed](#)
- [movementSpeed](#)
- [munitionID_Offset1](#)
- [munitionID_Offset1](#)
- [munitionID_Offset2](#)
- [munitionID_Offset2](#)
- [munitionId](#)
- [munitionId](#)
- [munitionOrientationOffset](#)
- [munitionType](#)
- [munitionType](#)
- [nature](#)
- [netBeam](#) *Returns network representation*
- [netDataRep](#) *Returns a pointer to the network representation, as might appear in a data PDU*
- [netHeader](#)
- [netPduStruct](#) *Casts the netPdu element to our network struct Non-virtual since you can't override based on return type*
- [netPduStruct](#) *Casts the netPdu element to our network struct*
- [netPduStruct](#) *Casts the netPdu element to our network struct Non-virtual since you can't override based on return type*

- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- [netPduStruct](#) Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- [netPduStruct](#) Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStruct](#) Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- [netPduStruct](#) Casts the netPdu element to our network struct
- [netPduStructL2](#) Casts the netPdu element to our network struct
- [netRead](#) Note: netRead() creates a new DtPdu on the heap
- [netRecordLength](#) Get the network data length for this list
- [netRecordSetLength](#) Returns the length of data used by the record sets beginning at the location specified by netRecordSetPduStruct()
- [netRecordSetPduStruct](#) Returns a pointer to the beginning of the record set data struct
- [netRecordSetPduStruct](#) Returns a pointer to the beginning of the record set data struct
- [netRecordSetPduStruct](#) Returns a pointer to the beginning of the record set data struct
- [netRecordSetPduStruct](#) Returns a pointer to the beginning of the record set data struct
- [netRep](#) Returns network representation, fills size with size of netRep

- [netRep](#)
- [netRep](#) Returns network representation, fills size with size of netRep
- [netRep](#) Returns network representation, fills size with size of netRep
- [netSize](#) Returns size of network representation
- [netSize](#) Returns size of network representation
- [netSize](#) Returns size of network representation
- [netStruct](#) Cast the buffer to a DtNetVarDatum struct
- [networkByteLength](#) Get the network byte length of this record
- [networkByteLength](#) Get the network byte length of the entire datum record
- [networkDataBuffer](#) Get a pointer to the buffer which stores the network representation of the datum record
- [networkRepresentation](#)
- [newInternalBuffer](#) Sets myInternalBuff to point to a malloc'ed buffer of size bytes
- [newReflectedAggregate](#) Create a new reflected aggregate
- [newReflectedEntity](#) Create a new reflected entity
- [newReflectedObj](#) Create a new object from a state message
- [newReflectedObj](#) Create a new object from a state message
- [newReflectedObj](#) Create a new object from a state message
- [newReflectedObj](#) Create a new object from a state message
- **next :**
 - in [DtListItem](#)
 - in [DtListedObj](#)
- [next](#) get the next and previous elements in the reflected aggregate list
- [next](#) Get the next remote emitterSystem
- [next](#) get the next and previous elements in the reflected entity list
- [nextEventId](#) Return an event ID consisting of the DtExerciseConn's site and host numbers, and the current value of an event number counter
- [nextId](#) Returns the next valid entity ID formed by calling makeEntityId with the current value of a entity number counter
- [nextObj](#) Get the next remote object
- [nextValidParamType](#) Get the next valid parameter type
- [nomenclature](#)
- [nomenclatureVersion](#)
- [north](#)
- [notifyFromObserved](#) Called by DtObservee when something has happened to the observed object
- [notifyMe](#)
- [notifyMe](#)
- [nozzleSystemRecordOffset](#) Returns a buffer offset to the given nozzle system data record
- [nthFixedDataRecord](#) Return a pointer to the index'th fixed data record in the network representation
- [nthFixedDataRecord](#) Return a pointer to the index'th fixed data record in the network representation

- **nthSupply :**
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [nthSupply](#)
- [nthVarDataRecord](#) *Return a pointer to the index'th variable data record in the network representation*
- [nthVarDataRecord](#)
- [nthVarDataRecord](#) *Return a pointer to the index'th variable data record in the network representation*
- [nullId](#)
- [nullId](#)
- [nullString](#)
- **nullType :**
 - in [DtEntityType](#)
 - in [DtRadioEntityType](#)
- [numAdditionalPassiveAct](#)
- [numArtParams](#) *Get the number of articulated parameters*
- [numArtParams](#)
- [numArtParams](#)
- [numArtParts](#)
- [numArtPlusAttachedParams](#)
- [numAttachedParams](#)
- [numAttachedParts](#)
- [numBeams](#)
- [numBeams](#)
- [numBeams](#) *Returns number of beams*
- [numBeams](#)
- [numCallbacks](#) *Get the number of callbacks registered for the PDU kind*
- [numDataBits](#)
- [numDataBits](#)
- [numDataBytes](#)
- [numDataBytes](#)
- [numDataBytes](#)
- [numDataBytes](#)
- [numDataValues](#)
- [numDataValues](#)
- [numEmitterSystems](#)
- [numEntities](#)
- **numEntitiesToAdd :**
 - in [DtGroupVCPdu](#)
 - in [DtModelessVCPdu](#)
- **numEntitiesToRemove :**

- in [DtGroupVCPdu](#)
- in [DtModelessVCPdu](#)
- [numFixedFields](#)
- [numFixedFields](#) *Get the number of fixed fields*
- [numFixedFields](#) *returns number of fixed fields*
- [numFixedFields](#) *returns number of fixed fields*
- [numGridAxes](#)
- [numMineTypes](#)
- [numMissingPDUs](#)
- [numNozzleSystems](#)
- [numObjects](#)
- [numObjects](#)
- [numParameterSets](#) *Get the number of fundamental parameter data records from the secondary operational parameters record*
- [numParameterSets](#)
- [numParams](#) *Get the number of parameters*
- [numParmRecords](#)
- [numPerimeterPoints](#)
- [numPerimeterPoints](#)
- [numPoints](#)
- [numPointsXi](#)
- [numPointsXi](#)
- [numPropulsionSystems](#)
- [numRecordSets](#)
- [numRecords](#) *Get the number of articulation parameter records (this field is automatically set by the PDU when the articulated part list or attached part list is set)*
- [numRecords](#) *Get the number of articulation parameter records (this field is automatically set by the PDU when the articulated part list or attached part list is set)*
- [numRecords](#)
- [numRecordsOffset](#) *Returns the offset (in bytes) of a field in the variable data region*
- [numSamples](#)
- [numSamples](#)
- [numSegments](#)
- [numSensorTypes](#)
- [numShafts](#)
- **numSupplyTypes :**
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [numSupplyTypes](#)
- [numSystems](#)

- [numSystems](#)
- [numTargetsInBeam](#)
- [numTargetsInBeam](#)
- [numVarFields](#)
- [numVarFields](#) *Get the number of variable fields*
- [numVarFields](#) *returns number of variable length fields*
- [numVarFields](#)
- [numVarFields](#) *returns number of variable length fields*
- [numberOfAggregates](#)
- [numberOfAppearanceRecords](#)
- [numberOfBeams](#)
- [numberOfDIS_Entities](#)
- [numberOfDIS_Entities](#)
- [numberOfDIS_Subaggregates](#)
- [numberOfDIS_Subaggregates](#)
- [numberOfDatumRecords](#)
- [numberOfDatumRecords](#) *Get the number of variable datum records*
- [numberOfDatumRecordsOffset](#) *Returns an offset from the beginning of variable length data to the number of variable datum records field*
- [numberOfEntities](#)
- [numberOfEnvironmentalRecords](#)
- [numberOfMines](#)
- [numberOfPdus](#)
- [numberOfSensorTypes](#)
- [numberOfSilentEntities](#)
- [numberOfSilentEntities](#)
- [numberOfSilentSubaggregates](#)
- [numberOfSilentSubaggregates](#)
- [numberOfTripWiresPaddingBytes](#) *Returns the number of padding bytes following number of wires records*
- [numberOfTripWiresPaddingOffset](#) *Returns a buffer offset to the padding following number of wires records*
- [numberOfTripWiresRecordOffset](#) *Returns a buffer offset to the number of wires records*
- [numberOfVerticesPaddingBytes](#) *Returns the number of padding bytes following number of vertices records*
- [numberOfVerticesPaddingOffset](#) *Returns a buffer offset to the padding following number of vertices records*
- [numberOfVerticesRecordOffset](#) *Returns a buffer offset to the number of vertices records*
- [objectAdded](#) *Called after an aggregate is added and before addition callbacks are processed*
- [objectAdded](#) *Called after an entity is added and before addition callbacks are processed*
- [objectAdded](#) *Called after an entity is added and before addition callbacks are processed*
- [objectAdded](#) *Called after a remote object has been added to the list*
- [object](#) *Get index'th object designator*

- [object](#) *Get index'th object designator*
- [objectID](#)
- [objectID](#)
- [objectID](#)
- [objectType](#)
- [objectType](#)
- [objectType](#)
- **offset :**
 - in [DtMimicTrackVCPdu](#)
 - in [DtMimicVCPdu](#)
 - in [DtTetherTrackVCPdu](#)
- [ones](#) *Get a vector set to all ones*
- [openFile](#) *Open the file fedFileName*
- [operational](#)
- **operator = :**
 - in [DtNet32Vector](#)
 - in [DtNet64Vector](#)
 - in [DtNetBurstDescriptor](#)
 - in [DtNetEntityIdentifier](#)
 - in [DtNetEntityType](#)
 - in [DtNetEulerAngles](#)
 - in [DtNetEventId](#)
- **operator = :**
 - in [DtNetFloat32](#)
 - in [DtNetFloat64](#)
 - in [DtNetInt16](#)
 - in [DtNetInt32](#)
- **operator = :**
 - in [DtNetRadioEntityType](#)
 - in [DtNetSimulationAddress](#)
- **operator = :**
 - in [DtNetU16](#)
 - in [DtNetU32](#)
- [operator DtFloat32](#)
- [operator DtFloat64](#)
- [operator DtInt16](#)
- [operator DtInt32](#)
- [operator DtNetBurstDescriptor](#)
- [operator DtNetEntityIdentifier](#)

- [operator DtNetEntityType](#)
- [operator DtNetEventId](#)
- [operator DtNetRadioEntityType](#)
- [operator DtTaitBryan](#)
- [operator DtU16](#)
- [operator DtU32](#) *Cast to a DtU32 representing the whole timestamp Useful mainly for backwards compatibility for applications that used to cast DtNetTimeStamp to DtU32 before accessing it*
- [operator DtU32](#)
- [operator DtVector*](#)
- **operator DtVector :**
 - in [DtNet32Vector](#)
 - in [DtNet64Vector](#)
- [operator const DtVector*](#)
- [operator const char*](#) *cast operator*
- [operator const double*](#)
- [operator double*](#)
- [operator!=](#) *inequality operator*
- [operator!=](#) *equality operator*
- **operator!= :**
 - in [DtEntityIdentifier](#)
 - in [DtEntityType](#) *!= returns 0 if they are equal, nonzero if they are not equal*
- [operator!=](#)
- [operator!=](#)
- [operator!=](#)
- [operator!=](#)
- [operator!=](#)
- [operator!=](#)
- [operator!=](#)
- [operator!=](#)
- [operator!=](#)
- [operator!=](#)
- [operator!=](#)
- [operator+=](#)
- [operator+](#) *concatenation operators*
- [operator<<](#)
- [operator<=](#)
- [operator<](#)
- [operator=](#) *assignment operator*
- [operator==](#) *equality operator*
- [operator==](#) *equality operator*

- **operator== :**
 - in [DtEntityIdentifier](#)
 - in [DtEntityType](#)
- [operator==](#)
- [operator==](#) *equivlance operators*
- [operator==](#) *equality operators*
- [operator==](#) *Comparison operators*
- [operator==](#) *Comparison operators*
- [operator==](#) *comparison operators*
- [operator==](#) *Get whether another tait bryan equals or does not equal this one*
- [operator==](#) *comparison operators*
- [operator==](#) *equivalance operators*
- [operator==](#) *comparison operators*
- [operator==](#)
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#)
- [operator=](#)
- [operator=](#) *Assignment operators*
- [operator=](#) *Assignment operators*

- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#)
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#)
- [operator=](#)
- [operator=](#)
- [operator=](#)
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *assignment operators*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *assignment operator - not really implemented*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#)
- [operator=](#) *Assignment operators*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operators*
- [operator=](#) *Assignment operators*
- [operator=](#) *Assignment operators*

- [operator=](#) *Assignment operator*
- [operator=](#)
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator - Does not replace the DtHashlist's list*
- [operator=](#) *Assignment operator*
- [operator=](#)
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operators*
- [operator=](#) *Assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#)
- [operator=](#) *Assignment operators*
- [operator=](#)
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#)
- [operator=](#) *Assignment operator*
- [operator=](#)
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#)
- [operator=](#) *Assignment operator*
- [operator=](#)
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*

- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#)
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#)
- [operator=](#) *Assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#)
- [operator=](#) *assignment operator*
- [operator=](#) *assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#) *Assignment operator*
- [operator=](#)

- [operator=](#) Assignment operator
- [operator=](#) assignment operator
- [operator=](#) Assignment operator is protected because it is not fully implemented
- [operator=](#) assignment operator
- [operator=](#) Assignment operator is protected because it is not fully implemented
- [operator=](#) assignment operator
- [operator=](#) Assignment operator is protected because it is not fully implemented
- [operator=](#) assignment operator
- [operator=](#) assignment operator
- [operator=](#) assignment operator
- [operator=](#) Assignment operator
- [operator=](#) Assignment operator
- [operator=](#) Assignment operator
- [operator=](#)
- [operator=](#)
- [operator=](#) assignment operator
- [operator=](#) assignment operator
- [operator=](#) assignment operator
- [operator=](#) Assignment operator
- [operator=](#) assignment operator
- [operator=](#) assignment operator
- [operator=](#) Assignment operator
- [operator=](#) assignment operators
- [operator=](#) assignment operator
- [operator=](#) Assignment operator
- [operator=](#) assignment operator
- [operator=](#)
- [operator=](#) assignment operator
- [operator=](#) assignment operator
- [operator=](#) assignment operator
- [operator=](#) assignment operator
- [operator=](#) Assignment operator
- [operator=](#) Assignment operator
- [operator=](#) Assignment operator
- [operator=](#) Assignment operator
- [operator=](#) Assignment operator
- [operator=](#)
- [operator=](#) assignment operator, convert geodetic to UTM

- [packet](#) Older function that returns a pointer to the network representation
- [packetGood](#) Returns whether packet is appropriate DIS version
- [paddingBytes](#) Returns the number of padding bytes which must follow the entity ID records
- [paddingOffset](#) Returns an offset from the beginning of variable length data to the padding following the entity ID records
- [paintSchemePaddingBytes](#) Returns the number of padding bytes following paint scheme records
- [paintSchemePaddingOffset](#) Returns a buffer offset to the padding following paint scheme records
- [paintSchemeRecordOffset](#) Returns a buffer offset to the paint scheme records
- [paramIndex](#)
- [paramNum](#) Returns the index of the parameter in a multiple part list as is found in HLA messages and DIS PDUs
- [parameter1](#)
- [parameter2](#)
- [parameterBit](#)
- [parameterCode](#)
- [parameterData](#)
- [parameterData](#)
- [parameter](#)
- [parameter](#) Returns a pointer to the first parameter record (after static data)
- [params](#) return correctly casted parameters pointer; see *IgrControlP*
- [params](#) return correctly casted parameters pointer; see *IgrControlP*
- [params](#) return correctly casted parameters pointer; see *IgrControlP*
- [params](#) return correctly casted parameters pointer; see *IgrControlP*
- [params](#) return correctly casted parameters pointer; see *IgrControlP*
- [params](#) return correctly casted parameters pointer; see *IgrControlP*
- [params](#) return correctly casted parameters pointer; see *IgrControlP*
- [params](#) return correctly casted parameters pointer; see *IgrControlP*
- [params](#) return correctly casted parameters pointer; see *IgrControlP*
- [params](#) return correctly casted parameters pointer; see *IgrControlP*
- [parmIndex](#)
- [parmRecordsLength](#) Get the parameters length
- [partAttachedTo](#) Get the part this part is attached to
- [partNum](#) Get the part number
- [partNumFromPartType](#) returns 0 if there is no such part, otherwise the part number
- [partNumInBounds](#) Returns whether the part number is within bounds
- [partType](#)
- [passiveActRecordOffset](#) Returns a buffer offset to the given additional passive activity record
- [passiveParameterIndex](#)
- [pduFactory](#)
- [pduKind](#)

- [pduKind](#)
- [pduNumber](#)
- [pduTotal](#)
- [perimeterPointRecordOffset](#) *Returns a buffer offset to a perimeter point record*
- [perimeterPointRecordOffset](#) *Returns a buffer offset to a perimeter point record*
- [phase](#)
- [phi](#)
- [phi](#)
- [phi](#)
- [phi](#)
- [phi](#)
- [phi](#)
- [pingClients](#)
- **pitch :**
 - in [DtCompassVCPdu](#)
 - in [DtGroupVCPdu](#)
 - in [DtOrbitVCPdu](#)
- **pitchOffset :**
 - in [DtCompassVCPdu](#)
 - in [DtGroupVCPdu](#)
 - in [DtOrbitVCPdu](#)
- [position](#)
- [position](#)
- [positionErrorOffset](#)
- [power](#)
- [power](#)
- [power](#)
- [power](#)
- [powerPlantOn](#)
- [powerSetting](#)
- **prev :**
 - in [DtListItem](#) *Inline Functions*
 - in [DtListedObj](#)
- [prev](#)
- [prev](#) *Get the previous remote emitterSystem*
- [prev](#)
- [prevObj](#) *Get the previous remote object*
- [primaryAmmunition](#)

- [illegible]

- [printData](#)
- [printData](#)
- [printData](#) *Print the state data*
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#)
- [printData](#) *Print the data contained by this PDU*
- [printData](#)
- [printData](#)
- [printData](#)
- [printData](#) *Print the data contained by this PDU*
- [printData](#)
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *print contents*
- [printData](#) *diagnostic output*
- [printData](#) *diagnostic output*
- [printData](#) *Print the data contained by this PDU*
- [printData](#)
- **printData :**
 - in [DtMimicTrackVCPdu](#)
 - in [DtMimicVCPdu](#)
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#)
- [printData](#)
- [printData](#)
- [printData](#) *Print the PDU's data (not including header)*

- [printData](#) *Implements DtPdu's virtual function to print the PDU's data*
- [printData](#) *Print the data contained by this PDU*
- [printData](#)
- [printData](#)
- [printData](#)
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#)
- [printData](#) *Print the data contained by this PDU*
- **printData :**
 - in [DtRepairCompletePdu](#)
 - in [DtRepairResponsePdu](#)
- **printData :**
 - in [DtResupplyCancelPdu](#)
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [printData](#)
- [printData](#)
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- [printData](#)
- [printData](#)
- [printData](#)
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print state data*
- [printData](#)
- [printData](#) *Print the data contained by this PDU*
- [printData](#) *Print the data contained by this PDU*
- **printData :**
 - in [DtTetherTrackVCPdu](#)
 - in [DtTetherVCPdu](#)
- [printData](#)
- [printData](#)
- [printData](#)
- [printData](#) *Print the data contained by this PDU*
- [printData](#)
- [printData](#) *Print the data contained by this PDU*
- [printData](#)

- [illegible]

- [print](#) *Print the information contained in the record*
- [print](#) *Print the information contained in the record*
- [print](#) *Print the information contained in the record*
- [print](#) *Print the information contained in the record*
- [print](#) *Print the information contained in the record*
- [print](#) *Print the contents of the PDU by calling printHeader and printData*
- [print](#) *Print the information contained in the record*
- [print](#) *Print this record*
- [print](#) *Print the SRs of all objects*
- [print](#) *Print the SRs of all objects*
- [print](#) *Print the SRs of all objects*
- [print](#) *Print the SRs of all objects*
- [print](#) *Print the record*
- [print](#) *Print the record*
- [print](#) *Print data*
- [print](#) *Print the information contained in the record*
- [print](#) *Print the information contained in the record*
- [print](#) *Print the information contained in the record*
- [print](#) *Print the information contained in the record*
- [print](#) *Print the contents of the record*
- [print](#) *Print the information contained in the record*
- [printHeader](#) *Print the header (override to include the new sub-protocol field)*
- [printHeader](#) *Print the header (override to include the new sub-protocol field)*
- [printHeader](#) *Print the header (override to include the new sub-protocol field)*
- [printHeader](#) *Print the header (override to include the new sub-protocol field)*
- [printHeader](#) *Print the header (override to include the new sub-protocol field)*
- [printHeader](#) *Print the PDU's header*
- [printModulationParams](#)
- [printSystem](#)
- [printSystem](#)
- [printTrackJam](#) *Prints track/jam target information*
- [processAggregateState](#) *Process an entity state PDU*
- [processAggregateState](#) *process an incoming aggregate state pdu*
- [processConnections](#)
- [processControlMsg](#)
- [processEntityState](#) *Process an entity state PDU*
- [processEntityState](#) *process an incoming entity state pdu*
- [processObjectAdditionCallbacks](#) *Process object addition/removal callbacks*

- [processObjectRemovalCallbacks](#)
- [processPdu](#) *Called by readAndProcess (which is called by drainInput)*
- [processPdu](#) *Call the callbacks registered for the PDU's kind*
- [processPostUpdateCallbacks](#) *process the post update callbacks*
- [processStateMessage](#) *Process an aggregate state PDU Implementation for abstract base class*
- [processStateMessage](#) *process an incoming state message*
- [processStateMessage](#) *Process an electronic emissions PDU Implementation for abstract base class*
- [processStateMessage](#) *process an incoming state message*
- [processStateMessage](#) *Process an entity state PDU*
- [processStateMessage](#) *process an incoming state message*
- [processStateMessage](#) *Process an entity state PDU*
- [processStateMessage](#) *process an incoming state message*
- [propulsionPlantConfig](#)
- [propulsionSystemRecordOffset](#) *Returns a buffer offset to the propulsion system data record*
- [protocolFamily](#)
- [protocolVersion](#)
- [protocolVersion](#)
- [protocolVersion](#)
- [provideAttributeValueUpdate](#)
- [psi](#)
- [psi](#)
- [psi](#)
- [psi](#)
- [psi](#)
- [psi](#)
- [psi](#)
- [psuedoNoiseSpectrumInUse](#)
- [pulseWidth](#)
- [pulseWidth](#)
- [purgeBeamList](#)
- [quantity](#)
- [quantity](#)
- [radarSigRepIndex](#)
- [radialOffset](#)
- [radioEntityType](#)
- [radioEntityType](#)
- [radiold](#)
- [radiold](#)

- [radiold](#)
- [radiold](#)
- [radiold](#)
- **radius :**
 - in [DtCompassVCPdu](#)
 - in [DtOrbitVCPdu](#)
- [rampDeployed](#)
- [range](#)
- [range](#)
- [rangeOffset](#)
- [rate](#)
- [rate](#)
- [readAndForward](#)
- [readAndProcess](#)
- [readFd](#)
- [readFd](#)
- [readFd](#) *not all DtSocket have associated file descriptors those that do not will have readFd()/writeFd() return -1*
- [readFd](#)
- [readFd](#)
- [readFed](#) *Read the FED file fedFileName, filling in intMgr and objMgr*
- [readInterClass](#) *Read the description of a single interaction class, filling in interMgr with data read*
- [readInteractions](#) *Starting from the line after "(interactions", read through the ")" that denotes the end of the interactions definitions, filling in interMgr as we go*
- [readLine](#) *Read a line from the file*
- [readObjClass](#) *Read the description of a single object class, filling in objMgr with data read*
- [readObjects](#) *Starting from the line after "(objects", read through the ")" that denotes the end of the objects definitions, filling in objMgr as we go*
- [readUntil](#) *Repeatedly call readAndProcess until predicate holds on the PDU that is being read, or until timeout seconds have elapsed*
- [realWorldTime](#)
- [realWorldTime](#)
- [realWorldTime](#)
- [realWorldTime](#)
- [reason](#)
- [reason](#)
- [receiveInteraction](#)
- [receivedPower](#)
- [receivedPower](#)
- [receiverId](#)
- [receiverId](#)

- [receiverState](#)
- [receiverState](#)
- [receivingEntityID](#)
- [receivingEntityID](#)
- [receivingEntityID](#)
- [receivingEntityID](#)
- [receivingEntityID](#)
- [receivingID](#)
- [receivingID](#)
- [receivingID](#)
- **receivingId :**
 - in [DtRepairCompletePdu](#)
 - in [DtRepairResponsePdu](#)
- **receivingId :**
 - in [DtResupplyCancelPdu](#)
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [recordBitLength](#)
- [recordByteLength](#)
- [recordByteSize](#) *Compute the total record size in bytes from the datum bit length*
- [recordCount](#)
- [recordLength](#) *Get the grid axis record length from a network structure*
- [recordLength](#) *Get the total number of bytes required for the network structure*
- [recordLength](#) *Get the total number of bytes required for the network structure*
- [recordLength](#) *Get the grid data record representation type from a network structure*
- [recordLength](#) *Get the grid data record representation type from a network structure*
- [recordLength](#) *Get the grid data record representation type from a network structure*
- [recordLength](#) *Get the record length*
- [recordOffset](#) *Returns an offset in bytes from the beginning of the variable length data region to the beginning of a particular record*
- [recordOffset](#) *Returns the offset (in bytes) of the given parameter record*
- [recordPointer](#) *Get a pointer to an environmental record*
- [recordSetLength](#) *Get the total number of bytes required for the network representation of the record set*
- [recordSetOffset](#) *Returns a buffer offset to the given record set*
- [recordSize](#) *Returns the record size for the grouped entity category*
- [recordType](#) *Get the grid data record representation type from a network structure*
- [recordType](#) *Get the record type*
- [recvFromNet](#)
- [recvWithStatus](#)

- [illegible]

- [illegible]

- [removeCallback](#)
- **removeCallback :**
 - in [DtRepairCompletePdu](#)
 - in [DtRepairResponsePdu](#)
- **removeCallback :**
 - in [DtResupplyCancelPdu](#)
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [removeCallback](#)
- [remove](#)
- [remove](#) *Remove the current association involving key from the list*
- [remove](#) *Remove something from the list*
- [remove](#) *Remove an element from the list*
- [remove](#) *Remove the current association involving key from the list*
- [removeEmitterAdditionCallback](#)
- [removeEmitterRemovalCallback](#)
- [removeEntityAdditionCallback](#)
- [removeEntityRemovalCallback](#)
- **removeFromList :**
 - in [DtIntrusiveList](#) *Override of virtual function*
 - in [DtList](#) *Remove item from the list*
- [removeObjectAdditionCallback](#)
- [removeObjectInstance](#)
- [removeObjectRemovalCallback](#)
- [removePduCallback](#)

- [removePduCallback](#)
- [removePostDrainCallback](#)
- [removePostUpdateCallback](#)
- [removePostUpdateCallback](#)
- [removePostUpdateCallback](#)
- [removeRemoteSite](#)
- [repairResult](#)
- [repairType](#)
- **repairingId :**
 - in [DtRepairCompletePdu](#)
 - in [DtRepairResponsePdu](#)
- [replaceSlashes](#) *Replaces slashes*
- [reqRelService](#)
- [reqRelService](#)
- [reqRelService](#)
- [reqRelService](#)
- [reqRelService](#)
- [reqRelService](#)
- [reqRelService](#)
- [reqRelService](#)
- [reqRelService](#)
- [reqRelService](#)
- [reqRelService](#)
- [reqRelService](#)
- [reqRelService](#)
- [requestAttributeOwnershipAssumption](#) //////////////////////////////////
- [requestAttributeOwnershipRelease](#)
- [requestFederationRestoreFailed](#)
- [requestFederationRestoreSucceeded](#)
- [requestID](#)
- [requestID](#)
- [requestID](#)
- [requestID](#)
- [requestID](#)
- [requestID](#)
- [requestID](#)
- [requestID](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)

- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestId](#)
- [requestRetraction](#)
- [requestServerCStates](#)
- [requestServerState](#)
- [requestStatus](#)
- [requestStatus](#)
- [requestedMineType](#)
- [requesterID](#)
- [requesterID](#)
- [requesterID](#)
- [requestingEntityID](#)
- [requestingEntityID](#)
- [requestingEntityID](#)
- [requestingId](#)
- [resizeGridAxisRecord](#) *Resizes a grid axis record*
- [resizeGridDataRecord](#) *Resizes a grid data record*
- [resizeRecord](#) *Resize a given parameter record to the size indicated*
- [responseFlag](#)
- [responseFlag](#)
- [responseSerialNumber](#)
- [restStatus](#)
- [result](#)
- [resultOffset](#)
- **roll :**

- in [DtCompassVCPdu](#)
- in [DtOrbitVCPdu](#)
- [rotationError](#)
- [rotationThreshold](#)
- [rotationThreshold](#)
- [rotationTraceThreshold](#) *Returns the "RotationTrace" threshold*
- [rotationTraceThreshold](#) *Returns the "RotationTrace" threshold*
- [rotationalVelocity](#)
- [rotationalVelocity](#)
- [rotationalVelocity](#)
- [rotationalVelocity](#)
- [rotationalVelocity](#)
- [rotationalVelocity](#)
- [rotationalVelocity](#)
- [rotationalVelocity](#)
- [sameEntity](#)
- [sameHost](#)
- [sampleRate](#)
- [sampleRate](#)
- [sampleTime](#)
- [sampleType](#)
- [scalarDetectionCoefficientRecordOffset](#) *Returns a buffer offset to the scalar detection coefficient records*
- [scanPattern](#)
- [secondaryAmmunition](#)
- [secondaryAmmunition](#)
- [secondaryAmmunition](#)
- [secondaryAmmunition](#)
- [secondary](#)
- [secondaryOperational](#)
- [secondaryWeaponState](#)
- [segmentDepth](#)
- [segmentHeight](#)
- [segmentLength](#)
- [segmentNumber](#)
- [segmentWidth](#)
- [select](#)
- [self](#) *Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type*
- [self](#) *Cast off const-ness for trivial internal change*
- [self](#) *Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on*

return type

- [illegible]

- [self](#) Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
- [self](#) Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- [self](#) Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- [self](#) Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- [self](#) Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- [self](#)
- [self](#) Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- [self](#) Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- [self](#) Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- [self](#) Cast off const-ness for trivial internal change
- [send](#) Send a state message to the network
- [send](#) Called by tick() to send a PDU
- [send](#) Send a PDU to the specified address, or the DtExerciseConn's default destination address if addr is DtUSE_DEFAULT_IP_ADDR
- [sendFinalPdu](#) not virtual since called from destructor
- [sendFinalPduOnDestruction](#)
- [sendNeeded](#) determine if a state message needs to be sent
- [sendNeeded](#) Returns true if a PDU needs to be sent, (based on thresholds, etc) false otherwise
- [sendStamped](#) Send an interaction to the network which has been time stamped with the current time
- [sendStamped](#) Send a stamped message - call down to sendStamped(DtPdu &pdu
- [sendToAll](#)
- [sendToAllRemoteSites](#) Send to all remote sites except dwSkipAddr
- [sendTo](#)
- [sendTo](#)
- [sendTo](#)
- [sendTo](#)
- [sendTo](#)
- [senderId](#)
- [senderId](#)
- [senderId](#)
- [sensorType](#)
- [sensorTypePaddingBytes](#) Returns the number of padding bytes following sensor type information
- [sensorTypePaddingOffset](#) Returns a buffer offset to the padding following sensor type information
- [sensorTypeRecordOffset](#) Returns a buffer offset to the sensor type records

- [sensorTypeRecordOffset](#) *Returns a buffer offset to a sensor type record*
- [separator](#) *returns the separator used in representing directories*
- [sequenceNumber](#)
- [sequenceNumber](#)
- [serialNumber](#)
- [serviceType](#)
- [servicingId](#)
- [setAPA](#) *Set/Get the APA value*
- [setAcceleration](#) *Set/Get absolute acceleration*
- [setAcceleration](#) *Set/Get the acceleration*
- [setAcceleration](#) *Set/Get the acceleration*
- [setAcceleration](#) *Set/Get absolute acceleration*
- [setAcceleration](#)
- [setAcceleration](#) *Set/Get absolute acceleration*
- [setAcceleration](#) *Set/Get the acceleration*
- [setAcknowledgeFlag](#)
- [setAcknowledgeFlag](#) *Set/Get the acknowledge flag*
- [setAcousticID](#) *Set/Get the acoustic ID number*
- [setAcousticName](#) *Set/Get the system name*
- [setAcousticSigRepIndex](#) *Set/Get acoustic signature representation index*
- [setAction](#) *mutator functions*
- [setAction](#) *set/get action to take place when logger switched to playback mode*
- [setAction](#) *set/get action to take when switching to record-mode*
- [setActionId](#)
- [setActionId](#) *Set/Get the action id*
- [setActiveEmissionParmIndex](#) *Set/Get the active emission parameter index*
- [setAddtnlPassiveAct](#) *Set/Get an additional passive activity data record*
- [setAfterburnerOn](#) *Set/Get whether entity has after burner on or not*
- [setAggDimensionThreshold](#) *Set/Get aggregate dimension threshold*
- [setAggregateId](#) *Set/Get aggregate entity identifier*
- [setAggregateId](#) *Set/Get the aggregate's identifier*
- [setAggregateState](#) *Set/Get the aggregate state*
- [setAggregateState](#) *Set/Get the aggregate state*
- [setAggregateType](#) *Set/Get the aggregate type*
- [setAggregateType](#) *Set/Get the aggregate type*
- [setAggregateType](#) *Set/Get the aggregate type*
- [setAirMaintenanceStatus](#) *Set/Get the air maintenance status*
- [setAirMaintenanceStatus](#) *Set/Get the air maintenance status*

- [setAlgorithm](#) *Set/Get the dead reckoning algorithm*
- [setAlgorithm](#)
- [setAlgorithm](#)
- [setAlt](#)
- [setAltParameter4](#) *Set/Get alternate parameter 4*
- [setAlternativeEntityType](#) *Set/Get alternative entity type*
- [setAngles](#) *Set/Get the beam direction*
- [setAngularVelocity](#) *Set/Get entity angular velocity*
- [setAntennaParamsLength](#)
- [setAntennaPattern](#) *Set/Get the antenna pattern*
- [setAntennaPatternParams](#)
- [setAntennaPatternType](#) *Set/Get the antenna pattern type*
- [setAntennaPatternType](#)
- [setAntiHandlingDevice](#) *Set/Get anti-handling device*
- [setAppearanceAttribute](#) *Set/Get appearance by attribute for the lower 16 bits*
- [setAppearanceAttribute](#) *Set/Get appearance by attribute*
- [setAppearanceAttribute](#) *Set/Get appearance by attribute*
- [setAppearanceAttribute](#) *Set/Get appearance by attribute*
- [setAppearanceAttribute](#) *Set/Get appearance by attribute*
- [setAppearanceAttribute](#)
- [setAppearanceAttribute](#) *Set/Get an appearance attribute*
- [setAppearanceAttribute](#) *Set/Get appearance by attribute*
- [setAppearanceAttribute](#) *Set/Get appearance by attribute*
- [setAppearanceAttribute](#) *Set/Get appearance by attribute, the lower 16 bits*
- [setAppearanceAttribute](#) *Set/Get appearance by attribute*
- [setAppearanceAttribute](#) *Set/Get appearance by attribute for the lower 16 bits*
- [setAppearanceAttribute](#) *Set/Get appearance by attribute*
- [setAppearance](#) *Set/Get appearance, the lower 16 bits (as an integer)*
- [setAppearance](#) *Set/Get appearance (as a 32 bit integer)*
- [setAppearance](#) *Set/Get appearance (as a 32 bit integer)*
- [setAppearance](#) *Set/Get appearance (as a 32 bit integer)*
- [setAppearance](#) *Set/Get appearance (as a 32 bit integer)*
- [setAppearance](#)
- [setAppearance](#)
- [setAppearance](#) *Set/Get appearance (as a 32 bit integer)*
- [setAppearance](#) *Set/Get appearance (as a 32 bit integer)*
- [setAppearance](#) *Set/Get appearance, the lower 16 bits (as an integer)*
- [setAppearance](#) *Set/Get appearance (as a 16 bit integer)*

- [setAppearance](#) Set/Get appearance, the lower 16 bits (as an integer)
- [setAppearance](#) Set/Get the appearance of an entity (as a 32 bit integer)
- [setAppearanceField1](#) Set appearance field 1
- [setAppearanceField1Flag](#) Set a flag of appearance field 1
- [setAppearanceField2](#) Set appearance field 2
- [setAppearanceField2Flag](#) Set a flag of appearance field 2
- [setApplicableModes](#) Set/Get the applicable modes
- [setApplication](#) Set/Get application
- [setApproximating](#) Set/Get whether or not to do approximating
- [setApproximator](#) Get and set the approximator used to provide approximated position and orientation information for the entity
- [setArtParamVal](#) Set/Get the specified part type's specified parameter type value
- [setArtParamVal](#)
- [setArtParamVal](#)
- [setArtPartParameters](#) Set/Get articulated parts parameters
- [setArtPartParameters](#) Set/Get articulated parts parameters
- [setAttPartParameters](#) Set/Get attached parts parameters
- [setAttPartParameters](#) Set/Get attached parts parameters
- **setAttachId :**
 - in [DtCompassVCPdu](#)
 - in [DtMimicTrackVCPdu](#)
 - in [DtMimicVCPdu](#)
 - in [DtOrbitVCPdu](#)
 - in [DtTetherTrackVCPdu](#)
 - in [DtTetherVCPdu](#)
- [setAttachedPartChange](#)
- [setAttachedPartType](#)
- [setAttackerId](#)
- [setAttackerId](#)
- [setAudioAppearance](#) Set/Get audio appearance (as a 32 bit integer)
- [setAzimuthCenter](#) Set/Get Azimuth Center
- [setAzimuthCenter](#) Set/Get the Azimuth Center
- [setAzimuthError](#) Set/Get azimuth error
- [setAzimuthRate](#)
- [setAzimuthSweep](#) Set/Get Azimuth Sweep
- [setAzimuthSweep](#) Set/Get the Azimuth Sweep
- [setAzimuthalBeamwidth](#) Set/Get the azimuthal beamwidth
- [setBandwidth](#)
- [setBasicFixedWingAircraftRecord](#) Set/Get a BasicFixedWingAircraftRecord

- [setBasicGroundCombatSoldierRecord](#) Set/Get a BasicGroundCombatSoldierRecord
- [setBasicGroundCombatVehicleRecord](#) Set/Get a BasicGroundCombatVehicleRecord
- [setBasicRotorWingAircraftRecord](#) Set/Get a BasicRotorWingAircraftRecord
- [setBeamCenterAzimuth](#) Set/Get the beam center azimuth
- [setBeamCenterD_E](#) Set/Get the beam center D/E
- [setBeamData](#) Set/Get beam data
- [setBeamDataLength](#) Set/Get the beam data length
- [setBeam](#) Set/Get beam fundamental data parameters for a beam
- [setBeamFunction](#)
- [setBeamFunction](#)
- [setBeamFunction](#) Set/Get Beam Function
- [setBeamID_Number](#) Set/Get the beam ID number
- [setBeamId](#)
- [setBeamId](#)
- [setBeamId](#) Set/Get Beam Id
- [setBeamId](#) set/get the beam Id
- [setBeamParameterIndex](#)
- [setBeamParameterIndex](#)
- [setBeamParameterIndex](#) Set/Get Beam Parameter Index
- [setBeamType](#) Set/Get beam type - track, jam or neither
- [setBeamwidth](#) Set/Get the azimuth bandwidth and elevation
- [setBodyHeading](#)
- [setBodyOrientation](#)
- [setBundling](#)
- [setBurst](#) NOT USED IN HLA
- [setBurst](#) NOT USED IN HLA
- [setBurstFuse](#) Set/Get burst fuse (optional)
- [setBurstFuse](#) Set/Get burst fuse (optional)
- [setBurstLength](#) Set/Get the burst length
- [setBurstMunitionType](#) Set/Get burst munition type
- [setBurstMunitionType](#) Set/Get burst munition type
- [setBurstQuantity](#) Set/Get burst quantity (optional)
- [setBurstQuantity](#) Set/Get burst quantity (optional)
- [setBurstRate](#) Set/Get burst rate (optional)
- [setBurstRate](#) Set/Get burst rate (optional)
- [setBurstWarhead](#) Set/Get burst warhead (optional)
- [setBurstWarhead](#) Set/Get burst warhead (optional)
- **setByInverse :**
 - in [DtCoordTransform](#)

- in [DtCoordTranslation](#)

- [setCallbackFunction](#) *set/get the callback function*
- [setCamouflageType](#) *Set/Get camouflage type*
- [setCapabilities](#)
- [setCapabilities](#) *Set/Get the capabilities field*
- [setCapabilities](#) *Set/Get capabilities (as a 32-bit integer)*
- [setCapability](#) *Set/Get a particular capability (as a Boolean)*
- [setCategory](#)
- [setCategory](#) *Set/Get the grouped entity category (setCategory will also set the number of entities to zero)*
- [setCategory](#) *Set/Get the category*
- [setChangeOptionsBit](#) *Set/Get the system change/option by bit value*
- [setChangeOptions](#) *Set/Get the system change/options*
- [setClassOfEnv](#) *Set/Get the environmental class*
- [setCmdType](#) *Set/Get intercom command type*
- [setCodeName](#)
- [setCodeName](#) *Set/Get the code name*
- [setCodeName](#)
- [setCoefficientOfRestitution](#) *Set/Get coefficient of restitution*
- [setCollideld](#)
- [setCollidingEntityID](#) *Set/Get colliding entity identifier*
- [setCollisionInterResultXX](#) *Set/Get collision intermediate result XX*
- [setCollisionInterResultXY](#) *Set/Get collision intermediate result XY*
- [setCollisionInterResultXZ](#) *Set/Get collision intermediate result XZ*
- [setCollisionInterResultYY](#) *Set/Get collision intermediate result YY*
- [setCollisionInterResultYZ](#) *Set/Get collision intermediate result YZ*
- [setCollisionInterResultZZ](#) *Set/Get collision intermediate result ZZ*
- [setCollisionType](#)
- [setCommClass](#) *Set/Get intercom communications channel class*
- [setCommType](#) *Set/Get intercom communications channel type*
- [setComment](#) *none besides those inherited from DtPduWithData*
- [setComment](#) *Set the comment*
- [setCommentInternal](#) *Set the comment*
- [setConstantGrid](#) *Set/Get constant grid indicator*
- [setContactVelocity](#) *Set/Get contact velocity*
- [setControlType](#) *Set/Get intercom control type*
- [setCoordSystem](#) *Set/Get coordinate system*
- [setCoordinateOffsetXi](#) *Set/Get the coordinate offset*
- [setCoordinateScaleXi](#) *Set/Get the coordinate scale*

- [setCountry](#)
- [setCountry](#) *Get/Get country*
- [setCryptoKeyId](#) *Set/Get the crypto key id*
- [setCryptoKeyId](#)
- [setCryptoMode](#) *Set/Get the crypto key id*
- [setCryptoSystem](#) *Set/Get the crypto system*
- [setCryptoSystem](#)
- [setCurrentShaftRPM](#) *Set/Get the current shaft RPM*
- [setDRAlgorithm](#) *Set/Get the dead reckoning algorithm*
- [setD_E_Beamwidth](#) *Set/Get the D/E beamwidth*
- [setDamageState](#) *Set/Get the damage state*
- [setDataByte](#) *Set/Get a data byte*
- [setDataByte](#) *Set/get a data byte*
- [setDataByte](#) *Set/Get a data byte*
- [setDataByte](#)
- [setDataBytes](#) *Set/Get multiple data bytes*
- [setData](#)
- [setData](#) *Set/Get multiple data bytes*
- [setData](#) *Set/get multiple data bytes*
- [setData](#)
- [setDataFilterBit](#) *Set/Get a data filter bit (this call erases any data which was set in the variable length fields)*
- [setDataFilter](#) *Set/Get the data filter as a 32-bit value (this call erases any data which was set in the variable length fields)*
- [setDataFilter](#) *Set/Get the data filter (as a 32-bit integer)*
- [setDataFilterField](#) *Set/Get data filter attributes (DtTrue indicates that the data for that field is requested, DtFalse indicates that the data for that field is not requested)*
- [setDataFloat32](#) *Set/Get data in 32-bit floating point format*
- [setDataFloat64](#) *Set/Get data in 64-bit floating point format*
- [setDataInt16](#) *Set/Get data in signed 16-bit integer format*
- [setDataInt32](#) *Set/Get data in signed 32-bit integer format*
- [setDataLength](#) *Set/Get the data length*
- [setDataU16](#) *Set/Get data in unsigned 16-bit integer format*
- [setDataU32](#) *Set/Get data in unsigned 32-bit integer format*
- [setDataValue](#) *Set/Get a data value*
- [setDataValue](#) *Set/Get a data value*
- [setDatumBitLength](#) *Set/Get the datum length (in bits)*
- [setDatumBitLength](#) *Set/Get the datum length in bits*
- [setDatumID](#) *Set/Get the variable datum ID*
- [setDatumID](#) *Set/Get the datum ID*

- [setDatumID](#) Set/Get the variable datum ID
- [setDatumParam](#) Set the datum ID in the index'th fixed or variable datum record to param, depending on whether fixedOrVar is DtFixed or DtVar
- [setDatumParam](#)
- [setDatumParam](#) Set the datum ID in the index'th fixed or variable datum record to param, depending on whether fixedOrVar is DtFIXED or DtVAR
- [setDatumValByteArray](#) Set/Get datum value as a byte array
- [setDatumValByteArray](#)
- [setDatumValByteArray](#)
- [setDatumValByteArray](#)
- [setDatumValByteArray](#) Set/Get datum value as a byte array
- [setDatumValFloat32](#) Set/Get datum value as a DtFloat32
- [setDatumValFloat32](#)
- [setDatumValFloat32](#)
- [setDatumValFloat32](#)
- [setDatumValFloat32](#) Set/Get datum value as a DtFloat32 Set will set the datum bit length to 32
- [setDatumValFloat64](#) Set/Get datum value as a DtFloat64
- [setDatumValFloat64](#) ***** SETTING DATA VALUES BY PARAM ID ***** Use one of the following to set the data value in a fixed or variable datum record, depending on the data type of your data
- [setDatumValFloat64](#)
- [setDatumValFloat64](#) ***** SETTING DATA VALUES BY PARAM ID ***** Use one of the following to set the data value in a fixed or variable datum record, depending on the data type of your data
- [setDatumValFloat64](#) Set/Get datum value as a DtFloat64 Set will set the datum bit length to 64
- [setDatumValInt16](#) Set/Get datum value as a DtInt16
- [setDatumValInt16](#) Set/Get datum value as a DtInt16 Set will set the datum bit length to 16
- [setDatumValInt32](#) Set/Get datum value as a DtInt32
- [setDatumValInt32](#)
- [setDatumValInt32](#)
- [setDatumValInt32](#)
- [setDatumValInt32](#) Set/Get datum value as a DtInt32 Set will set the datum bit length to 32
- [setDatumValU16](#) Set/Get datum value as a DtU16
- [setDatumValU16](#) Set/Get datum value as a DtU16 Set will set the datum bit length to 16
- [setDatumValU32](#) Set/Get datum value as a DtU32
- [setDatumValU32](#) Set/Get datum value as a DtU32 Set will set the datum bit length to 32
- [setDatumValU8](#) Set/Get datum value as a DtU8
- [setDatumValU8](#) Set/Get datum value as a DtU8 Set will set the datum bit length to 8
- [setDatumValUnsigned32](#)
- [setDatumValUnsigned32](#)
- [setDatumValUnsigned32](#)
- [setDeadReckoningAlgorithm](#)

- [setDeadReckoningAlgorithm](#) Set/Get dead reckoning algorithm
- [setDeadReckoningParameters](#) Set/Get dead reckoning parameters
- [setDeltaX](#) Set/Get delta X
- [setDeltaY](#) Set/Get delta Y
- [setDeltaZ](#) Set/Get delta Z
- [setDensity](#) Set/Get light state of entity
- [setDesignatedEntityId](#)
- [setDesignatedObject](#) Set/Get the designated object name
- [setDesignatingEntityId](#)
- [setDesignatorCode](#)
- [setDesignatorCode](#) Set/Get the designator code
- [setDesignatorId](#) Set/Get the designator object id
- [setDesignatorPower](#) Set/Get the designator power
- [setDesignatorSpotLocation](#) Set/Get the designator spot location
- [setDesignatorSpotRelative](#)
- [setDesignatorSpotWorld](#)
- [setDesignatorWavelength](#) Set/Get the designator wavelength
- [setDestDeviceID](#) Set/Get the destination device ID
- [setDestEntityID](#) Set/Get the destination entity identifier
- [setDestEntityID](#) Set/Get the destination entity identifier
- [setDestEquipID](#) Set/Get the destination equipment ID
- [setDestGroupBitField](#) Set/Get the group bit field
- [setDestLineID](#) Set/Get the destination line ID
- [setDestLineID](#) Set/Get the destination line ID
- [setDestLineStateCmd](#) Set/Get the destination line state command
- [setDestLineStateCmd](#) Set/Get the destination line state command
- [setDestPriority](#) Set/Get the destination priority
- [setDestPriority](#) Set/Get the destination priority
- [setDetail](#)
- [setDetailedModulationType](#) Set/Get the detailed modulation
- [setDeton2Flag](#) Set second detonation flag
- [setDeton2FlagsField](#) Set second detonation flags field
- [setDetonFlag](#) Set first detonation flag
- [setDetonFlagsField](#) Set first detonation flags field
- [setDetonationResult](#) Set/Get detonation result
- [setDeviceID](#) Set/get intercom ID
- [setDfltAggDimensionThreshold](#)
- [setDfltFractionalThreshold](#) Set/Get fractional threshold value

- [setDfltRotationThreshold](#) Set/Get rotational threshold value
- [setDfltRotationThreshold](#) These four functions are obsolete, but remain for backwards compatibility
- [setDfltRotationThreshold](#) Set/Get default rotation threshold in radians
- **setDfltSmoothPeriod :**
 - in [DtSmoothOrient](#) Set/Get default smooth period
 - in [DtSmoothPos](#) Set/Get default smooth period
- [setDfltSmoothPeriod](#) Set/Get the default smoothing period
- [setDfltThreshMode](#) Set/Get the threshold mode
- [setDfltTimeThreshold](#) Set/Get default time threshold
- [setDfltTimeThreshold](#)
- [setDfltTranslationThreshold](#) Set/Get translational threshold value
- [setDfltTranslationThreshold](#)
- [setDfltTranslationThreshold](#)
- [setDimensions](#) Set/Get the dimensions
- [setDimensions](#) Set/Get the dimensions
- [setDomain](#)
- [setDomain](#) Set/Get the domain
- [setDomainFinalXi](#) Set/Get the domain final Xi
- [setDomainInitialXi](#) Set/Get the domain initial Xi
- [setDomainNumPointsXi](#) Set/Get the domain number of points
- [setEM_Appearance](#) Set/Get EM appearance (as a 32 bit integer)
- [setERP](#) Set/Get ERP
- [setERP](#) Set/Get the ERP
- [setEast](#) inline functions
- [setElectricField](#) Set electric field
- [setElevationCenter](#) Set/Get Elevation Center
- [setElevationCenter](#) Set/Get the Elevation Center
- [setElevationError](#) Set/Get elevation error
- [setElevationSweep](#) Set/Get Elevation Sweep
- [setElevationSweep](#) Set/Get the Elevation Sweep
- [setEmitterFunction](#) Set/Get the emitter function
- [setEmitterId](#)
- [setEmitterId](#)
- [setEmitterId](#) set/get the emitter Id
- [setEmitterName](#) Set/Get the emitter name
- [setEmitterNumber](#) Set/Get the emitter number
- [setEmitterSystem](#) Set/Get an underwater acoustic emitter system
- [setEmittingEntityID](#) Set/Get emitting entity identifier
- [setEmittingEntityID](#) Set/Get emitting entity identifier

- [setEmittingEntityId](#)
- [setEmittingEntityId](#)
- [setEmittingSystemId](#) *Set/Get Emitting System Id*
- [setEncodingClass](#) *Set/get intercom encoding class*
- [setEncodingClass](#)
- [setEncodingType](#) *Set/get intercom encoding type*
- [setEncodingType](#)
- [setEngineRPM](#) *Set/Get the engine RPM*
- [setEngineSmokeOn](#) *Set/Get whether engine smoke is on or not*
- [setEnhancedFixedWingAircraftRecord](#) *Set/Get an EnhancedFixedWingAircraftRecord*
- [setEnhancedGroundCombatSoldierRecord](#) *Set/Get an EnhancedGroundCombatSoldierRecord*
- [setEnhancedGroundCombatVehicleRecord](#) *Set/Get an EnhancedGroundCombatVehicleRecord*
- [setEnhancedRotorWingAircraftRecord](#) *Set/Get an EnhancedRotorWingAircraftRecord*
- [setEntityCoords](#)
- [setEntity](#) *Set/Get entity*
- [setEntityID](#) *Set/Get a DIS entity ID*
- [setEntityID](#) *Set/Get a DIS entity ID*
- [setEntityID](#) *Set/Get the entity ID*
- [setEntityID](#) *Set/Get the entity ID*
- [setEntityID](#) *Set/Get the entity ID*
- [setEntityID](#) *Set/Get the entity ID*
- [setEntityID](#) *Set/Get the entity ID*
- [setEntityID](#) *Set/Get the entity ID*
- [setEntityID](#) *Set/Get the entity ID*
- [setEntityID](#) *Set/Get the entity ID*
- [setEntityID](#) *Set/Get the entity ID*
- [setEntityID](#) *Set/Get the entity's identifier*
- [setEntityId](#) *Set/Get the entity identifier*
- [setEntityId](#) *Set/Get Entity Id - not used after RPR 0*
- [setEntityId](#)
- [setEntityId](#)
- [setEntityId](#)
- [setEntityLocation](#)
- [setEntityNum](#)
- [setEntitySpeed](#) *Set/Get the entity speed*
- [setEntitySpeed](#) *Set/Get the entity speed*
- [setEntitySpeed](#) *Set/Get the entity speed*
- **setEntityToAdd :**
 - in [DtGroupVCPdu](#)

- in [DtModelessVCPdu](#)
- **setEntityToRemove :**
 - in [DtGroupVCPdu](#)
 - in [DtModelessVCPdu](#)
- [setEntityTypeAtStation](#) *Set/Get the entity type for a part*
- [setEntityType](#) *Set/Get the entity type for a part*
- [setEntityType](#) *Set/Get the entity type*
- [setEntityType](#)
- [setEntityType](#) *Set/Get Part Entity Type*
- [setEntityType](#) *Set/Get entity type*
- [setEntityType](#) *Set/Get the entity type*
- [setEnvSimAppID](#) *Set/Get environmental simulation application identifier*
- [setEnvironmentType](#) *Set/Get environment type*
- [setEnvironmentType](#) *Set/Get environment type*
- [setEnvironmentalProcessID](#) *Set/Get originating environmental process ID*
- [setEnvironmentalStatusBit](#) *Set/Get an environmental status bit*
- [setEnvironmentalStatus](#) *Set/Get the environmental status (as an 8-bit value)*
- [setEventID](#) *Set/Get event ID*
- [setEventID](#) *Set/Get event identifier*
- [setEventID](#) *Set/Get event identifier*
- [setEventID_Application](#) *Set/Get event ID application (optional)*
- [setEventID_Application](#) *Set/Get event ID application (optional)*
- [setEventID_Event](#) *Set/Get event ID event*
- [setEventID_Event](#) *Set/Get event ID event*
- [setEventID_Site](#) *Set/Get event ID site (optional)*
- [setEventID_Site](#) *Set/Get event ID site (optional)*
- [setEventId](#)
- [setEventId](#)
- [setEventId](#)
- [setEventId](#)
- [setEventId](#) *Set/Get Event Id - not used until RPR 0*
- [setEventId](#) *Set/Get Event Id - not used until RPR 0*
- [setEventId](#)
- [setEventNum](#)
- [setEventType](#)
- [setEventType](#) *Set/Get event type*
- [setEventType](#) *Set/Get the event type*
- [setEventType](#) *Set/Get the event type*
- [setExerciseld](#) *Set/Get the current exercise ID*

- [setExerciseld](#)
- [setExternalBuffer](#) Set the working buffer to an externally supplied buffer
- [setExtra](#)
- [setFieldNumber](#) Set/Get field number
- [setFieldOffset](#) Set/Get the field offset
- [setFieldScale](#) Set/Get the field scale
- [setFileOutput](#)
- [setFilename](#) set/get file to open for playback
- [setFilename](#) set/get the name of file into which traffic will be recorded
- [setFireFlag](#) Set a flag of the flag field
- [setFireFlagsField](#) Set Fire Flags
- [setFireMissionIndex](#)
- [setFirePowerDisabled](#) Set/Get whether entity is fire power disabled or not
- [setFiringEntityID](#) Set/Get firing entity ID
- [setFiringEntityID](#) Set/Get firing entity ID
- [setFixedDatumId](#)
- [setFixedDatumId](#) Set/Get the fixed datum id at the specified index
- [setFlamesPresent](#) Set/Get whether flames are present or not
- [setForceID](#) Set/Get the force ID
- [setForceID](#) Set/Get the force ID
- [setForceID](#) Set/Get the force ID
- [setForceID](#) Set/Get the force ID
- [setForceID](#) Set/Get the force ID
- [setForceID](#) Set/Get the force ID
- [setForceID](#) Set/Get the force ID
- [setForceId](#) Set/Get the force type
- [setForceId](#)
- [setForceId](#)
- [setFormation](#) Set/Get the formation
- [setFormation](#) Set/Get the formation
- [setFractionalThreshold](#) Set/Get fractional threshold (fraction between 0 and 1)
- [setFrequencyBandwidth](#) Set/Get the frequency bandwidth
- [setFrequency](#) Set/Get Frequency
- [setFrequency](#) Set/Get the frequency
- [setFrequency](#) Set the frequency, either as upper and lower word, or as full 64 bit integer
- [setFrequency](#)
- [setFrequencyHopInUse](#) Set/Get the frequency hop in use
- [setFrequencyRange](#) Set/Get Frequency Range
- [setFromNetBeam](#) Set from network representation

- [setFromNet](#) *Set from network representation*
- [setFromNet](#) *initializes the part types and parameters, and sets values as well*
- [setFromNet](#) *Set from network representation*
- **setFromNet :**
 - in [DtEntityIdentifier](#)
 - in [DtEntityType](#)
- [setFromNet](#)
- [setFromNet](#) *Set from network representation*
- [setFromNet](#)
- [setFromNetFixed](#) *Sets the fixed data from a network representation of a fixed datum record array*
- [setFromNetVar](#) *Sets the variable data from a network representation consisting of the variable length records concatenated together*
- [setFromOther](#) *Sets the contents of this hashlist to match those of other*
- **setFromString :**
 - in [DtAbsoluteVCPdu](#)
 - in [DtCompassVCPdu](#)
 - in [DtGroupVCPdu](#)
 - in [DtMimicTrackVCPdu](#)
 - in [DtMimicVCPdu](#)
 - in [DtOrbitVCPdu](#)
 - in [DtTetherTrackVCPdu](#)
 - in [DtTetherVCPdu](#)
 - in [DtTrackVCPdu](#)
 - in [DtViewControlPdu](#)
- [setFromVec](#)
- [setFrozenBehavior](#)
- [setFrozenBehavior](#) *Set/Get the frozen behavior*
- [setFrozen](#) *Set/Get whether frozen or not*
- [setFrozen](#) *Set/Get whether frozen or not*
- [setFuelStatus](#) *Set/Get the fuel status*
- [setFuelStatus](#) *Set/Get the fuel status*
- [setFuelStatus](#) *Set/Get the fuel status*
- [setFunction](#) *Set/Get the function*
- [setFuseType](#)
- [setFuseType](#)
- [setGeocentric](#) *Set from geocentric coordinate (convert geocentric to geodetic)*
- [setGeocentric](#) *Set from geocentric coordinate (convert geocentric to UTM)*
- [setGeodetic](#) *Set from geodetic coordinate (convert geodetic to UTM)*
- [setGridAxisRecord](#)

- [setGridDataRecord](#) Set/Get the grid data record type and values of record index to values specified by the initializer in the second argument
- [setGroundBurialDepthOffset](#) Set/Get a ground burial depth offset
- [setGroundLogisticsVehicleRecord](#) Set/Get a GroundLogisticsVehicleRecord
- [setGroundMaintenanceStatus](#) Set/Get the ground maintenance status
- [setGroupBitField](#) Set/Get the group bit field
- [setGroupCmd](#)
- [setGroupEntityID](#) Set/Get group entity ID
- [setGuise](#)
- [setGuise](#) Set/Get the alternate entity type or guise
- [setGunElevation](#) Set/Get the gun elevation
- [setGunElevation](#) Set/Get the gun elevation
- [setGunElevationRate](#) Set/Get the gun elevation rate
- [setGunElevationRate](#) Set/Get the gun elevation rate
- [setHasAmmunitionSupplyCap](#) Set/Get whether entity has ammunition supply capability or not
- [setHasFuelSupplyCap](#) Set/Get whether entity has fuel supplying capability
- [setHasRecoveryCap](#) Set/Get whether entity has recovery capability
- [setHasRepairCap](#) Set/Get whether entity has repair capability
- [setHatchState](#) Set/Get the entity's hatch state
- [setHeadAzimuth](#) Set/Get the head azimuth
- [setHeadElevation](#) Set/Get the head elevation
- [setHeadElevationRate](#) Set/Get the head elevation rate
- [setHeadScanRate](#) Set/Get the head scan rate
- **setHeading :**
 - in [DtGroupVCPdu](#)
 - in [DtOrbitVCPdu](#)
- **setHeadingOffset :**
 - in [DtCompassVCPdu](#)
 - in [DtGroupVCPdu](#)
 - in [DtOrbitVCPdu](#)
- [setHeartbeat](#) set/get the heartbeat
- [setHighDensity](#) Set/Get the high density track/jam flag
- [setHorDeflection](#) Set/Get the horizontal deflection
- [setHorizontalDeviation](#) Set/Get the horizontal deviation
- [setHorizontalDeviation](#) Set/Get the horizontal deviation
- [setHorizontalError](#) Set/Get horizontal error
- [setHost](#)
- [setHost](#)
- [setHostId](#) Set/Get the host identifier

- [setIR_Appearance](#) Set/Get IR appearance (as a 32 bit integer)
- [setId](#)
- [setIgnoreFlags](#)
- [setImmobilized](#) Set/Get whether the entity is immobilized or not
- [setInformationLayers](#) Set/Get the information layers field (each layer has a corresponding bit in this field which indicates the presence of that layer)
- [setInfraRedSigRepIndex](#) Set/Get infra-red signature representation index
- [setInitialIndexXi](#) Set/Get the initial index
- [setInputSource](#) Set/Get the input source
- [setInputSource](#)
- [setInterleafFactor](#) Set/Get the interleaf factor
- [setIsConcealed](#) Set/Get whether entity is concealed or not
- [setIssued](#) MUTATOR FUNCTIONS *****
- [setIssuingEntityID](#) Set/Get issuing entity identifier
- [setJammingModeSequence](#)
- [setJammingModeSequence](#)
- [setJammingModeSequence](#) Set/Get Jamming Mode Sequence
- [setKind](#) the funcs used to take longs, so for back-compat:
- [setKind](#) Set/Get the kind
- [setKind](#)
- [setLasedEntityId](#)
- [setLasedSpotRelative](#)
- [setLasedSpotWorld](#)
- [setLaserCode](#)
- [setLasingEntityId](#)
- [setLat](#)
- [setLatitude](#) Set/Get the reference latitude
- [setLauncherFlashPresent](#) Set/Get whether launcher flash is present or not
- [setLauncherRaised](#) Set/Get whether entity has launcher raised or not
- [setLayer](#) Set/Query an information layer field for a particular layer
- [setLayerInfo](#) Set/Get the layer specific information
- [setLayerLength](#) Set/Get the layer length
- [setLayerNumber](#) Set/Get the layer number
- [setLgrCommand](#) set is done by way of constructing the proper subclass
- [setLifeformState](#) Set/Get entity life form state
- [setLightState](#) Set/Get light state of entity
- [setLinearAcceleration](#)
- [setLinearAcceleration](#) Set/Get entity linear acceleration
- [setLinearSegmentParameter](#) Set/Get a linear segment parameter record

- [setLocation](#) *Set/Get the location of center of mass*
- [setLocation](#) *Set absolute location*
- [setLocation](#) *Set/Get the location of the center of mass in world coordinates*
- [setLocation](#) *Set/Get the location in local coordinates*
- [setLocation](#) *Set absolute location*
- [setLocation](#)
- [setLocation](#)
- [setLocation](#)
- [setLocation](#) *Set/Get location*
- [setLocation](#)
- [setLocation](#) *Set/Get location*
- [setLocation](#) *Set/Get part location*
- [setLocation](#) *Set/Get entity location (optional)*
- [setLocation](#) *Set/Get entity location*
- [setLocation](#) *Set/Get location*
- [setLocation](#) *Set/Get location of minefield*
- [setLocation](#) *Set/Get location*
- [setLocation](#) *Set absolute location*
- [setLocation](#) *Set/Get the location*
- [setLocation](#) *Set/Get the location in UTM coordinates*
- [setLocationEntity](#) *Set/Get entity location (entity coordinates)*
- [setLocationOfImpact](#) *Set/Get location of impact*
- [setLocationWorld](#) *Set/Get entity location (world coordinates)*
- [setLon](#) *Set/Get the longitude*
- [setLongitude](#) *Set/Get the reference longitude*
- [setMajor](#)
- [setMajorModulationType](#) *Set/Get the major modulation type*
- [setMarkingCharSet](#)
- [setMarkingCharSet](#)
- [setMarkingCharacterSet](#) *Set/Get marking character set*
- [setMarkingCharacterSet](#) *Set/Get marking character set*
- [setMarkingCharacterSet](#) *Set/Get marking character set*
- [setMarkingText](#) *Set/Get marking text (copies 31 bytes)*
- [setMarkingText](#) *Set/Get marking text (copies 31 bytes)*
- [setMarkingText](#)
- [setMarkingText](#) *Set/Get the marking text*
- [setMarkingText](#) *Set/Get marking text (copies 11 bytes)*
- [setMass](#) *Set/Get mass*

- [setMass](#)
- [setMasterChannelID](#) Set/Get master channel ID
- [setMasterDeviceID](#) Set/Get master communications device ID
- [setMasterEntityID](#) Set/Get master entity identifier
- **setMasterSmoothEnable :**
 - in [DtSmoothOrient](#) Set/Get the value of the "master" smoothing-enabled switch
 - in [DtSmoothPos](#) Set/Get the value of the "master" smoothing-enabled switch
- [setMasterSmoothEnable](#) Set/Get the value of the "master" smoothing-enabled switch
- [setMcastTtl](#)
- [setMcastTtl](#)
- [setMcastTtl](#)
- [setMeasuredSpeed](#) Set/Get measured speed
- [setMessage](#)
- [setMineEmplacementTime](#) Set/Get a mine emplacement time
- [setMineEntityID](#) Set/Get a mine entity ID
- [setMineLocation](#) Set/Get a mine location
- [setMineOrientation](#) Set/Get mine orientation
- [setMineType](#) Set/Get the mine type record
- [setMineType](#) Set/Get a mine type record
- [setMinefieldAlgae](#) Set/Get minefield algae
- [setMinefieldID](#) Set/Get minefield ID
- [setMinefieldID](#) Set/Get minefield ID
- [setMinefieldID](#) Set/Get minefield ID
- [setMinefieldID](#) Set/Get minefield ID
- [setMinefieldProtocolMode](#) Set/Get which network protocol mode is being used in the minefield transmission
- [setMinefieldSequenceNumber](#) Set/Get minefield sequence number
- [setMinefieldType](#) Set/Get minefield type
- [setModelType](#) Set/Get the model type
- [setModifications](#) Set/Get the modifications (by 8-bit integer)
- [setModifications](#) Set/Get the modifications (by 8-bit integer)
- [setModifications](#) Set/Get the modifications (by 8-bit integer)
- [setModificationsField](#) Set/Get the modifications (by field)
- [setModificationsField](#) Set/Get the modifications (by field)
- [setModificationsField](#) Set/Get the modifications (by field)
- [setModifier](#) Set/Get modifier
- [setModulationParams](#) Set/Get the modulation parameters' network representation
- [setModulationParams](#)
- [setModulationParamsLength](#)
- [setModulationSystemType](#) Set/Get the modulation system

- [setModulationType](#)
- [setMovementSpeed](#) Set/Get the movement speed
- [setMovementSpeed](#) Set/Get the movement speed
- [setMunitionID_Application](#) Set/Get munition ID application (optional)
- [setMunitionID_Application](#) Set/Get munition ID application (optional)
- [setMunitionID_Entity](#) Set/Get munition ID entity (optional)
- [setMunitionID_Entity](#) Set/Get munition ID entity (optional)
- [setMunitionID_Site](#) Set/Get munition ID site (optional)
- [setMunitionID_Site](#) Set/Get munition ID site (optional)
- [setMunitionId](#)
- [setMunitionId](#)
- [setMunitionType](#)
- [setMunitionType](#)
- [setNature](#) Set/Get nature
- [setNomenclature](#)
- [setNomenclatureVersion](#)
- [setNorth](#) Set/Get north component
- [setNozzleSystem](#) Set/Get a vectoring nozzle system data record
- **setNthSupply :**
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [setNthSupply](#)
- [setNumAdditionalPassiveAct](#) Set/Get number of additional passive activities May return DtFalse if PDU size would exceed the maximum PDU size
- [setNumArtParams](#)
- [setNumArtPlusAttachedParams](#)
- [setNumAttachedParts](#) Set/Get the number of attached parts
- [setNumBeams](#)
- [setNumBeams](#)
- [setNumBeams](#) Set/Get number of beam data records for a particular emitter system
- [setNumDataBits](#) Set/get data length in bits
- [setNumDataBits](#)
- [setNumDataBytes](#) Set/Get the number of data bytes When number of data bytes is changed, old values are not preserved
- [setNumDataBytes](#) Set/get data length in bytes
- [setNumDataBytes](#)
- [setNumDataValues](#) Set/Get the number of data values Old values are not preserved when number of values is changed
- [setNumDataValues](#) Set/Get the number of data values Old values are not preserved when the number of values is changed

- [setNumEmitterSystems](#) Set/Get number of emitter systems May return DtFalse if PDU size would exceed the maximum PDU size
- [setNumEntities](#) Set/Get the number of entities
- **setNumEntitiesToAdd :**
 - in [DtGroupVCPdu](#)
 - in [DtModelessVCPdu](#)
- **setNumEntitiesToRemove :**
 - in [DtGroupVCPdu](#)
 - in [DtModelessVCPdu](#)
- [setNumFixedFields](#) Sets the number of fixed or variable fields to newNum
- [setNumFixedFields](#) Sets the number of fixed or variable fields to newNum
- [setNumGridAxes](#) Set/Get number of grid axes
- [setNumMineTypes](#) Set/Get the number of mine types
- [setNumMissingPDUs](#) Set/Get the number of missing PDUs
- [setNumNozzleSystems](#) Set/Get number of nozzle systems
- [setNumObjects](#) Set/Get number of objects
- [setNumObjects](#) Set/Get number of objects
- [setNumParameterSets](#) Get the number of parameter sets
- [setNumParmRecords](#) Set/Get number of parameter records (this is not a PDU field)
- [setNumPerimeterPoints](#) Set/Get the number of perimeter points
- [setNumPerimeterPoints](#) Set/Get the number of perimeter points
- [setNumPoints](#) Set/Get the number of points
- [setNumPointsXi](#) Set/Get the number of points
- [setNumPointsXi](#) Set/Get the number of points
- [setNumPropulsionSystems](#) Set/Get number of propulsion systems
- [setNumRecordSets](#) Set/Get the number of record sets
- [setNumRecords](#) Set number of records
- [setNumRecords](#) Set number of records
- [setNumRecords](#) Set/Get the number of records to be queried for
- [setNumSamples](#) Set/get number of samples
- [setNumSamples](#)
- [setNumSegments](#) Set/Get the number of segments
- [setNumSensorTypes](#) Set/Get the number of sensor types
- [setNumShafts](#) Set/Get number of shafts May return DtFalse if PDU size would exceed the maximum PDU size
- **setNumSupplyTypes :**
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [setNumSupplyTypes](#)
- [setNumSystems](#)

- [setNumSystems](#)
- [setNumTargetsInBeam](#)
- [setNumTargetsInBeam](#)
- [setNumTripWires](#) Set/Get the number of trip/detonation wires
- [setNumVarFields](#)
- [setNumVarFields](#)
- [setNumVarFields](#)
- [setNumVertices](#) Set/Get the number of vertices
- [setNumberOfAggregates](#) Set/Get the number of aggregates
- [setNumberOfAppearanceRecords](#) Set/Get the number of appearance records
- [setNumberOfBeams](#) Set/Get the number of beams
- [setNumberOfDIS_Entities](#) Set/Get the number of DIS entities
- [setNumberOfDIS_Entities](#) Set/Get the number of DIS entities
- [setNumberOfDIS_Subaggregates](#) Set/Get the number of DIS sub-aggregates
- [setNumberOfDIS_Subaggregates](#) Set/Get the number of DIS sub-aggregates
- [setNumberOfDatumRecords](#) Set/Get the number of datum records
- [setNumberOfDatumRecords](#) Set the number of variable datum records
- [setNumberOfEntities](#) Set/Get the number of aggregates
- [setNumberOfEnvironmentalRecords](#) Set/Get number of environment records
- [setNumberOfMines](#) Set/Get the number of mines (this call erases any data which was set in the variable length fields)
- [setNumberOfPdus](#) Set/Get the number of PDUs
- [setNumberOfSensorTypes](#) Set/Get the number of sensor types (this call erases any data which was set in the variable length fields)
- [setNumberOfSilentEntities](#) Set/Get the number of silent entities
- [setNumberOfSilentEntities](#) Set/Get the number of silent entities
- [setNumberOfSilentSubaggregates](#) Set/Get the number of silent aggregates
- [setNumberOfSilentSubaggregates](#) Set/Get the number of silent aggregates
- [setObject](#) Set index'th object designator
- [setObject](#) Set index'th object designator
- [setObjectID](#) Set/Get object ID
- [setObjectID](#) Set/Get object ID
- [setObjectID](#) Set/Get object ID
- [setObjectType](#) Set/Get the object type
- [setObjectType](#) Set/Get the object type
- [setObjectType](#) Set/Get the object type
- [setOffset](#) set the origin of the local system in geocentric coords
- **setOffset :**
 - in [DtCoordTransform](#)

- in [DtCoordTranslation](#)
- **setOffset :**
 - in [DtMimicTrackVCPdu](#)
 - in [DtMimicVCPdu](#)
 - in [DtTetherTrackVCPdu](#)
- [setOperational](#) *Set/Get fundamental operational data*
- [setOperator](#)
- [setOperator](#)
- [setOrderedShaftRPM](#) *Set/Get the ordered shaft RPM*
- [setOrientation](#)
- [setOrientation](#) *Set/Get orientation*
- [setOrientation](#) *Set absolute orientation*
- [setOrientation](#) *Set/Get the orientation*
- [setOrientation](#) *Set/Get the orientation*
- [setOrientation](#) *Set absolute orientation*
- [setOrientation](#)
- [setOrientation](#) *Set/Get orientation*
- [setOrientation](#) *Set/Get orientation*
- [setOrientation](#) *Set/Get entity orientation*
- [setOrientation](#) *Set/Get entity orientation*
- [setOrientation](#) *Set/Get orientation*
- [setOrientation](#) *Set/Get orientation*
- [setOrientation](#) *Set/Get orientation*
- [setOrientation](#) *Set absolute orientation*
- [setOrientation](#)
- [setOrientation](#) *Set/Get the orientation*
- [setOrientationError](#) *Set/Get orientation error*
- [setOriginatingEntityID](#) *Set/Get originating entity identifier*
- [setOriginatingEntityID](#) *Set/Get originating entity identifier*
- [setOriginatingEntityID](#) *Set/Get originating entity identifier*
- [setOriginatingEntityID](#) *Set/Get originating entity identifier*
- [setOriginatingEntityID](#) *Set/Get originating entity identifier*
- [setOriginatingEntityID](#) *Set/Get originating entity identifier*
- [setPDU_SequenceNumber](#) *Set/Get the PDU sequence number*
- [setPDU_SequenceNumber](#) *Set/Get a PDU sequence number*
- [setPRF](#) *Set/Get PRF*
- [setPacketServerExternal](#)
- [setPacketServerInternal](#)
- [setPacketServerUnbundling](#)

- [setPaintScheme](#) Set/Get paint scheme
- [setParamNum](#) Set the parameter number of the specified parameter type
- [setParameter1](#) Set/Get parameter 1
- [setParameter2](#) Set/Get parameter 2
- [setParameterBit](#) Set/Get parameter code bit
- [setParameterCode](#) Set/Get parameter code elements
- [setParameterData](#)
- [setParameterData](#)
- [setParameter](#) Set/Get a parameter (unsigned 16-bit values)
- [setParmData](#) Set/Get the fundamental parameter data records for the second layer
- [setParmIndex](#) Set/Get the parameter index
- [setParmRecord](#)
- [setPartNum](#)
- [setPassiveParameterIndex](#) Set/Get passive parameter index
- [setPduFactory](#) Rather than just modifying the ExerciseConn's current PDU factory, you can tell the ExerciseConn to use a different PDU factory object
- [setPduNumber](#) Set/Get the PDU index number
- [setPduTotal](#) Set/Get total number of PDUs used to transmit the environmental data
- [setPerimeterPoint](#) Set/Get a perimeter point coordinate record
- [setPerimeterPoint](#) Set/Get a perimeter point coordinate record, returns DtFalse if the index is out of range
- [setPgRF](#) Set/Get the PgRF
- [setPhase](#) Set/Get the phase
- [setPhi](#) Set/Get the entity orientation phi
- [setPhi](#) Set/Get the entity orientation phi
- [setPhi](#) Set/Get the entity orientation phi
- [setPhi](#) Set/Get the entity orientation phi
- [setPhi](#) Set/Get the entity orientation phi
- [setPhi](#) Set/Get Phi
- [setPhi](#)
- **setPitch :**
 - in [DtCompassVCPdu](#)
 - in [DtGroupVCPdu](#)
 - in [DtOrbitVCPdu](#)
- **setPitchOffset :**
 - in [DtCompassVCPdu](#)
 - in [DtGroupVCPdu](#)
 - in [DtOrbitVCPdu](#)
- [setPointLocation](#) Set/Get a point location
- [setPosition](#) Set/Get position

- [setPositionError](#) Set/Get position error
- [setPower](#)
- [setPower](#)
- [setPower](#) Set/Get the power
- [setPower](#)
- [setPowerPlantOn](#) Set/Get whether entity's power plant is on or not
- [setPowerSetting](#) Set/Get the power setting
- [setPrimaryAmmunition](#) Set/Get the primary ammunition
- [setPrimaryAmmunition](#) Set/Get the primary ammunition
- [setPrimaryAmmunition](#) Set/Get the primary ammunition
- [setPrimaryAmmunition](#) Set/Get the primary ammunition
- [setPrimary](#)
- [setPrimaryFusing](#) Set/Get primary fusing
- [setPrimaryWeaponState](#) Set/Get primary weapon deployment state
- [setPropulsionPlantConfig](#) Set/Get propulsion plant configuration
- [setPropulsionSystem](#) Set/Get a propulsion system data record
- [setPsMsgTimeout](#)
- [setPsi](#) Set/Get the entity orientation psi
- [setPsi](#) Set/Get the entity orientation psi
- [setPsi](#) Set/Get the entity orientation psi
- [setPsi](#) Set/Get the entity orientation psi
- [setPsi](#) Set/Get the entity orientation psi
- [setPsi](#) Set/Get Psi
- [setPsi](#)
- [setPsuedoNoiseSpectrumInUse](#) Set/Get the Psuedo Noise Spectrum In Use
- [setPulseWidth](#) Set/Get Pulse Width
- [setPulseWidth](#) Set/Get the pulse width
- [setQuantity](#)
- [setQuantity](#)
- [setRadarSigRepIndex](#) Set/Get radar signature representation index
- [setRadialOffset](#)
- [setRadioEntityType](#) Set/Get the radio entity type
- [setRadioEntityType](#)
- [setRadioid](#) Set/Get the Receiver radio number
- [setRadioid](#) Set/Get the Transmitter radio number
- [setRadioid](#)
- [setRadioid](#)
- [setRadioid](#)
- **setRadius :**

- in [DtCompassVCPdu](#)
- in [DtOrbitVCPdu](#)
- [setRampDeployed](#) *Set/Get whether entity has ramp deployed or not*
- [setRange](#)
- [setRange](#) *Set/Get range*
- [setRate](#)
- [setRate](#)
- [setRealWorldTime](#)
- [setRealWorldTime](#) *Set/Get real world time*
- [setRealWorldTime](#)
- [setRealWorldTime](#) *Set/Get real world time*
- [setReason](#)
- [setReason](#) *Set/Get the stop reason*
- [setReceivedPower](#) *Set/Get the received power*
- [setReceivedPower](#)
- [setReceiverId](#) *set/get Object Identifier of intended recipient (logger)*
- [setReceiverId](#)
- [setReceiverState](#) *Set/Get the receiver state*
- [setReceiverState](#)
- [setReceivingEntityID](#) *Set/Get receiving entity identifier*
- [setReceivingEntityID](#) *Set/Get receiving entity identifier*
- [setReceivingEntityID](#) *Set/Get receiving entity identifier*
- [setReceivingEntityID](#) *Set/Get receiving entity identifier*
- [setReceivingEntityID](#) *Set/Get receiving entity identifier*
- [setReceivingID](#) *Set/Get the receiving ID*
- [setReceivingID](#) *Set/Get the receiving ID*
- [setReceivingID](#) *Set/Get the receiving ID*
- **setReceivingId :**
 - in [DtRepairCompletePdu](#)
 - in [DtRepairResponsePdu](#)
- **setReceivingId :**
 - in [DtResupplyCancelPdu](#)
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [setRecordBitLength](#) *Set/Get the record length in bits*
- [setRecordBitSize](#) *Set the size of an environmental record (in bits)*
- [setRecordByteLength](#) *Set/Get the record length in bytes*
- [setRecordCount](#) *Set/Get the number of records in this set When the number of records is changed, old records are not preserved*

- [setRecordID](#) Set/Get a record ID
- [setRecordSet](#) Set/Get a record set
- [setReferencePoint](#) Set/Get the reference point
- [setReferenceSystem](#) Set/Get the beam reference system
- [setReferencedObjectID](#) Set/Get referenced object ID
- [setReferencedObjectID](#) Set/Get referenced object ID
- [setReferencedObjectID](#) Set/Get referenced object ID
- [setReflectance](#) Set/Get a reflectance value
- [setRelativeAntennaLoc](#) Set/Get the relative location (wrapper around EmbedSysSR member func, for compatibility with DIS transmitter PDU)
- [setRelativeAntennaLoc](#)
- [setRelativePosition](#) Set/Get the relative position
- [setRelativeSpotLocation](#) Set/Get the relative spot location
- [setRepairResult](#)
- [setRepairType](#)
- **setRepairingId :**
 - in [DtRepairCompletePdu](#)
 - in [DtRepairResponsePdu](#)
- [setReqRelService](#) Set/Get the required reliability service
- [setReqRelService](#) Set/Get the required reliability service
- [setReqRelService](#) Set/Get the required reliability service
- [setReqRelService](#) Set/Get the required reliability service
- [setReqRelService](#) Set/Get required reliability service
- [setReqRelService](#) Set/Get required reliability service
- [setReqRelService](#) Set/Get the required reliability service
- [setReqRelService](#) Set/Get the required reliability service
- [setReqRelService](#) Set/Get required reliability service
- [setReqRelService](#) Set/Get the required reliability service
- [setReqRelService](#) Set/Get the required reliability service
- [setReqRelService](#) Set/Get required reliability service
- [setRequestID](#) Set/Get the request ID
- [setRequestID](#) Set/Get request ID
- [setRequestID](#) Set/Get request ID
- [setRequestID](#) Set/Get request ID
- [setRequestID](#) Set/Get request ID
- [setRequestID](#) Set/Get request ID
- [setRequestID](#) Set/Get request ID
- [setRequestID](#) Set/Get request ID
- [setRequestid](#)
- [setRequestid](#) Set/Get the request id

- [setRequestId](#)
- [setRequestId](#) *Set/Get the request id*
- [setRequestId](#)
- [setRequestId](#) *Set/Get the request id*
- [setRequestId](#)
- [setRequestId](#) *Set/Get the request id*
- [setRequestId](#)
- [setRequestId](#)
- [setRequestId](#) *Set/Get the request id*
- [setRequestId](#) *Set/Get the request id*
- [setRequestId](#)
- [setRequestId](#) *Set/Get the request id*
- [setRequestId](#)
- [setRequestId](#) *Set/Get the request id*
- [setRequestId](#)
- [setRequestStatus](#)
- [setRequestStatus](#) *Set/Get the request status*
- [setRequestedMineType](#) *Set/Get requested mine type*
- [setRequesterID](#) *Set/Get the requester ID*
- [setRequesterID](#) *Set/Get the requester ID*
- [setRequesterID](#) *Set/Get the requester ID*
- [setRequestingEntityID](#) *Set/Get requesting entity ID*
- [setRequestingEntityID](#) *Set/Get requesting entity ID*
- [setRequestingEntityID](#) *Set/Get requesting entity ID*
- [setRequestingId](#)
- [setResponseFlag](#)
- [setResponseFlag](#) *Set/Get the response flag*
- [setResponseSerialNumber](#) *Set/Get the response serial number*
- [setRestStatus](#) *Set/Get the rest status*
- [setResult](#)
- **setRoll :**
 - in [DtCompassVCPdu](#)
 - in [DtOrbitVCPdu](#)
- [setRotation](#) *set the rotation matrix that converts from geocentric to local coords*
- [setRotation](#)
- [setRotationError](#) *Set/Get rotation error*

- [setRotationThreshold](#) *Set/Get rotation threshold in radians*
- [setRotationThreshold](#) *Set/Get rotation threshold in radians*
- [setRotationalVelocity](#) *Set/Get absolute rotational velocity*
- [setRotationalVelocity](#) *Set/Get the rotational velocity*
- [setRotationalVelocity](#) *Set/Get the rotational velocity*
- [setRotationalVelocity](#) *Set/Get absolute rotational velocity*
- [setRotationalVelocity](#)
- [setRotationalVelocity](#) *Set/Get absolute rotational velocity*
- [setRotationalVelocity](#) *Set/Get the rotational velocity*
- [setSampleRate](#) *Set/get sample rate*
- [setSampleRate](#)
- [setSampleTime](#) *Set/Get sample time*
- [setSampleType](#) *Set/Get the grid data record sample type*
- [setScalarDetectionCoefficient](#) *Set/Get scalar detection coefficient*
- [setScanPattern](#) *Set/Get the acoustic scan pattern*
- [setSecondaryAmmunition](#) *Set/Get the secondary ammunition*
- [setSecondaryAmmunition](#) *Set/Get the secondary ammunition*
- [setSecondaryAmmunition](#) *Set/Get the secondary ammunition*
- [setSecondaryAmmunition](#) *Set/Get the secondary ammunition*
- [setSecondary](#)
- [setSecondaryFusing](#) *Set/Get secondary fusing*
- [setSecondaryOperational](#) *Set/Get secondary operational parameters*
- [setSecondaryWeaponState](#) *Set/Get secondary weapon deployment state*
- [setSegmentDepth](#) *Set/Get the segment depth*
- [setSegmentHeight](#) *Set/Get the segment height*
- [setSegmentLength](#) *Set/Get the segment length*
- [setSegmentNumber](#) *Set/Get the segment number*
- [setSegmentWidth](#) *Set/Get the segment width*
- [setSendNeeded](#) *force an state message to be generated*
- [setSendNeeded](#)
- [setSenderId](#) *set/get Object Identifier of sender, if applicable*
- [setSenderId](#)
- [setSenderId](#)
- [setSensorTypeCategory](#) *Set/Get a sensor type category*
- [setSensorType](#) *Set/Get the sensor type*
- [setSensorTypeSubcategory](#) *Set/Get a sensor type sub-category*
- [setSensorTypeSubcategory](#)
- [setSequenceNumber](#) *Set/Get sequence number*

- [setSequenceNumber](#) Set/Get minefield sequence number
- [setSerialNumber](#) Set/Get the serial number
- [setServiceType](#)
- [setServicingId](#)
- [setShaftRPM_RateOfChange](#) Set/Get the shaft RPM rate of change
- [setShaftSpeed](#) Set/Get a shaft speed data record
- [setSilentAggregateList](#) Set/Get a silent aggregate list
- [setSilentAggregateList](#) Set/Get a silent aggregate list
- [setSilentEntityList](#) Set/Get a silent entity list
- [setSilentEntityList](#) Set/Get a silent entity list
- [setSimulationTime](#)
- [setSimulationTime](#) Set/Get simulation time
- [setSite](#)
- [setSite](#)
- [setSite](#) Set/Get site
- [setSizeLength](#)
- [setSizeOffset](#)
- [setSmokePlumePresent](#) Set/Get whether a smoke plume is present or not
- **setSmoothEnable :**
 - in [DtSmoothOrient](#) Set/Get the individual smoothing switch
 - in [DtSmoothPos](#) Set/Get the individual smoothing switch
- [setSmoothEnable](#) Set/Get the individual smoothing switch
- **setSmoothPeriod :**
 - in [DtSmoothOrient](#) Set/Get smoothPeriod
 - in [DtSmoothPos](#) Set/Get smoothPeriod
- [setSmoothPeriod](#) Set/Get smoothing period - the amount of time over which smoothing takes place
- [setSnowBurialDepthOffset](#) Set/Get a snow burial depth offset
- [setSourceDeviceID](#) Set/Get source communications device ID
- [setSourceEntityID](#) Set/Get source entity identifier
- [setSourceLineID](#) Set/Get source line ID
- [setSpecificAppearanceAttribute](#) Set/Get specific appearance by attribute, the upper 32 bits
- [setSpecificAppearanceAttribute](#) Set/Get specific appearance by attribute, the upper 32 bits
- [setSpecificAppearanceAttribute](#) Set/Get specific appearance by attribute, the upper 32 bits
- [setSpecificAppearance](#) Set/Get specific appearance, the upper 32 bits (as a 32 bit integer)
- [setSpecificAppearance](#) Set/Get specific appearance, the upper 32 bits (as a 32 bit integer)
- [setSpecificAppearance](#) Set/Get specific appearance, the upper 32 bits (as a 32 bit integer)
- [setSpecific](#)
- [setSpectrum](#)
- [setSpotLinearAcceleration](#) Set/Get the spot linear acceleration

- [setStateChangeUpdate](#) Set/Get state/change update indicator
- [setStateUpdateIndicator](#)
- [setStateUpdateIndicator](#)
- [setStationName](#) Set/Get station name
- [setStationNumber](#) Set/Get the station number for a part
- [setStationNumber](#) Set/Get Station number
- [setStatus](#)
- [setString](#) Set/Get for myString - char pointer that holds the string non-virtual because only called by constructor
- [setSubCategory](#)
- [setSubProtocol](#) Set/Get the sub-protocol number
- [setSubProtocol](#) Set/Get the sub-protocol number
- [setSubProtocol](#) Set/Get the sub-protocol number
- [setSubProtocol](#) Set/Get the sub-protocol number
- [setSubProtocol](#) Set/Get the sub-protocol number
- [setSubaggregateID](#) Set/Get a DIS sub-aggregate ID
- [setSubaggregateID](#) Set/Get a DIS sub-aggregate ID
- [setSubcategory](#) Set/Get the subcategory
- [setSupplementalFuelStatus](#) Set/Get the fuel status
- [setSupplementalFuelStatus](#) Set/Get the fuel status
- **setSupplyingId :**
 - in [DtResupplyCancelPdu](#)
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [setSweepSync](#) Set/Get Sweep Sync
- [setSweepSync](#) Set/Get the Sweep Sync
- [setSystemFunction](#)
- [setSystemFunction](#)
- [setSystemID](#) Set/Get system ID
- [setSystemMode](#) Set/Get the system mode
- [setSystemName](#)
- [setSystemName](#)
- [setSystemName](#) Set/Get the system name
- [setSystemSpecificData](#) Set/Get the system specific data (will copy systemSpecificDataLength() bytes on set/get)
- [setSystemSpecificDataLength](#) Set/Get the system specific data length
- [setSystemStatusBit](#) Set/Get the system status by bit
- [setSystemStatus](#) Set/Get the system status
- [setSystemType](#) Set/Get the system type
- [setSystemType](#)
- **setT1data :**

- in [DtSmoothOrient](#)
- in [DtSmoothPos](#)
- [setTDL_Type](#) Set/get intercom TDL type
- [setTargetEntityID](#) Set/Get target entity ID (optional)
- [setTargetEntityID](#) Set/Get target entity ID (optional)
- [setTargetId](#)
- [setTargetId](#)
- [setTdlType](#)
- [setTentDeployed](#) Set/Get whether a tent is deployed or not
- [setTest](#)
- [setThermalContrast](#) Set/Get a thermal contrast value
- [setTheta](#) Set/Get the entity orientation theta
- [setTheta](#) Set/Get the entity orientation theta
- [setTheta](#) Set/Get the entity orientation theta
- [setTheta](#) Set/Get the entity orientation theta
- [setTheta](#) Set/Get the entity orientation theta
- [setTheta](#) Set/Get Theta
- [setTheta](#)
- [setThreshMode](#) Set/Get the threshold mode
- [setTime](#) set/get time to jump to within logger playback file
- [setTime](#) set/get amount of time to skip forward or back in playback file
- [setTime](#) Get the time contained in the timestamp to time % an hour
- [setTime](#) Set/Get the time
- [setTimeForSets](#)
- [setTimeHopInUse](#) Set/Get the Time Hop In Use
- [setTimeInterval](#)
- [setTimeInterval](#) Set/Get the time interval
- [setTimeScale](#) set/get playback speed
- [setTimeStamp](#)
- [setTimeStampType](#) Set/Get flag indicating whether we're using absolute timestamps for dead reckoning
- [setTimeStampType](#) Set/Get time stamp type (DtTimeStampAbsolute or DtTimeStampRelative)
- [setTimeThreshold](#) Set/get threshold for timeouts
- [setTimeThreshold](#) Set/Get time threshold
- [setTimeout](#) set/get timeout
- [setTimeoutInterval](#) Set/Get the remote object timeout period
- [setTimeoutProcessing](#) Set whether or not timeout processing occurs
- [setTopoUpOffset](#)
- [setTotalValues](#) Set/Get total values
- **setTrackId :**

- in [DtMimicTrackVCPdu](#)
- in [DtTetherTrackVCPdu](#)
- in [DtTrackVCPdu](#)
- [setTrackJamTarget](#)
- [setTrackJamTarget](#)
- [setTrackingAllTargets](#)
- [setTrackingAllTargets](#)
- [setTrailState](#) *Set/Get trailing effect state*
- [setTransEntityID](#) *Set/get transmitting entity identifier*
- [setTransferType](#) *Set/Get transfer type*
- [setTransferredEntityID](#) *Set/Get transferred entity identifier*
- [setTranslationThreshold](#) *Set/Get translation threshold in meters*
- [setTranslationThreshold](#) *Set/Get translation threshold in meters*
- [setTransmitEntityId](#)
- [setTransmitLineState](#) *Set/Get transmit line state*
- [setTransmitPriority](#) *Set/Get transmit priority*
- [setTransmitRadioid](#) *Set/Get the received transmitter*
- [setTransmitRadioid](#)
- [setTransmitState](#) *Set/Get the transmit state*
- [setTransmitState](#)
- [setTspiField](#) *Set TSPI Flags*
- [setTspiFlag](#) *Set a flag of the TSPI field*
- [setTurretAzimuth](#) *Set/Get the turret azimuth*
- [setTurretAzimuth](#) *Set/Get the turret azimuth*
- [setTurretSlewRate](#) *Set/Get the turret slew rate*
- [setTurretSlewRate](#) *Set/Get the turret slew rate*
- [setType](#) *Set/Get timestamp type*
- [setUnbundling](#)
- [setUnbundling](#)
- [setUnbundling](#)
- [setUnitSurfaceNormal](#) *Set/Get unit surface normal*
- [setUp](#)
- [setUpdateNumber](#) *Set/Get update number*
- [setUpdateNumber](#) *Set/Get update number*
- [setUpdateNumber](#) *Set/Get update number*
- [setUserData](#) *set/get the user data*
- [setVal](#) *Set/Get the value for the specified parameter type*
- [setValsFromNet](#) *sets values from net representation, but does not change the set of valid parts and parameters*

- [setVarDataBytes](#) *Allocates length bytes for the index'th variable datum record*
- [setVarDataBytes](#)
- [setVarDataBytes](#) *Allocates length bytes for the index'th variable datum record*
- [setVarDatumId](#)
- [setVarDatumId](#) *Set/Get the variable datum id at the specified index*
- [setVariableDatumRecord](#) *Set/Get a variable datum record*
- [setVariableDatumRecord](#) *Set/Get a variable datum record*
- [setVectorDim](#) *Set/Get vector dimension*
- [setVelocity](#) *Set/Get the velocity*
- [setVelocity](#) *Set absolute velocity*
- [setVelocity](#) *Set/Get the velocity*
- [setVelocity](#) *Set/Get the velocity*
- [setVelocity](#)
- [setVelocity](#) *Set absolute velocity*
- [setVelocity](#)
- [setVelocity](#)
- [setVelocity](#) *Set/Get velocity*
- [setVelocity](#)
- [setVelocity](#) *Set/Get entity linear velocity*
- [setVelocity](#) *Set/Get entity linear velocity*
- [setVelocity](#) *Set/Get entity linear velocity*
- [setVelocity](#) *Set absolute velocity*
- [setVelocity](#) *Set/Get the velocity*
- [setVersion](#) *Set/Get protocol version from PDU header*
- [setVertDeflection](#) *Set/Get the vertical deflection*
- [setVertex](#) *Set/Get a vertex*
- [setVerticalDeviation](#) *Set/Get the vertical deviation*
- [setVerticalDeviation](#) *Set/Get the vertical deviation*
- [setVerticalError](#) *Set/Get vertical error*
- [setViewerCommand](#)
- [setViewerId](#)
- [setVisualAppearanceAttribute](#) *Set/Get appearance by attribute*
- [setVisualAppearance](#) *Set/Get visual appearance (as a 32 bit integer)*
- [setWarheadType](#)
- [setWarheadType](#)
- [setWaterBurialDepthOffset](#) *Set/Get a water burial depth offset*
- [setWaterStatus](#) *Set/Get the water status*
- [setWavelength](#)

- [setWavelength](#)
- [setWorldAntennaLoc](#) *Set/Get the world location*
- [setWorldAntennaLoc](#)
- [setWorldLocation](#)
- [setWorldLocation](#)
- [setWorldLocation](#)
- [setX](#) *Set/Get X*
- [setX](#) *Set/Get X*
- [setX](#) *Set/Get the X Component*
- [setX_Offset](#) *Set/Get the entity location X offset*
- [setX_Offset](#) *Set/Get the entity location X offset*
- [setX_Offset](#) *Set/Get the entity location X offset*
- [setX_Offset](#) *Set/Get the entity location X offset*
- [setX_Offset](#) *Set/Get the entity location X offset*
- [setXi](#) *Set/Get an Xi value*
- [setY](#) *Set/Get Y*
- [setY](#) *Set/Get Y*
- [setY](#) *Set/Get the Y Component*
- [setY_Offset](#) *Set/Get the entity location Y offset*
- [setY_Offset](#) *Set/Get the entity location Y offset*
- [setY_Offset](#) *Set/Get the entity location Y offset*
- [setY_Offset](#) *Set/Get the entity location Y offset*
- [setY_Offset](#) *Set/Get the entity location Y offset*
- [setZ](#) *Set/Get Z*
- [setZ](#) *Set/Get Z*
- [setZ_Offset](#) *Set/Get the entity location Z offset*
- [setZ_Offset](#) *Set/Get the entity location Z offset*
- [setZ_Offset](#) *Set/Get the entity location Z offset*
- [setZ_Offset](#) *Set/Get the entity location Z offset*
- [setZ_Offset](#) *Set/Get the entity location Z offset*
- **setdata :**
 - in [DtDatumMapEntry](#)
 - in [DtDmeGlobalFcns](#)
- [shaftRPM_RateOfChange](#)
- [shaftSpeedRecordOffset](#) *Returns a buffer offset to the given shaft speed record*
- [shouldBeTimedOut](#) *Returns whether or not the object should be timed out*
- [silentAggregateOffset](#) *Returns an offset from the beginning of variable length data to a particular silent aggregate list*
- [silentEntityOffset](#) *Returns an offset from the beginning of variable length data to a particular silent entity system*

list

- [simulationTime](#)
- [simulationTime](#)
- [site](#)
- [site](#)
- [site](#)
- [sizeChanged](#) *Called by changeSize*
- [sizeChanged](#) *Really, should not be pure virtual in DtBaseMsg, but since it is, we need a null definition here*
- [sizeChanged](#) *Called by changeSize*
- [sizeLength](#) *Get/Set mySizeLength*
- [sizeOffset](#) *Get/Set mySizeOffset*
- [smokePlumePresent](#)
- **smoothEnable :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [smoothEnable](#)
- **smoothPeriod :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [smoothPeriod](#)
- [snipBack](#)
- [snipBackToFirst](#)
- [snipFront](#) *truncates everything before or after the character, including the character, respectively Normally, snipFront stops at the first occurrence of "ch", and snipBack stops at the last occurrence of "ch"*
- [snipFrontToLast](#)
- [snowBurialDepthRecordOffset](#) *Returns a buffer offset to the snow burial depth records*
- [socket](#)
- [socketList](#)
- [socketMgr](#)
- [sourceDeviceID](#)
- [sourceEntityID](#)
- [sourceLineID](#)
- [specificAppearanceAttribute](#)
- [specificAppearanceAttribute](#)
- [specificAppearanceAttribute](#)
- [specificAppearance](#)
- [specificAppearance](#)
- [specificAppearance](#)
- [specific](#)

- [spectrum](#)
- [spotLinearAcceleration](#)
- [sr](#) *Synonym for stateRep()*
- [startRegistrationForObjectClass](#) //////////////////////////////////
- [stateChangeUpdate](#)
- [stateMsg](#) *create a state message from a local and remote state repository*
- [stateMsg](#)
- [stateRep](#) *Returns a pointer to the StateRepository - through which you can inspect current state information*
- [stateRep](#) *Returns a pointer to the StateRepository - through which you can inspect current state information*
- [stateRep](#) *Returns a pointer to the StateRepository - through which you can inspect current state information*
- [stateRep](#) *Returns a pointer to the StateRepository - through which you can inspect current state information*
- [stateUpdateIndicator](#)
- [stateUpdateIndicator](#)
- [stationList](#) *Returns a zero-termination list of station numbers that are valid in this attached part list*
- [stationName](#)
- [stationNumber](#) *Returns the station number for a part*
- [stationNumber](#)
- [status](#)
- [stopRegistrationForObjectClass](#)
- **string :**
 - in [DtAbsoluteVCPdu](#)
 - in [DtCompassVCPdu](#)
- **string :**
 - in [DtEntityIdentifier](#)
 - in [DtEntityType](#)
- [string](#)
- **string :**
 - in [DtGroupVCPdu](#)
 - in [DtMimicTrackVCPdu](#)
 - in [DtMimicVCPdu](#)
 - in [DtOrbitVCPdu](#)
- [string](#)
- [string](#) *returns the char* internal representation of the string*
- [string](#)
- **string :**
 - in [DtTetherTrackVCPdu](#)
 - in [DtTetherVCPdu](#)
- [string](#) *returns a string representation*
- [string](#)

- [string](#) *Get a string representation of the vector*
- [string](#)
- [stringRep](#) *get the string representation of this key*
- [stringRep](#) *get the string representation of this key*
- [stringRep](#) *get the string representation of this key*
- [stringRep](#) *get the string representation of this key*
- [stringRep](#) *get the string representation of this key*
- [subAggregates](#) *Return the list of published subaggregates in the aggregate*
- [subCategory](#)
- [subProtocol](#)
- [subProtocol](#)
- [subProtocol](#)
- [subProtocol](#)
- [subProtocol](#)
- [subaggregateID_Offset](#) *Returns an offset from the beginning of variable length data to a particular subaggregate ID*
- [subcategory](#)
- [subtractInterestInMcastAddr](#)
- [subtractInterestInPduKind](#)
- [supplementalFuelStatus](#)
- [supplementalFuelStatus](#)
- **supplyQuantity :**
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [supplyQuantity](#)
- **supplyingId :**
 - in [DtResupplyCancelPdu](#)
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [sweepSync](#)
- [sweepSync](#)
- [synchronizationPointRegistrationFailed](#)
- [synchronizationPointRegistrationSucceeded](#) //////////////////////////////////////
- [systemDataWords](#)
- [systemDataWords](#)
- [systemFunction](#)
- [systemFunction](#)
- [systemID](#)
- [systemMode](#)

- [systemName](#)
- [systemName](#)
- [systemName](#)
- [systemSpecificDataLength](#)
- [systemSpecificOffset](#)
- [systemStatusBit](#)
- [systemStatus](#)
- [systemType](#)
- [systemType](#)
- [targetAddr](#)
- [targetAddr](#)
- [targetEntityID_Offset](#)
- [targetEntityID_Offset](#)
- [targetId](#)
- [targetId](#)
- [tdlType](#)
- [tentDeployed](#)
- [terminatePacketServer](#)
- [testClients](#)
- [thermalContrastRecordOffset](#) *Returns a buffer offset to the thermal contrast records*
- [theta](#)
- [theta](#)
- [theta](#)
- [theta](#)
- [theta](#)
- [theta](#)
- [theta](#)
- [threshMode](#)
- [thresolder](#) *Returns the SR's threshold, which contains thresholds needed for encoding*
- [tick](#) *Tick the publisher*
- [tick](#) *This function should be called once per frame for each entity*
- **tick :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [timeAdvanceGrant](#)
- [timeConstrainedEnabled](#)
- [time](#)
- [time](#)
- [time](#) *Get the time contained in the timestamp (in seconds)*

- [time](#)
- [timeForSets](#)
- [timeHopInUse](#)
- [timeInterval](#)
- [timeInterval](#)
- [timeRegulationEnabled](#) //////////////////////////////////
- [timeScale](#)
- [timeSet](#)
- [timeStamp](#) *Return the time (in seconds past an hour) encoded in the PDU header's time stamp*
- [timeStampType](#)
- [timeStampType](#)
- [timeStampType](#)
- [timeThreshold](#)
- [timeThreshold](#)
- [timedRecv](#)
- [timeout](#)
- [timeoutInterval](#)
- [topoUpOffset](#)
- [totalValues](#)
- **trackId :**
 - in [DtMimicTrackVCPdu](#)
 - in [DtTetherTrackVCPdu](#)
 - in [DtTrackVCPdu](#)
- [trackJamTarget](#)
- [trackJamTarget](#)
- [trackJamTargets](#) *Get set of track jam targets*
- [trackingAllTargets](#)
- [trackingAllTargets](#)
- [trailState](#)
- [transEntityID](#)
- [transferType](#)
- [transferredEntityID](#)
- [translationThreshold](#)
- [translationThreshold](#)
- [translations](#) *Get the translations*
- [transmitEntityId](#)
- [transmitLineState](#)
- [transmitPriority](#)
- [transmitRadioid](#)

- [illegible]

- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record (for 8-bit fields)*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/unload data from a packet record*
- [unloadNetRecord](#) *Load/unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecord](#) *Load/Unload data from a packet record*
- [unloadNetRecordSet](#) *Load/unload data from a packet record*
- [up](#)
- [updateNumber](#)
- [updateNumber](#)
- [updateNumber](#)
- [useDeadReckoner](#) *Create and use a new (private) dead-reckoner as our approximator*
- [useSmoother](#) *Create and use a new (private) smoother as our approximator*
- [userData](#)
- [usingDeadReckoner](#) *If the ESR is using a dead reckoner, returns it*
- [usingSmoother](#) *If the ESR is using a smoother, returns it*
- [usr](#) *Get the user data*
- [utmToGeocMatrix](#) *Get the matrix which transforms from UTM to geocentric*
- [val](#) *Get the value for the specified parameter type*
- [validateRecordSets](#) *This function checks the data records for consistency between the actual size of the PDU and the size suggested by the PDU*
- [varDatumId](#)
- [varDatumId](#)
- [varDatumId](#) *Returns the datum param found in the index'th variable datum record*

- [varDatumId](#)
- [varDatumId](#) Returns the datum param found in the index'th variable datum record
- [varDatumSizeBits](#)
- [varDatumSizeBytes](#)
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [var](#) Returns a pointer to the first byte of variable length data
- [variableDataLengthAfterReset](#) Returns the length of the variable data section in the PDU after the variable data has been reset
- [variableDataLength](#) Returns the length of the variable data section in the PDU
- [variableDatumRecordConstRef](#) Get a constant reference to a variable datum record
- [variableDatumRecordConstRef](#) Get a constant reference to a variable datum record
- [variableDatumRecordRef](#) Get a reference to a variable datum record
- [vecTrans](#)
- [vectorDim](#)
- [velocity](#)
- [velocity](#)
- [velocity](#)
- [velocity](#)
- [velocity](#)

- [velocity](#)
- [velocity](#)
- [velocity](#)
- [velocity](#)
- [velocity](#)
- [velocity](#)
- [velocity](#)
- [velocityOffset](#)
- [velocityOffset](#)
- [velocityOffset](#)
- [vertDeflection](#)
- [vertexRecordOffset](#) *Returns a buffer offset to the vertex records*
- [verticalDeviation](#)
- [verticalDeviation](#)
- [verticalError](#)
- [viewerCommand](#)
- [viewerId](#)
- [visualAppearanceOffset](#)
- [vrlinkVersion](#) *Returns a string indicating the VR-Link version number*
- [warheadType](#)
- [warheadType](#)
- [waterBurialDepthRecordOffset](#) *Returns a buffer offset to the water burial depth records*
- [waterStatus](#)
- [wavelength](#)
- [wavelength](#)
- [worldAntennaLoc](#)
- [worldAntennaLoc](#)
- [worldLocation](#)
- [worldLocation](#)
- [worldLocation](#)
- [wrapNext](#)
- [wrapNext](#) *get the next and previous elements, wrap at the ends of the list*
- [wrapNext](#) *Get the next remote emitterSystem*
- [wrapNext](#) *get the next and previous elements, wrap at the ends of the list*
- [wrapNextObj](#) *Get the next remote object*
- [wrapPrev](#)
- [wrapPrev](#)
- [wrapPrev](#) *Get the previous remote emitterSystem*

- [wrapPrev](#)
- [wrapPrevObj](#) *Get the previous remote object*
- [writeFd](#)
- [writeFd](#)
- [writeFd](#)
- [writeFd](#)
- [writeFd](#)
- [x](#)
- [x](#)
- [x](#)
- [xi](#)
- [y](#)
- [y](#)
- [y](#)
- [z](#)
- [z](#)
- [zero](#)
- [zero](#) *Get a zero'ed vector*
- [zeroPduInterestList](#)
- [~DtAbsNotifyObserver](#) *destructor*
- [~DtAcknowledgeRPdu](#) *destructor*
- [~DtActionRequestRPdu](#) *destructor*
- [~DtActionResponseRPdu](#) *destructor*
- [~DtAggregatePublisher](#) *destructor*
- [~DtAggregateStateDecoder](#) *destructor*
- [~DtAggregateStateEncoder](#) *destructor*
- [~DtAggregateStatePdu](#) *destructor*
- [~DtAggregateStateRepository](#) *destructor*
- [~DtAntennaPatternRecord](#) *destructor*
- [~DtApproximator](#) *destructor*
- [~DtArealObjectStatePdu](#) *destructor*
- [~DtArtPart](#) *destructor*
- [~DtArtPartList](#) *destructor*
- [~DtArtPartThresholder](#) *Destructor*
- [~DtAttachedPartList](#) *destructor*
- [~DtBaseEntityStateRepository](#) *destructor*
- [~DtBaseExerciseConn](#) *destructor*
- [~DtBaseHashKey](#) *destructor*

- [~DtBaseMsg](#) *Destructor*
- [~DtBaseReflectedAggregate](#) *destructor*
- [~DtBaseReflectedEntity](#) *destructor*
- [~DtBasicFixedWingAircraftRecord](#) *Destructor*
- [~DtBasicGroundCombatSoldierRecord](#) *Destructor*
- [~DtBasicGroundCombatVehicleRecord](#) *Destructor*
- [~DtBasicRotorWingAircraftRecord](#) *Destructor*
- [~DtBeamAntennaPatternRecord](#) *destructor*
- [~DtCartesianView](#) *destructor*
- [~DtClientSocket](#)
- [~DtCollisionElasticPdu](#) *destructor*
- [~DtCommentRPdu](#) *destructor*
- [~DtCreateEntityRPdu](#) *destructor*
- [~DtCsFilterTest](#)
- [~DtDataQueryRPdu](#) *destructor*
- [~DtDataRPdu](#) *destructor*
- [~DtDatumRecordSet](#) *Destructor*
- [~DtDeadReckoner](#) *destructor*
- [~DtDesignatorRepository](#) *destructor*
- [~DtDisPduFactory](#) *destructor*
- [~DtEmbeddedSystemRepository](#) *destructor*
- [~DtEmitterBeamRepository](#) *destructor*
- [~DtEmitterSystemRepository](#) *destructor*
- [~DtEnhancedFixedWingAircraftRecord](#) *Destructor*
- [~DtEnhancedGroundCombatSoldierRecord](#) *Destructor*
- [~DtEnhancedGroundCombatVehicleRecord](#) *Destructor*
- [~DtEnhancedRotorWingAircraftRecord](#) *Destructor*
- [~DtEntityPublisher](#) *Destructors*
- [~DtEntityStateDecoder](#) *destructor*
- [~DtEntityStateRepository](#) *destructor*
- [~DtEntityStateUpdatePdu](#) *destructor*
- [~DtEnvironmentTypeRecord](#) *destructor*
- [~DtEnvironmentalProcessPdu](#) *destructor*
- [~DtEventReptRPdu](#) *destructor*
- [~DtExerciseConn](#) *Destructor*
- [~DtFedReader](#) *Destructor*
- [~DtFedTarget](#) *Destructor*
- [~DtFilename](#) *destructor*

- [~DtFilterTest](#)
- [~DtGeodeticCoord](#) *destructor*
- [~DtGlobalIdHashlist](#) *destructor*
- [~DtGlobalObjectDesignatorList](#) *destructor*
- [~DtGridAxisRecordFixed](#) *Destructor*
- [~DtGridAxisRecordVariable](#) *Destructor*
- [~DtGridDataRecord](#) *Destructor*
- [~DtGridDataRecordType0](#) *Destructor*
- [~DtGridDataRecordType1](#) *Destructor*
- [~DtGridDataRecordType2](#) *Destructor*
- [~DtGriddedDataPdu](#) *destructor*
- [~DtGroundLogisticsVehicleRecord](#) *Destructor*
- [~DtGroupSocket](#) *Destructor*
- [~DtHashItem](#) *destructor*
- [~DtHashlist](#) *Destructor*
- [~DtIffAtcNavAidsBeamData](#) *Destructor*
- [~DtIffAtcNavAidsLayerHeader](#) *Destructor*
- [~DtIffAtcNavAidsOperational](#) *Destructor*
- [~DtIffAtcNavAidsParameter](#) *Destructor*
- [~DtIffAtcNavAidsPdu](#) *destructor*
- [~DtIffAtcNavAidsSecondOperational](#) *Destructor*
- [~DtIffAtcNavAidsSystemID](#) *Destructor*
- [~DtIntHashKey](#) *destructor*
- [~DtInteraction](#) *destructor*
- [~DtIntercomControlPdu](#) *destructor*
- [~DtIntercomEntityDestRec](#) *Destructor*
- [~DtIntercomGroupAsgmtRec](#) *Destructor*
- [~DtIntercomGroupDestRec](#) *Destructor*
- [~DtIntercomParmRec](#) *Destructor*
- [~DtIntercomSignalPdu](#) *destructor*
- [~DtIsGroupOfPdu](#) *destructor*
- [~DtIsPartOfPdu](#) *destructor*
- [~DtLeAppearancePdu](#) *destructor*
- [~DtLeArticulatedPartsPdu](#) *destructor*
- [~DtLeDeadReckoningParameters](#) *Destructor*
- [~DtLeDetonationPdu](#) *destructor*
- [~DtLeEntityID](#) *Destructor*
- [~DtLeEntityLinearVelocity](#) *Destructor*

- [~DtLeEntityLocationEntityCoordinates](#) *Destructor*
- [~DtLeEntityLocationWorldCoordinates](#) *Destructor*
- [~DtLeEntityOrientation](#) *Destructor*
- [~DtLeFirePdu](#) *destructor*
- [~DtLeOrientationError](#) *Destructor*
- [~DtLePositionError](#) *Destructor*
- [~DtLeTspiPdu](#) *destructor*
- [~DtLgrControlPdu](#) *destructor*
- [~DtLgrEndPdu](#) *destructor*
- [~DtLgrEofActionPdu](#) *destructor*
- [~DtLgrHeartbeatPdu](#) *destructor*
- [~DtLgrPausePdu](#) *destructor*
- [~DtLgrPbFilePdu](#) *destructor*
- [~DtLgrPlayActionPdu](#) *destructor*
- [~DtLgrPlayPdu](#) *destructor*
- [~DtLgrQuitPdu](#) *destructor*
- [~DtLgrRecFilePdu](#) *destructor*
- [~DtLgrRecordActionPdu](#) *destructor*
- [~DtLgrRecordPdu](#) *destructor*
- [~DtLgrSetTimePdu](#) *destructor*
- [~DtLgrSkipTimePdu](#) *destructor*
- [~DtLgrStartPdu](#) *destructor*
- [~DtLgrStopPdu](#) *destructor*
- [~DtLgrTimeScalePdu](#) *destructor*
- [~DtLgrTimeoutPdu](#) *destructor*
- [~DtLinearObjectStatePdu](#) *destructor*
- [~DtLinearSegmentParameter](#) *Destructor*
- [~DtList](#) *Destructor - does NOT free memory of data associated with the items*
- [~DtListItem](#) *Destructor*
- [~DtListedObj](#) *Destructor*
- [~DtMinefieldDataPdu](#) *destructor*
- [~DtMinefieldPerimeterRecord](#) *Destructor*
- [~DtMinefieldQueryPdu](#) *destructor*
- [~DtMinefieldResponseNackPdu](#) *destructor*
- [~DtMinefieldStatePdu](#) *destructor*
- [~DtNetSocket](#)
- [~DtNetTimeStamp](#) *destructor*
- [~DtNullFederateAmbassador](#)

- [~DtObjectIdHashKey](#) destructor
- [~DtObjectIdHashlist](#) destructor
- [~DtObjectType](#) Destructor
- [~DtPduCallbackInfo](#) destructor
- [~DtPduCallbackManager](#) destructor
- [~DtPduDatumRecordSet](#) Destructor
- [~DtPdu](#) Destructor
- [~DtPduFactory](#) destructor
- [~DtPduWithData](#) Destructor
- [~DtPointObjectStatePdu](#) destructor
- [~DtPropulsionSystem](#) Destructor
- [~DtRadioReceiverRepository](#) destructor
- [~DtRadioTransmitterRepository](#) destructor
- [~DtRecordPdu](#) destructor
- [~DtRecordQueryPdu](#) destructor
- [~DtRecordSet](#) Destructor
- [~DtRecordSetPdu](#) destructor
- [~DtReflectedAggregate](#) Destructor
- [~DtReflectedAggregateList](#) Destructor
- [~DtReflectedEmitterSystem](#) destructor
- [~DtReflectedEmitterSystemList](#) Destructor
- [~DtReflectedEntity](#) Destructor
- [~DtReflectedEntityList](#) Destructor
- [~DtReflectedObjectCallbackInfo](#) destructor
- [~DtReflectedObject](#) destructor
- [~DtReflectedObjectList](#) destructor Warning: makes non-virtual call to removeAndDelete
- [~DtRemoveEntityRPdu](#) destructor
- [~DtScript](#)
- [~DtSetDataRPdu](#) destructor
- [~DtSetRecordPdu](#) destructor
- [~DtSilentAggregateList](#) Destructor
- [~DtSilentEntityList](#) Destructor
- [~DtSmoothOriWithDr](#) destructor
- [~DtSmoothPos](#)
- [~DtSmoothPosWithDr](#) destructor
- [~DtSmoother](#) destructor
- [~DtSocket](#)
- [~DtStartResumeRPdu](#) destructor

- [~DtStateMsg](#) *destructor*
- [~DtStateRepository](#) *destructor*
- [~DtStopFreezeRPdu](#) *destructor*
- [~DtString](#) *destructor*
- [~DtStringHashKey](#) *destructor*
- [~DtSuppEmisEntStatePdu](#) *destructor*
- [~DtSystemHashKey](#) *destructor*
- [~DtTcpForwarder](#)
- [~DtTcpSocket](#) *Destructor*
- [~DtTcpSocketMgr](#)
- [~DtThresholder](#) *destructor*
- [~DtTopoView](#) *destructor*
- [~DtTrackJamDesignator](#) *destructor*
- [~DtTrackJamDesignatorList](#) *destructor*
- [~DtTransferControlRequestPdu](#) *destructor*
- [~DtUnderwaterAcousticAddtnlPassiveAct](#) *Destructor*
- [~DtUnderwaterAcousticBeam](#) *Destructor*
- [~DtUnderwaterAcousticEmitterSystem](#) *Destructor*
- [~DtUnderwaterAcousticPdu](#) *destructor*
- [~DtUnderwaterAcousticShaftSpeed](#) *Destructor*
- [~DtUtmCoord](#) *destructor*
- [~DtUtmView](#) *destructor*
- [~DtVariableDatumRecord](#) *destructor*
- [~DtVector](#) *destructor*
- [~DtVectoringNozzleSystem](#) *Destructor*

Variables

- [BeamFuncJammer](#)
- [Detail](#)
- [DetailedModulationUnmodulated](#)
- [Dt105m](#)
- [Dt120mm](#)
- [Dt25mm](#)
- [Dt7_62mm](#)
- [Dt7_62mmUK](#)
- [Dt7_62mmUSSR](#)
- [DtA10](#)
- [DtA2ATD_CAC2](#)

- [DtA2ATD_SINGARS_ERF](#)
- [DtAA3Anan](#)
- [DtAA9Amos](#)
- [DtADPCM_per_CCITT_G721](#)
- [DtAGM88HARM](#)
- [DtAGTS](#)
- [DtAH1W](#)
- [DtAH64](#)
- [DtAMRAAM](#)
- [DtAN_PSS_11](#)
- [DtAN_PSS_12](#)
- [DtASCII_CharacterSet](#)
- [DtASTAMIDS](#)
- [DtASTAMIDS_I](#)
- [DtASTAMIDS_II](#)
- [DtAbbreviated_Command_And_Control](#)
- [DtAckCreate](#)
- [DtAckRemove](#)
- [DtAckStart](#)
- [DtAckStop](#)
- [DtAcknowledgeInteraction](#)
- [DtAcknowledgePduKind](#)
- [DtAcknowledgeRPduKind](#)
- [DtAcknowledged](#)
- [DtActionAcceptSubscription](#)
- [DtActionChangeAggregateState](#)
- [DtActionDamage](#)
- [DtActionDataQuery](#)
- [DtActionDismount](#)
- [DtActionFireDisabled](#)
- [DtActionInitializeInternalParameters](#)
- [DtActionInitiateHitchFollow](#)
- [DtActionInitiateHitchLead](#)
- [DtActionInitiateServiceStationResupply](#)
- [DtActionInitiateTailgateResupply](#)
- [DtActionInitiateTetherFollow](#)
- [DtActionInitiateTetherLead](#)
- [DtActionJoinExercise](#)

- [DtActionKilledInAction](#)
- [DtActionLocalStorage](#)
- [DtActionLockSiteConfiguration](#)
- [DtActionMobilityDisabled](#)
- [DtActionMount](#)
- [DtActionMountIntent](#)
- [DtActionOther2](#)
- [DtActionOther](#)
- [DtActionOutOfAmmo](#)
- [DtActionOutOfFuel](#)
- [DtActionQuerySiteConfiguration](#)
- [DtActionRecallCheckpointData](#)
- [DtActionRecallInitialParameters](#)
- [DtActionReconstitute](#)
- [DtActionRequestInteraction](#)
- [DtActionRequestPduKind](#)
- [DtActionRequestRPduKind](#)
- [DtActionRequestStartPDU](#)
- [DtActionResignExercise](#)
- [DtActionResponseInteraction](#)
- [DtActionResponsePduKind](#)
- [DtActionResponseRPduKind](#)
- [DtActionRun](#)
- [DtActionSaveInternalParameters](#)
- [DtActionSendObjectStateData](#)
- [DtActionSendPlanData](#)
- [DtActionSimulateMalfunction](#)
- [DtActionStartDRC](#)
- [DtActionStatusRequest](#)
- [DtActionStopDRC](#)
- [DtActionSynchronizeInternalClocks](#)
- [DtActionTeleportEntity](#)
- [DtActionTetheringInformation](#)
- [DtActionTimeAdvance](#)
- [DtActionUnhitch](#)
- [DtActionUnlockSiteConfiguration](#)
- [DtActionUnsubscribe](#)
- [DtActionUntether](#)

- [DtActionUpdateSiteConfiguration](#)
- [DtActionWakeUpGetReadyForInitialization](#)
- [DtActuatorCatapults](#)
- [DtActuatorCoolingSys](#)
- [DtActuatorCranes](#)
- [DtActuatorLaunchers](#)
- [DtActuatorWaterSupply](#)
- [DtActuatorWinches](#)
- [DtAfsk](#)
- [DtAftTorpedoTube](#)
- [DtAggregateStatePduKind](#)
- [DtAggregated](#)
- [DtAircraftWingstation](#)
- [DtAm](#)
- [DtAmmunitionShip](#)
- [DtAmmunitionSupplyCap](#)
- [DtAmphibiousAssaultShip](#)
- [DtAmphibiousCargoShip](#)
- [DtAmphibiousTransportDock](#)
- [DtAmplitudeAndAngle](#)
- [DtAmplitudeAnglePulse](#)
- [DtAmplitude](#)
- [DtAngle](#)
- [DtAntiAir](#)
- [DtAntiArmor](#)
- [DtAntiArmorGuided](#)
- [DtAntiArmorUnguided](#)
- [DtAntiGuidedMunition](#)
- [DtAntiHandlingDevice](#)
- [DtAntiPersonnel](#)
- [DtAntiPersonnelGuided](#)
- [DtAntiPersonnelUnguided](#)
- [DtAntiRadar](#)
- [DtAntiRadarGuided](#)
- [DtAntiRadarUnguided](#)
- [DtAntiSatellite](#)
- [DtAntiShip](#)
- [DtAntiShipGuided](#)

- [DtAntiShipUnguided](#)
- [DtAntiSubmarine](#)
- [DtAntiSubmarineGuided](#)
- [DtAntiSubmarinePatrol](#)
- [DtAntiSubmarineUnguided](#)
- [DtAnyExercise](#)
- [DtApAzimuth](#)
- [DtApAzimuthRate](#)
- [DtApElevation](#)
- [DtApElevationRate](#)
- [DtApExtension](#)
- [DtApExtensionRate](#)
- [DtApPosition](#)
- [DtApPositionRate](#)
- [DtApRotation](#)
- [DtApRotationRate](#)
- [DtApX](#)
- [DtApXRate](#)
- [DtApY](#)
- [DtApYRate](#)
- [DtApZ](#)
- [DtApZRate](#)
- [DtAppDamageBitShift](#)
- [DtAppDamage](#)
- [DtAppDummyBitShift](#)
- [DtAppDummy](#)
- [DtAppFinalPduBitShift](#)
- [DtAppFinalPdu](#)
- [DtAppFirePowerKillBitShift](#)
- [DtAppFirePowerKill](#)
- [DtAppFlamingBitShift](#)
- [DtAppFlaming](#)
- [DtAppFrozenBitShift](#)
- [DtAppFrozen](#)
- [DtAppHatchBitShift](#)
- [DtAppHatch](#)
- [DtAppLightsBitShift](#)
- [DtAppLights](#)

- [DtAppMobilityKillBitShift](#)
- [DtAppMobilityKill](#)
- [DtAppPaintSchemeBitShift](#)
- [DtAppPaintScheme](#)
- [DtAppPowerPlantStatusBitShift](#)
- [DtAppPowerPlantStatus](#)
- [DtAppReqTransferEntity](#)
- [DtAppReqTransferEnvProc](#)
- [DtAppSmokeBitShift](#)
- [DtAppSmoke](#)
- [DtAppTrailingEffectsBitShift](#)
- [DtAppTrailingEffects](#)
- [DtAppearInvalid](#)
- [DtApplicationSpecific](#)
- [DtArealObjectStatePduKind](#)
- [DtArmoredFightingVehicle](#)
- [DtArmoredUtilityVehicle](#)
- [DtArmy](#)
- [DtArtPartAggregateThresh](#)
- [DtArtPartDfltThreshMode](#)
- [DtArtPartIndividualThresh](#)
- [DtArtPartSpec](#)
- [DtArticulatedParts](#)
- [DtAssemblyFormation](#)
- [DtAttachedPartSuccess](#)
- [DtAttachedTo](#)
- [DtAttachedToBase](#)
- [DtAttack](#)
- [DtAttackHelicopter](#)
- [DtAttributeChangeResultOther](#)
- [DtAuxCraftEjectionSeat](#)
- [DtAuxCraftLandingCraft](#)
- [DtAuxCraftLifeBoat](#)
- [DtBGM109Tomahawk](#)
- [DtBGM71TOW](#)
- [DtBMP1](#)
- [DtBMP2](#)
- [DtBRDM1](#)

- [DtBadBeamId](#)
- [DtBasebandEncryption](#)
- [DtBasicF16](#)
- [DtBasicFixedWingAircraft](#)
- [DtBasicGroundCombatSoldier](#)
- [DtBasicGroundCombatVehicle](#)
- [DtBasicRotorWingAircraft](#)
- [DtBasicT72](#)
- [DtBattalion](#)
- [DtBattle_Command](#)
- [DtBattlefield105mm](#)
- [DtBattlefield105mmUK](#)
- [DtBattlefield120mm](#)
- [DtBattlefield20mm](#)
- [DtBattlefield30mm](#)
- [DtBattlefieldSupport](#)
- [DtBattlefieldSupportGuided](#)
- [DtBattlefieldSupportUnguided](#)
- [DtBattleship](#)
- [DtBeam](#)
- [DtBeamFuncAcquisitionAndTracking](#)
- [DtBeamFuncAcquisition](#)
- [DtBeamFuncActiveRadarMissileSeeker](#)
- [DtBeamFuncCommandGuidance](#)
- [DtBeamFuncHeightFinder](#)
- [DtBeamFuncIllumination](#)
- [DtBeamFuncMissileBeacon](#)
- [DtBeamFuncMissileFuze](#)
- [DtBeamFuncOther](#)
- [DtBeamFuncRangeOnlyRadar](#)
- [DtBeamFuncSearch](#)
- [DtBeamFuncTracking](#)
- [DtBelleauWood](#)
- [DtBest_Effort](#)
- [DtBinoculars](#)
- [DtBlackPaintScheme](#)
- [DtBluePaintScheme](#)
- [DtBodyAirframe](#)

- [DtBodyHull](#)
- [DtBodyRates](#) *Angular rotation about body X, Y, Z axes*
- [DtBombBay](#)
- [DtBomber](#)
- [DtBridgeConcreteTwoLane](#)
- [DtBrownPaintScheme](#)
- [DtBuildingFourStory](#)
- [DtBuildingOneStory](#)
- [DtBuildingThreeStory](#)
- [DtBuildingTwoStory](#)
- [DtC130](#)
- [DtCCSIL](#)
- [DtCCTT_SINGARS_DATA](#)
- [DtCCTT_SINGARS_ERF_HOPSET](#)
- [DtCCTT_SINGARS_ERF_LOCKOUT](#)
- [DtCCTT_SINGARS_OTAR](#)
- [DtCOBRA3_5](#)
- [DtCOBRA8_12](#)
- [DtCONUSNAD27](#)
- [DtCV59](#)
- [DtCVSD_per MIL STD 188 113](#)
- [DtCamouflageDesertPaintScheme](#)
- [DtCamouflageGravelPaintScheme](#)
- [DtCamouflageJunglePaintScheme](#)
- [DtCamouflagePavementPaintScheme](#)
- [DtCamouflageSandPaintScheme](#)
- [DtCamouflageSnowPaintScheme](#)
- [DtCargoBay](#)
- [DtCargo](#)
- [DtCarrier](#)
- [DtCategory](#)
- [DtCategory](#)
- [DtCategory](#)
- [DtChallengerMBT](#)
- [DtChange](#)
- [DtChangeDataUpdate](#)
- [DtCharacterSet](#)
- [DtCivilianCar](#)

- [DtClearPaintScheme](#)
- [DtCloud](#)
- [DtCloudWithRainFalling](#)
- [DtCloudWithSnowFalling](#)
- [DtCoPilot](#)
- [DtCollisionElasticPduKind](#)
- [DtCollisionInteraction](#)
- [DtCollisionPduKind](#)
- [DtCollisionStaticSize](#)
- [DtCollisionTypeOther](#)
- [DtColumnFormation](#)
- [DtCombatStoresShip](#)
- [DtCombination](#)
- [DtCommandFuse](#)
- [DtCommandShip](#)
- [DtCommander](#)
- [DtCommentInteraction](#)
- [DtCommentPduKind](#)
- [DtCommentRPduKind](#)
- [DtCompany](#)
- [DtComputerGeneratedForcesCarriedAsCargo](#)
- [DtConstDcm](#)
- [DtConstVector](#)
- [DtContinuous](#)
- [DtCorps](#)
- [DtCouldNotOpenFed](#)
- [DtCountry](#)
- [DtCountryCodeOther](#)
- [DtCountry](#)
- [DtCountry](#)
- [DtCreateEntityInteraction](#)
- [DtCreateEntityPduKind](#)
- [DtCreateEntityRPduKind](#)
- [DtCreateObjectResultOther](#)
- [DtCtlReqTransferEntity](#)
- [DtCtlReqTransferEnvProc](#)
- [DtCulturalFeatureDomainAir](#)
- [DtCulturalFeatureDomainLand](#)

- [DtCulturalFeatureDomainOther](#)
- [DtCulturalFeatureDomainSpace](#)
- [DtCulturalFeatureDomainSubsurface](#)
- [DtCulturalFeatureDomainSurface](#)
- [DtCulturalFeature](#)
- [DtCw](#)
- [DtD20](#)
- [DtDamageDestroyed](#)
- [DtDamageModerate](#)
- [DtDamageNone](#)
- [DtDamageSlight](#)
- [DtDataInteraction](#)
- [DtDataPduKind](#)
- [DtDataQueryInteraction](#)
- [DtDataQueryPduKind](#)
- [DtDataQueryRPduKind](#)
- [DtDataRPduKind](#)
- [DtDatumAltCategory](#)
- [DtDatumAltCountry](#)
- [DtDatumAltDomain](#)
- [DtDatumAltExtra](#)
- [DtDatumAltKind](#)
- [DtDatumAltSpecific](#)
- [DtDatumAltSubCategory](#)
- [DtDatumAmmoQuantity](#)
- [DtDatumAmountAmmoSupplies](#)
- [DtDatumAppearance](#)
- [DtDatumArtParamId](#)
- [DtDatumArtParamType](#)
- [DtDatumArtParamValue](#)
- [DtDatumBadId](#)
- [DtDatumBadIndex](#)
- [DtDatumBadLength](#)
- [DtDatumBodyPartDamagedRatio](#)
- [DtDatumCapability](#)
- [DtDatumCategory](#)
- [DtDatumCountry](#)
- [DtDatumDomain](#)

- [DtDatumDrAlgorithm](#)
- [DtDatumExtra](#)
- [DtDatumForceId](#)
- [DtDatumFuelQuantity](#)
- [DtDatumKind](#)
- [DtDatumLoggerFile](#)
- [DtDatumMarkingCS](#)
- [DtDatumMarking](#)
- [DtDatumMass](#)
- [DtDatumMovingRadioTime](#)
- [DtDatumNumArtParam](#)
- [DtDatumPhi](#)
- [DtDatumPsi](#)
- [DtDatumRadarSysStatus](#)
- [DtDatumRadioCommSysStatus](#)
- [DtDatumSpecific](#)
- [DtDatumStationaryRadioTime](#)
- [DtDatumSubCategory](#)
- [DtDatumTerrainDatabase](#)
- [DtDatumTheta](#)
- [DtDatumValid](#)
- [DtDatumXaccel](#)
- [DtDatumXangvel](#)
- [DtDatumXpos](#)
- [DtDatumXvel](#)
- [DtDatumYaccel](#)
- [DtDatumYangvel](#)
- [DtDatumYpos](#)
- [DtDatumYvel](#)
- [DtDatumZaccel](#)
- [DtDatumZangvel](#)
- [DtDatumZpos](#)
- [DtDatumZvel](#)
- [DtDcmRef](#)
- [DtDeferredEntityId](#)
- [DtDegMinSec](#)
- [DtDensityClear](#)
- [DtDensityDense](#)

- [DtDensityHazy](#)
- [DtDensityOpaque](#)
- [DtDensityVeryDense](#)
- [DtDesertCamouflage](#)
- [DtDesignatorCodeOther](#)
- [DtDesignatorNameOther](#)
- [DtDesignatorNameTBD](#)
- [DtDesignatorPduKind](#)
- [DtDestroyer](#)
- [DtDestroyerTender](#)
- [DtDetResAirBurst](#)
- [DtDetResAirHit](#)
- [DtDetResArmorPiercingHit](#)
- [DtDetResBuildingHitLarge](#)
- [DtDetResBuildingHitMedium](#)
- [DtDetResBuildingHitSmall](#)
- [DtDetResDetonation](#)
- [DtDetResDirtBlastLarge](#)
- [DtDetResDirtBlastMedium](#)
- [DtDetResDirtBlastSmall](#)
- [DtDetResEntityImpact](#)
- [DtDetResEntityProximate](#)
- [DtDetResEnvironmentObjectImpact](#)
- [DtDetResEnvironmentObjectProximate](#)
- [DtDetResGroundImpact](#)
- [DtDetResGroundProximate](#)
- [DtDetResHE_HitLarge](#)
- [DtDetResHE_HitMedium](#)
- [DtDetResHE_HitSmall](#)
- [DtDetResMineClearingLineCharge](#)
- [DtDetResNone](#)
- [DtDetResOther](#)
- [DtDetResWaterBlastLarge](#)
- [DtDetResWaterBlastMedium](#)
- [DtDetResWaterBlastSmall](#)
- [DtDetResWaterImpact](#)
- [DtDetailedAmplitudeAndAngle](#)
- [DtDetailedModulationDummy](#)

- [DtDetonationInteraction](#)
- [DtDetonationPduKind](#)
- [DtDigitalDataDevice](#)
- [DtDipphaseEncryption](#)
- [DtDisaggregated](#)
- [DtDismountedInfantry](#)
- [DtDivision](#)
- [DtDockLandingShip](#)
- [DtDomain](#)
- [DtDomain](#) *this conflicts with enum of same name, so must be before funcs that use it*
- [DtDomain](#)
- [DtDrDrmFpb](#)
- [DtDrDrmFpw](#)
- [DtDrDrmFvb](#)
- [DtDrDrmFvw](#)
- [DtDrDrmRpb](#)
- [DtDrDrmRpw](#)
- [DtDrDrmRvb](#)
- [DtDrDrmRvw](#)
- [DtDrOther](#)
- [DtDrStatic](#)
- [DtDragonsTeeth](#)
- [DtDriveTrainAlternator](#)
- [DtDriveTrainBattery](#)
- [DtDriveTrainBrakes](#)
- [DtDriveTrainEngineCoolantLeak](#)
- [DtDriveTrainEngineOilLeak](#)
- [DtDriveTrainFilters](#)
- [DtDriveTrainFuelFilter](#)
- [DtDriveTrainGenerator](#)
- [DtDriveTrainMotor](#)
- [DtDriveTrainPumps](#)
- [DtDriveTrainStarter](#)
- [DtDriveTrainTransmissionOilLeak](#)
- [DtDriveTrainTransmission](#)
- [DtDriver](#)
- [DtDsb](#)
- [DtDumpFile](#)

- [DtDuplicateBeamId](#)
- [DtED50](#)
- [DtEParm](#) *Euler parameter routines*
- [DtEast](#)
- [DtElastic](#)
- [DtElectromagneticEmissionPduKind](#)
- [DtElectronicCoders](#)
- [DtElectronicComputers](#)
- [DtElectronicDecoders](#)
- [DtElectronicDetectionSys](#)
- [DtElectronicIntercoms](#)
- [DtElectronicLasers](#)
- [DtElectronicRadios](#)
- [DtElectronicRangeFinder](#)
- [DtElectronicSys](#)
- [DtElectronicWarfare](#)
- [DtElectronicWarfareSys](#)
- [DtEmissionFuncAcquisitionDetection](#)
- [DtEmissionFuncDecoy](#)
- [DtEmissionFuncEarlyWarningSurveillance](#)
- [DtEmissionFuncFireControl](#)
- [DtEmissionFuncFiringPointLaunchPointLocation](#)
- [DtEmissionFuncGuidanceIllumination](#)
- [DtEmissionFuncHeightFinding](#)
- [DtEmissionFuncIdentificationClassification](#)
- [DtEmissionFuncImaging](#)
- [DtEmissionFuncInstrumentation](#)
- [DtEmissionFuncJammingDeception](#)
- [DtEmissionFuncJammingNoise](#)
- [DtEmissionFuncMotionDetection](#)
- [DtEmissionFuncMultiFunction](#)
- [DtEmissionFuncNavigation](#)
- [DtEmissionFuncOther](#)
- [DtEmissionFuncRadarAltimeter](#)
- [DtEmissionFuncRanging](#)
- [DtEmissionFuncTracking](#)
- [DtEmissionFuncWeaponLethal](#)
- [DtEmissionFuncWeaponNonLethal](#)

- [DtEmissionFuncWeather](#)
- [DtEmitterMountedOnHost](#)
- [DtEmitterSuccess](#)
- [DtEmitterSysOther](#)
- [DtEncodedVoice](#)
- [DtEncodingTypeOther](#)
- [DtEncoding_16_bit_linear_PCM](#)
- [DtEncoding_8_bit_linear_PCM](#)
- [DtEncoding_8_bit_mu_law](#)
- [DtEnhancedFixedWingAircraft](#)
- [DtEnhancedGroundCombatSoldier](#)
- [DtEnhancedGroundCombatVehicle](#)
- [DtEnhancedRotorWingAircraft](#)
- [DtEntity](#)
- [DtEntityBase](#)
- [DtEntity](#)
- [DtEntity](#)
- [DtEntityKind](#)
- [DtEntityKind](#)
- [DtEntityKindOther](#)
- [DtEntityStatePduKind](#)
- [DtEntityStateUpdatePduKind](#)
- [DtEntityStationedAt](#)
- [DtEnvModelInvalid](#)
- [DtEnvironmentalDomainAir](#)
- [DtEnvironmentalDomainAirOther](#)
- [DtEnvironmentalDomainLand](#)
- [DtEnvironmentalDomainOther](#)
- [DtEnvironmentalDomainSpace](#)
- [DtEnvironmentalDomainSubsurface](#)
- [DtEnvironmentalDomainSurface](#)
- [DtEnvironmental](#)
- [DtEnvironmentalProcessPduKind](#)
- [DtEnvironmentalStatusLast](#)
- [DtEnvironmentalStatusOn](#)
- [DtErrorReadingFed](#)
- [DtEvent](#)
- [DtEventDamage](#)

- [DtEvent](#)
- [DtEvent](#)
- [DtEventID](#)
- [DtEventKilledInAction](#)
- [DtEventOther](#)
- [DtEventOutOfAmmo](#)
- [DtEventOutOfFuel](#)
- [DtEventReportInteraction](#)
- [DtEventReportPduKind](#)
- [DtEventReptRPduKind](#)
- [DtExerciseld](#)
- [DtExtra](#)
- [DtExtra](#)
- [DtExtra](#)
- [DtF117](#)
- [DtF15](#)
- [DtF15E](#)
- [DtF16C](#)
- [DtF16](#)
- [DtF18](#)
- [DtFFG7](#)
- [DtFLIR_SensorType](#)
- [DtFalse](#)
- [DtFamilyDistributedEmissionRegeneration](#)
- [DtFamilyEntityInteraction](#)
- [DtFamilyEntityManagement](#)
- [DtFamilyLiveEntity](#)
- [DtFamilyLogistics](#)
- [DtFamilyMinefield](#)
- [DtFamilyNonRealTime](#)
- [DtFamilyOther](#)
- [DtFamilyRadioCommunication](#)
- [DtFamilySimulationManagement](#)
- [DtFamilySimulationManagementWithReliability](#)
- [DtFamilySyntheticEnvironment](#)
- [DtFamilyWarfare](#)
- [DtFedReadOK](#)
- [DtFighter](#)

- [DtFireInteraction](#)
- [DtFirePduKind](#)
- [DtFireTeam](#)
- [DtFirstOfficer](#)
- [DtFixed](#)
- [DtFloat32](#)
- [DtFloat64](#)
- [DtFm](#)
- [DtFog](#)
- [DtForceFriendly](#)
- [DtForceNeutral](#)
- [DtForceOpposing](#)
- [DtForceOther](#)
- [DtForestCamouflage](#)
- [DtForwardTorpedoTube](#)
- [DtFrance](#)
- [DtFrequencyHoppingTimeHopping](#)
- [DtFrequencyHopping](#)
- [DtFrigate](#)
- [DtFsk](#)
- [DtFtInetAddr](#)
- [DtFtNetU8](#)
- [DtFuelCarriedAsCarge](#)
- [DtFuelGauges](#)
- [DtFuelLines](#)
- [DtFuelSupplyCap](#)
- [DtFuelTransferPump](#)
- [DtFullyDisaggregated](#)
- [DtFuze](#)
- [DtFuzeAcoustic](#)
- [DtFuzeAltitude](#)
- [DtFuzeCommand](#)
- [DtFuzeContactDelayed](#)
- [DtFuzeContact](#)
- [DtFuzeContactInstant](#)
- [DtFuzeDepth](#)
- [DtFuze](#)
- [DtFuze](#)

- [DtFuzeOther](#)
- [DtFuzeProximity](#)
- [DtFuzeTimed](#)
- [DtGAZ66](#)
- [DtGSTAMIDS3_5](#)
- [DtGSTAMIDS8_12](#)
- [DtGSTAMIDS](#)
- [DtGSTAMIDS_I](#)
- [DtGSTAMIDS_II](#)
- [DtGazelle](#)
- [DtGeneric3_5](#)
- [DtGeneric8_12](#)
- [DtGenericAntenna](#)
- [DtGeneric](#)
- [DtGenericGPR](#)
- [DtGenericPduKind](#)
- [DtGenericProbe](#)
- [DtGlobalIdHashList](#)
- [DtGlobalObjectDesignator](#)
- [DtGreenPaintScheme](#)
- [DtGridAxisFixed](#)
- [DtGridAxisVariable](#)
- [DtGridDataRecType0](#)
- [DtGridDataRecType1](#)
- [DtGridDataRecType2](#)
- [DtGriddedDataPduKind](#)
- [DtGroundLogisticsVehicle](#)
- [DtGuidedMissileCruiser](#)
- [DtGuidedMissileDestroyer](#)
- [DtGuidedMissileFrigate](#)
- [DtGunmountAttachedToHost](#)
- [DtGunner](#)
- [DtHMMWV](#)
- [DtHMMWV_OccActivelySearching](#)
- [DtHMMWV_OccNotActivelySearching](#)
- [DtHSTAMIDS3_5](#)
- [DtHSTAMIDS8_12](#)
- [DtHSTAMIDS_I](#)

- [DtHSTAMIDS_II](#)
- [DtHashList](#)
- [DtHatchClosed](#)
- [DtHatchNA](#)
- [DtHatchOpenAndPerson](#)
- [DtHatchOpen](#)
- [DtHatchPoppedAndPerson](#)
- [DtHatchPopped](#)
- [DtHawk](#)
- [DtHeartbeatMode](#)
- [DtHeavyAlgae](#)
- [DtHeavyCruiser](#)
- [DtHellfire](#)
- [DtHost](#)
- **DtHost :**
 - in [duplicate](#)
 - in [DtNetSimulationAddress](#)
- [DtHostFireableMunition](#)
- [DtHour2Sec](#)
- [DtHq](#)
- [DtHqIIA](#)
- [DtHqII](#)
- [DtIffAtcNavaidATCRBS](#)
- [DtIffAtcNavaidAltParm4Invalid](#)
- [DtIffAtcNavaidAltParm4NoResponse](#)
- [DtIffAtcNavaidAltParm4Other](#)
- [DtIffAtcNavaidAltParm4Valid](#)
- [DtIffAtcNavaidAlternateMode4](#)
- [DtIffAtcNavaidAlternateMode5C](#)
- [DtIffAtcNavaidAppModelInvalid](#)
- [DtIffAtcNavaidChangeIndicator](#)
- [DtIffAtcNavaidInterrogator](#)
- [DtIffAtcNavaidLayerInfoInvalid](#)
- [DtIffAtcNavaidMarkX](#)
- [DtIffAtcNavaidMarkXII](#)
- [DtIffAtcNavaidMarkX_XII_ATCRBS](#)
- [DtIffAtcNavaidMarkX_XII_ATCRBS_Modes](#)
- [DtIffAtcNavaidModelInvalid](#)
- [DtIffAtcNavaidModes](#)

- [DtIffAtcNavAidsModifierInvalid](#)
- [DtIffAtcNavAidsOperationalStatus](#)
- [DtIffAtcNavAidsOtherSysName](#)
- [DtIffAtcNavAidsOtherSysType](#)
- [DtIffAtcNavAidsParameter1](#)
- [DtIffAtcNavAidsParameter2](#)
- [DtIffAtcNavAidsParameter3](#)
- [DtIffAtcNavAidsParameter4](#)
- [DtIffAtcNavAidsParameter5](#)
- [DtIffAtcNavAidsParameter6](#)
- [DtIffAtcNavAidsParameterDamage](#)
- [DtIffAtcNavAidsParameterMalfunction](#)
- [DtIffAtcNavAidsParameterStatus](#)
- [DtIffAtcNavAidsParameterTCAS](#)
- [DtIffAtcNavAidsParm1Invalid](#)
- [DtIffAtcNavAidsParm2Invalid](#)
- [DtIffAtcNavAidsPduKind](#)
- [DtIffAtcNavAidsSoviet](#)
- [DtIffAtcNavAidsSovietInterrogator](#)
- [DtIffAtcNavAidsSovietTransponder](#)
- [DtIffAtcNavAidsSystemOnOff](#)
- [DtIffAtcNavAidsTransponder](#)
- [DtImageIntensifier](#)
- [DtInelastic](#)
- [DtInetAddr](#) ----- *inet utils* -----
- [DtInetAddrAny](#)
- [DtInetBroadcastMask](#)
- [DtInitErrorFatal](#)
- [DtInitErrorIgnore](#)
- [DtInitErrorWarn](#)
- [DtInsideOf](#)
- [DtInt16](#)
- [DtInt32](#)
- [DtInt8](#)
- [DtIntercomAck](#)
- [DtIntercomCmdStatus](#)
- [DtIntercomConnect](#)
- [DtIntercomControlPduKind](#)

- [DtIntercomCtlStatus](#)
- [DtIntercomDisconnect](#)
- [DtIntercom](#)
- [DtIntercomEntDestRec](#)
- [DtIntercomFDX](#)
- [DtIntercomGrpAsgmtRec](#)
- [DtIntercomGrpDestRec](#)
- [DtIntercomHDX](#)
- [DtIntercomHDX_RecOnly](#)
- [DtIntercomHDX_TransOnly](#)
- [DtIntercomLS_NoCmd](#)
- [DtIntercomNack](#)
- [DtIntercomNoCmd](#)
- [DtIntercomNotApplicable](#)
- [DtIntercomNotTransmitting](#)
- [DtIntercomOff](#)
- [DtIntercomOn](#)
- [DtIntercomReqAck](#)
- [DtIntercomReqNoAck](#)
- [DtIntercomReset](#)
- [DtIntercomRetLocLS_Ctl](#)
- [DtIntercomSetLS_NotTrans](#)
- [DtIntercomSetLS_Trans](#)
- [DtIntercomSignalPduKind](#)
- [DtIntercomSimCommChnl](#)
- [DtIntercomSimCommSupChnl](#)
- [DtIntercomTransmitting](#)
- [DtInterfaceFilters](#)
- [DtInterfacePropeller](#)
- [DtInternalArtParam](#)
- [DtIraq](#)
- [DtIsGroupOfPduKind](#)
- [DtIsPartOfPduKind](#)
- [DtIsb](#)
- [DtJamBeam](#)
- [DtKC130](#)
- [DtLE_AppearanceAlternativeType](#)
- [DtLE_AppearanceAudio](#)

- [DtLE_AppearanceCapabilities](#)
- [DtLE_AppearanceEM](#)
- [DtLE_AppearanceFlagField2](#)
- [DtLE_AppearanceForceID](#)
- [DtLE_AppearanceIR](#)
- [DtLE_AppearanceMarking](#)
- [DtLE_AppearanceType](#)
- [DtLE_AppearanceVisual](#)
- [DtLE_DetonBurstQuantityRate](#)
- [DtLE_DetonBurstWarheadFuse](#)
- [DtLE_DetonEventNumber](#)
- [DtLE_DetonEventSiteApp](#)
- [DtLE_DetonField2](#)
- [DtLE_DetonLocationEntityCoords](#)
- [DtLE_DetonMunitionID](#)
- [DtLE_DetonMunitionOrientation](#)
- [DtLE_DetonMunitionSiteApp](#)
- [DtLE_DetonTargetID](#)
- [DtLE_FireBurstQuantityRate](#)
- [DtLE_FireBurstWarheadFuse](#)
- [DtLE_FireEventSiteApp](#)
- [DtLE_FireLocation](#)
- [DtLE_FireMunitionID](#)
- [DtLE_FireMunitionSiteApp](#)
- [DtLE_FireTargetID](#)
- [DtLE_TSPI_DeadReckoning](#)
- [DtLE_TSPI_EntityLinearVelocity](#)
- [DtLE_TSPI_EntityOrientation](#)
- [DtLE_TSPI_MeasuredSpeed](#)
- [DtLE_TSPI_OrientationError](#)
- [DtLE_TSPI_PositionError](#)
- [DtLE_TSPI_Reserved](#)
- [DtLE_TSPI_SystemSpecific](#)
- [DtLGR_ALL](#)
- [DtLGR_BACKWARDS](#)
- [DtLGR_EMPTY](#)
- [DtLGR_END](#)
- [DtLGR_FORWARDS](#)

- [DtLGR_LOOP](#)
- [DtLGR_MIDDLE](#)
- [DtLGR_MINE](#)
- [DtLGR_NOFILE](#)
- [DtLGR_NONE](#)
- [DtLGR_NOPKT](#)
- [DtLGR_PAUSE](#)
- [DtLGR_PLAY](#)
- [DtLGR_PRISTINE](#)
- [DtLGR_RECORD](#)
- [DtLGR_RVA](#)
- [DtLGR_START](#)
- [DtLGR_STOP](#)
- [DtLGR_STOPPED](#)
- [DtLandingCraft](#)
- [DtLandingGear](#)
- [DtLarge](#)
- [DtLargeTrackedUtilityVehicle](#)
- [DtLargeWheeledUtilityVehicle](#)
- [DtLaserGeneric](#)
- [DtLaserSensorType](#)
- [DtLeAppearancePduKind](#)
- [DtLeArticulatedPartsPduKind](#)
- [DtLeDetonationPduKind](#)
- [DtLeFirePduKind](#)
- [DtLeTspiPduKind](#)
- [DtLeftAileron](#)
- [DtLeftFlap](#)
- [DtLeftWeaponBayDoor](#)
- [DtLevi_Civita](#)
- [DtLgrControllInteraction](#)
- [DtLgrControlPduKind](#)
- [DtLgrEnd](#)
- [DtLgrEofAction](#)
- [DtLgrHeartbeat](#)
- [DtLgrPause](#)
- [DtLgrPbFile](#)
- [DtLgrPlayAction](#)

- [DtLgrPlay](#)
- [DtLgrQuit](#)
- [DtLgrRecFile](#)
- [DtLgrRecordAction](#)
- [DtLgrRecord](#)
- [DtLgrSetTime](#)
- [DtLgrSkipTime](#)
- [DtLgrStart](#)
- [DtLgrStop](#)
- [DtLgrTimeScale](#)
- [DtLgrTimeout](#)
- [DtLifeFormDomainAir](#)
- [DtLifeFormDomainLand](#)
- [DtLifeFormDomainOther](#)
- [DtLifeFormDomainSpace](#)
- [DtLifeFormDomainSubsurface](#)
- [DtLifeFormDomainSurface](#)
- [DtLifeForm](#)
- [DtLifeSupportAirSupply](#)
- [DtLifeSupportCBRProtection](#)
- [DtLifeSupportDecontaminationSys](#)
- [DtLifeSupportFilters](#)
- [DtLifeSupportRefrigerationSys](#)
- [DtLifeSupportWaterSupply](#)
- [DtLifeSupportWaterWashSys](#)
- [DtLifeformCrawling](#)
- [DtLifeformCrouching](#)
- [DtLifeformJumping](#)
- [DtLifeformKneeling](#)
- [DtLifeformNA](#)
- [DtLifeformParachuting](#)
- [DtLifeformProne](#)
- [DtLifeformRunning](#)
- [DtLifeformSitting](#)
- [DtLifeformSquatting](#)
- [DtLifeformStanding](#)
- [DtLifeformSwimming](#)
- [DtLifeformWading](#)

- [DtLifeformWalking](#)
- [DtLightAlgae](#)
- [DtLightsFormation](#)
- [DtLightsNavigation](#)
- [DtLightsNone](#)
- [DtLightsRunning](#)
- [DtLineFormation](#)
- [DtLinearObjectStatePduKind](#)
- [DtLoader](#)
- [DtLowerWord](#)
- [DtLsb](#)
- [DtM109](#)
- [DtM113](#)
- [DtM1](#)
- [DtM2](#)
- [DtM3](#)
- [DtM88](#)
- [DtMAX_PARAMS_PER_PART](#)
- [DtMH53](#)
- [DtMH60](#)
- [DtMI24](#)
- [DtMI28](#)
- [DtMOPMigratorPduKind](#)
- [DtMTLB](#)
- [DtMagneticFuse](#)
- [DtMagneticGeneric](#)
- [DtMagneticSensorType](#)
- [DtMapDatum](#)
- [DtMaverick](#)
- [DtMaximumSocketNumber](#) *the following is used as the first argument to the PC version of select*
- [DtMedium](#)
- [DtMessagePdu](#)
- [DtMessagePduKind](#)
- [DtMiG23](#)
- [DtMiG29](#)
- [DtMineCounterMeasureShip](#)
- [DtMineEmplacementAge](#)
- [DtMineFusing](#)

- [DtMineGroundBurialDepthOffset](#)
- [DtMineOrientation](#)
- [DtMinePaintScheme](#)
- [DtMineReflectance](#)
- [DtMineScalarDetectionCoefficient](#)
- [DtMineSnowBurialDepthOffset](#)
- [DtMineThermalContrast](#)
- [DtMineTripDetonationWireLocation](#)
- [DtMineWaterBurialDepthOffset](#)
- [DtMinefieldDataPduKind](#)
- [DtMinefieldQueryPduKind](#)
- [DtMinefieldResponseNackPduKind](#)
- [DtMinefieldStatePduKind](#)
- [DtMk48AntiSubmarine](#)
- [DtMobileCommandAndControlEntityCarriedAboardHost](#)
- [DtModLocation](#)
- [DtModOrientation](#)
- [DtModSAF_Artillery_Fire_Control](#)
- [DtModSAF_FWA_Forward_Air_Controller](#)
- [DtModSAF_Text_Radio](#)
- [DtModSAF_Threat_ADA_C3](#)
- [DtModerateAlgae](#)
- [DtMortar](#)
- [DtMultispectralGeneric](#)
- [DtMultispectralSensorType](#)
- [DtMunitionCarriedAsCargo](#)
- [DtMunitionDomainOther](#)
- [DtMunition](#)
- [DtMunition](#)
- [DtMunition](#)
- [DtMunitionGuided](#)
- [DtMunitionUnguided](#)
- [DtNASANAD27](#)
- [DtNASBNAD27](#)
- [DtNPCollisionPduKind](#)
- [DtNPFieldPduKind](#)
- [DtNPSTSurfaceContactPduKind](#)
- [DtNassau](#)

- [DtNaturalWoodPaintScheme](#)
- [DtNet16Bits](#)
- [DtNet32Bits](#)
- [DtNet32Vector](#)
- [DtNet64Bits](#)
- [DtNet64Vector](#)
- [DtNet8Bits](#)
- [DtNetAngularVelocity](#)
- [DtNetArtParamRecord](#)
- [DtNetArtPartRecord](#)
- [DtNetAttachedPart](#)
- [DtNetBurstDescriptor](#)
- [DtNetClockTime](#)
- [DtNetEntityCoordinates](#)
- [DtNetEntityIdentifier](#)
- [DtNetEntityType](#)
- [DtNetEulerAngles](#)
- [DtNetEventId](#)
- [DtNetFloat32](#)
- [DtNetFloat64](#)
- [DtNetGlobalObjectDesignator](#)
- [DtNetInt16](#)
- [DtNetInt32](#)
- [DtNetInt8](#)
- [DtNetLinearVelocity](#)
- [DtNetObjectId](#)
- [DtNetRadioEntityType](#)
- [DtNetSimulationAddress](#)
- [DtNetTopoUpCoord](#)
- [DtNetU16](#)
- [DtNetU32](#)
- [DtNetU8](#)
- [DtNetWorldCoordinates](#)
- [DtNoAlgae](#)
- [DtNoAntiHandlingDevice](#)
- [DtNoAttachedParts](#) *This constant can be passed to a VR-Link function that expects a list of attached part stations to indicate an empty list*
- [DtNoFuse](#)
- [DtNoRepairsPerformed](#)

- [DtNoSuchAttachStation](#)
- [DtNoWay](#)
- [DtNoWeapon](#)
- [DtNorth](#)
- [DtNotTrackingAllTargets](#)
- [DtNotifyLevel](#)
- [DtOH58](#)
- [DtObjectCulturalFeature](#)
- [DtObjectId](#)
- [DtObjectId](#)
- [DtObjectIdHashList](#)
- [DtObstacleBreach](#)
- [DtObstacle](#)
- [DtObstacleMarker](#)
- [DtOff](#)
- [DtOlivePaintScheme](#)
- [DtOmniDirectional](#)
- [DtOn](#)
- [DtOnStation_RNG_BRG](#)
- [DtOnStation_XYZ](#)
- [DtOnTopOf](#)
- [DtOpticalSensorType](#)
- [DtOtherAggregateFormation](#)
- [DtOtherAggregateState](#)
- [DtOther](#)
- [DtOtherFuse](#)
- [DtOtherNature](#)
- [DtOtherObjectKind](#)
- [DtOtherPaintScheme](#)
- [DtOtherPduKind](#)
- [DtOtherPosition](#)
- [DtOtherSensorTypeCategory](#)
- [DtOtherStationName](#)
- [DtOtherTransfer](#)
- [DtPartAttached](#)
- [DtPartNotAttached](#)
- [DtPartiallyDisaggregated](#)
- [DtPassageway](#)

- [DtPatriot](#)
- [DtPatrolCraft](#)
- [DtPeleliu](#)
- [DtPeriscope](#)
- [DtPhoenix](#)
- [DtPhysicalSensorType](#)
- [DtPilot](#)
- [DtPlatformDomainAir](#)
- [DtPlatformDomainLand](#)
- [DtPlatformDomainOther](#)
- [DtPlatformDomainSpace](#)
- [DtPlatformDomainSubsurface](#)
- [DtPlatformDomainSurface](#)
- [DtPlatform](#)
- [DtPlatoon](#)
- [DtPm](#)
- [DtPointObjectStatePduKind](#)
- [DtPreRecordedVoice](#)
- [DtPreparedPosition](#)
- [DtPressureFuse](#)
- [DtPrimaryDefenseSys10](#)
- [DtPrimaryDefenseSys1](#)
- [DtPrimaryDefenseSys2](#)
- [DtPrimaryDefenseSys3](#)
- [DtPrimaryDefenseSys4](#)
- [DtPrimaryDefenseSys5](#)
- [DtPrimaryDefenseSys6](#)
- [DtPrimaryDefenseSys7](#)
- [DtPrimaryDefenseSys8](#)
- [DtPrimaryDefenseSys9](#)
- [DtPrimaryGun10](#)
- [DtPrimaryGun1](#)
- [DtPrimaryGun2](#)
- [DtPrimaryGun3](#)
- [DtPrimaryGun4](#)
- [DtPrimaryGun5](#)
- [DtPrimaryGun6](#)
- [DtPrimaryGun7](#)

- [DtPrimaryGun8](#)
- [DtPrimaryGun9](#)
- [DtPrimaryLauncher10](#)
- [DtPrimaryLauncher1](#)
- [DtPrimaryLauncher2](#)
- [DtPrimaryLauncher3](#)
- [DtPrimaryLauncher4](#)
- [DtPrimaryLauncher5](#)
- [DtPrimaryLauncher6](#)
- [DtPrimaryLauncher7](#)
- [DtPrimaryLauncher8](#)
- [DtPrimaryLauncher9](#)
- [DtPrimaryRadar10](#)
- [DtPrimaryRadar1](#)
- [DtPrimaryRadar2](#)
- [DtPrimaryRadar3](#)
- [DtPrimaryRadar4](#)
- [DtPrimaryRadar5](#)
- [DtPrimaryRadar6](#)
- [DtPrimaryRadar7](#)
- [DtPrimaryRadar8](#)
- [DtPrimaryRadar9](#)
- [DtPrimaryTurret10](#)
- [DtPrimaryTurret1](#)
- [DtPrimaryTurret2](#)
- [DtPrimaryTurret3](#)
- [DtPrimaryTurret4](#)
- [DtPrimaryTurret5](#)
- [DtPrimaryTurret6](#)
- [DtPrimaryTurret7](#)
- [DtPrimaryTurret8](#)
- [DtPrimaryTurret9](#)
- [DtProbeMetalContent](#)
- [DtProbeNoMetalContent](#)
- [DtProtocolVersionToRecvMax](#)
- [DtProtocolVersionToRecvMin](#)
- [DtProtocolVersionToSend](#)
- [DtPseudoDisaggregated](#)

- [DtPseudoNoise](#)
- [DtPseudoNoiseFrequencyHopping](#)
- [DtPseudoNoiseFrequencyHoppingTimeHopping](#)
- [DtPseudoNoiseTimeHopping](#)
- [DtPulse](#)
- [DtQ1](#)
- [DtQ2](#)
- [DtQ3](#)
- [DtQRP_Mode](#)
- [DtQuantity](#)
- [DtQuantity](#)
- [DtQuantity](#)
- [DtQuaternion](#)
- [DtRADAR_Generic](#)
- [DtRADAR_SensorType](#)
- [DtRAH66](#)
- [DtRadarBeam](#)
- [DtRadioCryptoSysKY_28](#)
- [DtRadioCryptoSysKY_8](#)
- [DtRadioCryptoSysNarrowSpectrumSecureVoice_NSVE](#)
- [DtRadioCryptoSysOther](#)
- [DtRadioCryptoSysSINCGARS_ICOM](#)
- [DtRadioCryptoSysWideSpectrumSecureVoice_WSVE](#)
- [DtRadioDetailModCombo](#)
- [DtRadioDetailModOther](#)
- [DtRadioDetailModUnmod](#)
- [DtRadioDetailedModulation](#)
- [DtRadioInputOther](#)
- [DtRadioMajorModOther](#)
- [DtRadioModAmpAngOther](#)
- [DtRadioModeAngleOther](#)
- [DtRadioOff](#)
- [DtRadioOn](#)
- [DtRadioOnNotReceiving](#)
- [DtRadioOnNotTrasmitting](#)
- [DtRadioSignalEncoding](#)
- [DtRadioSpreadOther](#)
- [DtRadioSystemOther](#)

- [DtRate](#)
- [DtRate](#)
- [DtRate](#)
- [DtRawBinary](#)
- [DtReal](#)
- [DtReceive](#)
- [DtReceiverPduKind](#)
- [DtReconnaissance](#)
- [DtRecordPduKind](#)
- [DtRecordQueryPduKind](#)
- [DtRecoveryCap](#)
- [DtRedPaintScheme](#)
- [DtReferenceSysEntityCoordinates](#)
- [DtReferenceSysWorldCoordinates](#)
- [DtRegiment](#)
- [DtRegularBeam](#)
- [DtReliable](#)
- [DtRemoveEntityInteraction](#)
- [DtRemoveEntityPduKind](#)
- [DtRemoveEntityRPduKind](#)
- [DtRemoveObjectResultOther](#)
- [DtRepairCap](#)
- [DtRepairCompletePduKind](#)
- [DtRepairEverything](#)
- [DtRepairResponseInteraction](#)
- [DtRepairResponsePduKind](#)
- [DtRepairResultCancelled](#)
- [DtRepairResultEnded](#)
- [DtRepairResultInterrupted](#)
- [DtRepairResultInvalid](#)
- [DtRepairResultOther](#)
- [DtReqExchangeEntity](#)
- [DtReqExchangeEnvProc](#)
- [DtRequestStatusComplete](#)
- [DtRequestStatusExecuting](#)
- [DtRequestStatusInvalidTimeParameters](#)
- [DtRequestStatusOther](#)
- [DtRequestStatusPartial](#)

- [DtRequestStatusPending](#)
- [DtRequestStatusRejected](#)
- [DtRequestStatusRequestDone](#)
- [DtRequestStatusRetransmitLater](#)
- [DtRequestStatusRetransmitNow](#)
- [DtRequestStatusSimulationTimeExceeded](#)
- [DtResupplyCancelInteraction](#)
- [DtResupplyCancelPduKind](#)
- [DtResupplyOfferInteraction](#)
- [DtResupplyOfferPduKind](#)
- [DtResupplyRecInteraction](#)
- [DtResupplyReceivedPduKind](#)
- [DtRightAileron](#)
- [DtRightFlap](#)
- [DtRightWeaponBayDoor](#)
- [DtRoland](#)
- [DtRudder](#)
- [DtSA10FlapLidRadar](#)
- [DtSA10Grumble](#)
- [DtSA2Guideline](#)
- [DtSA3Goa](#)
- [DtSA4Ganef](#)
- [DtSA6Gainful](#)
- [DtSA6Launcher](#)
- [DtSA7Grail](#)
- [DtSA8Gecko](#)
- [DtSA8LauncherRadar](#)
- [DtSA9Gaskin](#)
- [DtSAMSite](#)
- [DtSBBNOhio](#)
- [DtSONAR_Generic](#)
- [DtSONAR_SensorType](#)
- [DtSS1ScudMissileLauncher](#)
- [DtSU25](#)
- [DtSaipan](#)
- [DtSeaGoingHelicopter](#)
- [DtSeaSparrow](#)
- [DtSec2Hour](#)

- [DtSecondaryDefenseSys10](#)
- [DtSecondaryDefenseSys1](#)
- [DtSecondaryDefenseSys2](#)
- [DtSecondaryDefenseSys3](#)
- [DtSecondaryDefenseSys4](#)
- [DtSecondaryDefenseSys5](#)
- [DtSecondaryDefenseSys6](#)
- [DtSecondaryDefenseSys7](#)
- [DtSecondaryDefenseSys8](#)
- [DtSecondaryDefenseSys9](#)
- [DtSecondaryGun10](#)
- [DtSecondaryGun1](#)
- [DtSecondaryGun2](#)
- [DtSecondaryGun3](#)
- [DtSecondaryGun4](#)
- [DtSecondaryGun5](#)
- [DtSecondaryGun6](#)
- [DtSecondaryGun7](#)
- [DtSecondaryGun8](#)
- [DtSecondaryGun9](#)
- [DtSecondaryLauncher10](#)
- [DtSecondaryLauncher1](#)
- [DtSecondaryLauncher2](#)
- [DtSecondaryLauncher3](#)
- [DtSecondaryLauncher4](#)
- [DtSecondaryLauncher5](#)
- [DtSecondaryLauncher6](#)
- [DtSecondaryLauncher7](#)
- [DtSecondaryLauncher8](#)
- [DtSecondaryLauncher9](#)
- [DtSecondaryRadar10](#)
- [DtSecondaryRadar1](#)
- [DtSecondaryRadar2](#)
- [DtSecondaryRadar3](#)
- [DtSecondaryRadar4](#)
- [DtSecondaryRadar5](#)
- [DtSecondaryRadar6](#)
- [DtSecondaryRadar7](#)

- [DtSecondaryRadar8](#)
- [DtSecondaryRadar9](#)
- [DtSecondaryTurret10](#)
- [DtSecondaryTurret1](#)
- [DtSecondaryTurret2](#)
- [DtSecondaryTurret3](#)
- [DtSecondaryTurret4](#)
- [DtSecondaryTurret5](#)
- [DtSecondaryTurret6](#)
- [DtSecondaryTurret7](#)
- [DtSecondaryTurret8](#)
- [DtSecondaryTurret9](#)
- [DtSelfPropelledArtillery](#)
- [DtServiceRequestInteraction](#)
- [DtServiceRequestPduKind](#)
- [DtServiceTypeOther](#)
- [DtServiceTypeRepair](#)
- [DtServiceTypeResupply](#)
- [DtSetDataInteraction](#)
- [DtSetDataPduKind](#)
- [DtSetDataRPduKind](#)
- [DtSetRecordPduKind](#)
- [DtShipsAftGunmountCenterline](#)
- [DtShipsAftGunmountPort](#)
- [DtShipsAftGunmountStarboard](#)
- [DtShipsForwardGunmountCenterline](#)
- [DtShipsForwardGunmountPort](#)
- [DtShipsForwardGunmountStarboard](#)
- [DtSidewinder](#)
- [DtSignalClassApplicationSpecific](#)
- [DtSignalClassEncodedVoice](#)
- [DtSignalClassPreRecorded](#)
- [DtSignalClassRawBinary](#)
- [DtSignalInteraction](#)
- [DtSignalPduKind](#)
- [DtSimulationAddress](#)
- [DtSimulator](#)
- [DtSimulator](#) *Public for backwards compatibility*
- **DtSimulator :**

- in [DtNetEntityIdentifier](#)
 - in [DtNetEventId](#)
- [DtSincgards](#)
- [DtSite](#)
- [DtSmall](#)
- [DtSmallTrackedUtilityVehicle](#)
- [DtSmallWheeledUtilityVehicle](#)
- [DtSmoke](#)
- [DtSnorkel](#)
- [DtSoldier1HourSleep](#)
- [DtSoldier2HourSleep](#)
- [DtSoldier3HourSleep](#)
- [DtSoldier4HourSleep](#)
- [DtSoldier5HourSleep](#)
- [DtSoldier6HourSleep](#)
- [DtSoldier7HourSleep](#)
- [DtSoldierFullyRested](#)
- [DtSoldierNotRested](#)
- [DtSouth](#)
- [DtSparrow](#)
- [DtSpecialOperations](#)
- [DtSpecific](#)
- [DtSpecificAH1W](#)
- [DtSpecificCV59](#)
- [DtSpecific](#)
- [DtSpecific](#)
- [DtSpecificFFG7](#)
- [DtSpecificSA10](#)
- [DtSpecificSA8](#)
- [DtSpeedBrake](#)
- [DtSphericalHarmonic](#)
- [DtSpheroid](#)
- [DtSquad](#)
- [DtSsbFull](#)
- [DtSsbReduc](#)
- [DtStandardPaintScheme](#)
- [DtStartResumeInteraction](#)
- [DtStartResumePduKind](#)

- [DtStartResumeRPduKind](#)
- [DtStateUpdate](#)
- [DtStinger](#)
- [DtStopFreezeInteraction](#)
- [DtStopFreezePduKind](#)
- [DtStopFreezeRPduKind](#)
- [DtStopReasonEntityReconstitution](#)
- [DtStopReasonOther](#)
- [DtStopReasonRecess](#)
- [DtStopReasonSecurityViolation](#)
- [DtStopReasonSystemFailure](#)
- [DtStopReasonTermination](#)
- [DtStrategic](#)
- [DtSubCategory](#)
- [DtSubCategory](#)
- [DtSubCategory](#)
- [DtSubProtocolInvalid](#)
- [DtSuppEmisEntStatePduKind](#)
- [DtSupply](#)
- [DtSupplyFood](#)
- [DtSupplyFuels](#)
- [DtSupplyLubricants](#)
- [DtSupplyOils](#)
- [DtSupplyOther](#)
- [DtSupplyPersonnel](#)
- [DtSupplySpareParts](#)
- [DtSureThing](#)
- [DtT64](#)
- [DtT72](#)
- [DtTU26](#)
- [DtTU95](#)
- [DtTacticalDataLinkOther](#)
- [DtTacticalSmoke](#)
- [DtTailHook](#)
- [DtTanPaintScheme](#)
- [DtTankAPCHatch](#)
- [DtTank](#)
- [DtTankLandingShip](#)

- [DtTarawaClass](#)
- [DtTarawa](#)
- [DtTdlGC3](#)
- [DtTeamMemberInFormationWith](#)
- [DtText](#)
- [DtThomasSGates](#)
- [DtThreeIndex](#) *Unobvious Constants*
- [DtTiconderogaClass](#)
- [DtTiconderoga](#)
- [DtTiltRodFuse](#)
- [DtTime](#)
- [DtTimeHopping](#)
- [DtTimeStamp](#) *Host representation of a timeStamp*
- [DtTimeStampAbsolute](#)
- [DtTimeStampRelative](#)
- [DtTimestamp](#)
- [DtTowedArtillery](#)
- [DtTrackBeam](#)
- [DtTrackedVehicleOccClosedHatchActSearching](#)
- [DtTrackedVehicleOccClosedHatchNotActSearching](#)
- [DtTrackedVehicleOccOpenHatchActSearching](#)
- [DtTrackedVehicleOccOpenHatchNotActSearching](#)
- [DtTrackingAllTargets](#)
- [DtTrailerBed](#)
- [DtTrailingEffectsLarge](#)
- [DtTrailingEffectsMedium](#)
- [DtTrailingEffectsNone](#)
- [DtTrailingEffectsSmall](#)
- [DtTransferControlRequestPduKind](#)
- [DtTransmitterPduKind](#)
- [DtTripWireFuse](#)
- [DtTruckBed](#)
- [DtTruckOccActivelySearching](#)
- [DtTruckOccNotActivelySearching](#)
- [DtTrue](#)
- [DtType](#)
- [DtU16](#)
- [DtU32](#)

- [DtU8](#)
- [DtUAV](#)
- [DtUSCivilianCar](#)
- [DtUnacknowledged](#)
- [DtUnaidedEyeActivelySearching](#)
- [DtUnaidedEyeNotActivelySearching](#)
- [DtUndefinedGroupEntityCategory](#)
- [DtUnderwaterAcousticALFS](#)
- [DtUnderwaterAcousticAN_AQS_901](#)
- [DtUnderwaterAcousticAN_AQS_902](#)
- [DtUnderwaterAcousticAN_BQQ_5](#)
- [DtUnderwaterAcousticAN_SQS_23](#)
- [DtUnderwaterAcousticAN_SQS_26](#)
- [DtUnderwaterAcousticAN_SQS_53](#)
- [DtUnderwaterAcousticAN_SSQ_62](#)
- [DtUnderwaterAcousticActiveOtherIndex](#)
- [DtUnderwaterAcousticBattery](#)
- [DtUnderwaterAcousticChangedDataUpdate](#)
- [DtUnderwaterAcousticConical](#)
- [DtUnderwaterAcousticContinuousSearch](#)
- [DtUnderwaterAcousticDiesel](#)
- [DtUnderwaterAcousticDiesel_electric](#)
- [DtUnderwaterAcousticGasTurbine](#)
- [DtUnderwaterAcousticGrahamsMKV_CoffeeMaker](#)
- [DtUnderwaterAcousticHelical](#)
- [DtUnderwaterAcousticHullMountedMaskerOff](#)
- [DtUnderwaterAcousticHullMountedMaskerOn](#)
- [DtUnderwaterAcousticLFA](#)
- [DtUnderwaterAcousticMineHunting](#)
- [DtUnderwaterAcousticNavigation](#)
- [DtUnderwaterAcousticNuclear](#)
- [DtUnderwaterAcousticOtherAdditionalIndex](#)
- [DtUnderwaterAcousticOtherConfig](#)
- [DtUnderwaterAcousticOtherFunction](#)
- [DtUnderwaterAcousticOtherPassiveIndex](#)
- [DtUnderwaterAcousticOtherSystem](#)
- [DtUnderwaterAcousticPduKind](#)
- [DtUnderwaterAcousticPlatformSearch_detect_track](#)

- [DtUnderwaterAcousticRaster](#)
- [DtUnderwaterAcousticScanPatternNotUsed](#)
- [DtUnderwaterAcousticSectorSearch](#)
- [DtUnderwaterAcousticStateUpdate](#)
- [DtUnderwaterAcousticSteam](#)
- [DtUnderwaterAcousticTurbineReduction](#)
- [DtUnderwaterAcousticWeaponSearch_detect_track_detect](#)
- [DtUnionOfSovietSocialistRepublics](#)
- [DtUnitedKingdom](#)
- [DtUnitedStates](#)
- [DtUnmannedAirVehicle](#)
- [DtUnmodulated](#)
- [DtUnusedCharacterSet](#)
- [DtUpperWord](#)
- [DtUral375](#)
- [DtUsb](#)
- [DtUseSimTimeForApproxRealTime](#)
- [DtUserGC3](#)
- [DtUserProtocolOther](#)
- [DtUtilityHelicopter](#)
- [DtUtmLatRef1_1](#) *Origin of the IITSEC '92 terrain database*
- [DtUtmLatRefZone10](#) *The standard UTM reference for a zone is the point along the zone's central meridian at the equator*
- [DtUtmLatRef_IITSEC_93](#) *Origin of the IITSEC '94 & '95 terrain database*
- [DtUtmLonRef1_1](#)
- [DtUtmLonRefZone10](#)
- [DtUtmLonRef_IITSEC_93](#)
- [DtVQ_Vector_Quantization](#)
- [DtValleyForge](#)
- [DtValue](#)
- [DtVar](#)
- [DtVectorRef](#)
- [DtVeeFormation](#)
- [DtVehicleCarriedAsCargo](#)
- [DtVeryLarge](#)
- [DtVerySmall](#)
- [DtViewControlInteraction](#)
- [DtViewControlPduKind](#)
- [DtViewerAbsolute](#)

- [DtViewerCompass](#)
- [DtViewerGroup](#)
- [DtViewerMimic](#)
- [DtViewerMimicTrack](#)
- [DtViewerModeless](#)
- [DtViewerOrbit](#)
- [DtViewerTether](#)
- [DtViewerTetherTrack](#)
- [DtViewerTrack](#)
- [DtViewerTurret](#)
- [DtVincennes](#)
- [DtVsb](#)
- [DtWGS84](#)
- [DtWNCP_data](#)
- [DtWarhead](#)
- [DtWarheadBiologicalGeneral](#)
- [DtWarheadChemicalBlisterAgent](#)
- [DtWarheadChemicalBloodAgent](#)
- [DtWarheadChemicalGeneral](#)
- [DtWarheadChemicalNerveAgent](#)
- [DtWarhead](#)
- [DtWarhead](#)
- [DtWarheadHighExplosiveAntiTank](#)
- [DtWarheadHighExplosiveBomblets](#)
- [DtWarheadHighExplosive](#)
- [DtWarheadHighExplosiveFragmentation](#)
- [DtWarheadHighExplosiveIncendiary](#)
- [DtWarheadHighExplosivePlastic](#)
- [DtWarheadHighExplosiveShapedCharge](#)
- [DtWarheadIllumination](#)
- [DtWarheadKinetic](#)
- [DtWarheadNuclear](#)
- [DtWarheadOther](#)
- [DtWarheadPractice](#)
- [DtWarheadSmoke](#)
- [DtWarheadUnused](#)
- [DtWeaponCommandersGPSExtension](#)
- [DtWeaponDeployed](#)

- [DtWeaponGunElevationDrive](#)
- [DtWeaponGunStabilizationSys](#)
- [DtWeaponGunneryPrimarySight](#)
- [DtWeaponInFirePosition](#)
- [DtWeaponLoadingMechanism](#)
- [DtWeaponStowed](#)
- [DtWedgeFormation](#)
- [DtWellDeck](#)
- [DtWest](#)
- [DtWhitePaintScheme](#)
- [DtWingSweep](#)
- [DtWinterCamouflage](#)
- [DtX](#)
- [DtY](#)
- [DtY](#)
- [DtYellowPaintScheme](#)
- [DtYorktown](#)
- [DtZ](#)
- [DtZ](#)
- [DtZIL157](#)
- [DtZSU23_4](#)
- [DtkObjectIsDis](#)
- [DtkObjectIsHla](#)
- [DtkObjectIsUnknown](#)
- [Dto_EQ](#)
- [Dto_NEQ](#)
- [GtQueueElement](#)
- [Major](#)
- [North](#)
- [PacketQueueIn](#)
- [PacketQueueOut](#)
- [RW_flags](#)
- [Spectrum](#)
- [StealthInstruction](#)
- [Sys](#)
- [TRANSPose_M1_AND_M2](#)
- [TRANSPose_M1](#)
- [TRANSPose_M2](#)

- [TRANSPPOSE_NONE](#)
- [Up](#)
- [a1](#)
- [a2](#)
- [acc](#)
- [ackFlag](#)
- [addr](#)
- [addr](#)
- [ap](#)
- [appid](#)
- [asSeenByRemote](#)
- [att](#)
- [att](#)
- [attachedType](#)
- [availEventNum](#)
- [beamAzCenter](#)
- [beamAzCenter](#)
- [beamAzSweep](#)
- [beamAzSweep](#)
- [beamElCenter](#)
- [beamElCenter](#)
- [beamElSweep](#)
- [beamElSweep](#)
- [beamId](#)
- [beamId](#)
- [beamSweepSync](#)
- [beamSweepSync](#)
- [bits](#)
- [blocking](#)
- [boundToDestAddr](#)
- [buffer](#)
- [bundling](#)
- [cacheData](#)
- [cacheData](#)
- [cache](#)
- [cacheEvent](#)
- [cacheEvent](#)
- [cacheld](#)

- [cachedId](#)
- [cacheLoc](#)
- [cacheLoc](#)
- [cacheTarget](#)
- [cacheTarget](#)
- [caddr_t](#)
- **category :**
 - in [duplicate](#)
 - in [DtNetRadioEntityType](#)
- [clientId](#)
- [clientSocketPath](#)
- [col](#)
- [controlMsg](#)
- **country :**
 - in [duplicate](#)
 - in [DtNetRadioEntityType](#)
- [data](#)
- [datalen](#)
- [datumShift](#)
- [deleteSocket](#)
- [destAddr](#)
- [destInetAddr](#)
- [dfitNumBuckets](#) *Not static member since g++ doesn't like to use static members as default arguments*
- [direction](#)
- **domain :**
 - in [duplicate](#)
 - in [DtNetRadioEntityType](#)
- [dp](#)
- [dv](#)
- [effectiveRadiatedPower](#)
- [effectiveRadiatedPower](#)
- [emitterId](#)
- [emitterId](#)
- [ent](#)
- [ent](#)
- [entityId](#)
- [entityId](#)
- [event](#)
- [event](#)

- [event](#)
- [exercise](#)
- [exercise](#)
- [exid](#)
- [filterMsg](#)
- [filtering](#)
- [firstExId](#)
- [firstInBundle](#)
- [firstProtocol](#)
- [forceSendNeeded](#)
- [frequency](#)
- [frequency](#)
- [frequencyRange](#)
- [frequencyRange](#)
- [from](#)
- [from_addr](#)
- **frozen :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [getFcn](#)
- [hourFrac](#)
- [initErrorAction](#)
- [internal](#)
- [iss](#)
- [kind](#)
- [kind](#)
- [kind](#)
- **lastT :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [length](#)
- [list](#)
- [loc](#)
- [loc](#)
- [loc](#)
- [loc](#)
- [loopbackAddr](#)
- **lsbyte :**
 - in [DtNetInt16](#)

- in [DtNetU16](#) *Most Significant Byte*
- [lslong](#) *Most Significant Long*
- **lsword :**
 - in [DtNetInt32](#) *Most Significant Word*
 - in [DtNetU32](#) *Most Significant Word*
- [m_bExternalSocket](#)
- [m_devAddr](#)
- [markingBuff](#)
- [maxBundleSize](#)
- [maxfd](#)
- [mcastAddrs](#)
- [min](#)
- [modType](#)
- **msbyte :**
 - in [DtNetInt16](#)
 - in [DtNetU16](#)
- [mslong](#)
- **msword :**
 - in [DtNetInt32](#)
 - in [DtNetU32](#)
- [mun](#)
- [mun](#)
- [myAPA](#)
- [myAccel](#)
- [myAcceleration](#)
- [myAcousticID](#)
- [myAcousticName](#)
- [myAction](#) *parameter*
- [myAction](#) *parameter*
- [myAction](#) *parameter*
- [myActiveEmissionParmIndex](#)
- [myAggDimThresh](#)
- [myAggregateDecoder](#)
- [myAggregateDecoder](#)
- [myAggregateEncoder](#)
- [myAggregateState](#)
- [myAggregateStateRep](#)
- [myAggregateStateRep](#)
- [myAggregateStateRep](#)
- [myAirMaintenanceStatus](#)

- [myAirMaintenanceStatus](#)
- [myAlgorithm](#)
- [myAlgorithm](#)
- [myAltParameter4](#)
- [myAngles](#)
- [myAngles](#)
- [myAngularVelocity](#)
- [myAntPat](#)
- [myAntennaPatternType](#)
- [myAppearance](#)
- [myAppearance](#)
- [myAppearance](#)
- [myAppearance](#)
- [myAppearance](#)
- [myAppearance](#)
- [myAppearance](#)
- [myApplicableModes](#)
- [myApplication](#)
- [myApproxFlag](#)
- [myApproximator](#) *holds what's currently being used as our approximator*
- [myArtPartList](#) *Does not store state*
- [myArtPartList](#)
- [myArtPartThresh](#)
- [myAsPdu](#)
- [myAsSeenByRemote](#)
- [myAttPartList](#)
- [myAttachedTo](#)
- [myAxisType](#)
- [myAzimuthBandwidth](#)
- [myAzimuthCenter](#) *Common to emitter beams and track jam beams*
- [myAzimuthCenter](#) *Data members:*
- [myAzimuthError](#) *Data members:*
- [myAzimuthSweep](#)
- [myAzimuthSweep](#)
- [myAzimuthalBeamwidth](#)
- [myBeamCenterAzimuth](#)
- [myBeamCenterD_E](#)
- [myBeamData](#)

- [myBeamDataLength](#) *Data members:*
- [myBeamDirection](#)
- [myBeamFunction](#)
- [myBeamID_Number](#)
- [myBeamId](#)
- [myBeamId](#)
- [myBeamList](#) *list of beams in this system*
- [myBeamParameterIndex](#)
- [myBeamType](#)
- [myBeginItem](#)
- [myBeginSocket](#)
- [myBodyToGeocAtTimeOfValidity](#)
- [myBodyToGeocDirty](#)
- [myBodyToGeoc](#)
- [myBodyToLocal](#)
- [myBodyToLocal](#)
- [myBuckets](#) *Array of buckets (DtLists of DtHashItems)*
- [myBurstLength](#)
- [myCallbackFunction](#)
- [myCapabilities](#)
- [myCategory](#)
- [myCbList](#)
- [myChange](#)
- [myChangeOptions](#)
- [myCodeName](#)
- [myConn](#)
- [myCoordinateOffsetXi](#)
- [myCoordinateScaleXi](#)
- [myCount](#)
- [myCreators](#) *myCreators is an array of theMaxPduKinds items*
- [myCryptoKeyId](#)
- [myCryptoMode](#)
- [myCryptoSystem](#)
- [myCurrentEventId](#)
- [myCurrentId](#)
- [myCurrentShaftRPM](#) *Data members:*
- [myDRAlgorithm](#)
- [myDRBodyToGeoc](#)

- [myDRBodyToGeocTime](#)
- [myDREuler](#)
- [myDREulerTime](#)
- [myDRLoc](#)
- [myDRPosTime](#)
- [myDRVel](#)
- [myD_E_Beamwidth](#)
- [myData](#)
- [myData](#)
- [myData](#)
- [myData](#)
- [myData](#)
- [myDataLength](#) *Data members:*
- [myDatumID](#)
- [myDeadReckoner](#) *holds what was most recently passed to useDeadReckoner*
- [myDeadReckoningAlgorithm](#) *Data members:*
- [myDeltaX](#)
- [myDeltaY](#)
- [myDeltaZ](#)
- [myDesHashKey](#)
- [myDesignatedObject](#)
- [myDesignatorCode](#)
- [myDesignatorId](#)
- [myDesignatorPower](#)
- [myDesignatorSpotLocation](#)
- [myDesignatorWavelength](#)
- [myDestDeviceID](#)
- [myDestEntityID](#)
- [myDestEntityID](#)
- [myDestEquipID](#)
- [myDestGroupBitField](#)
- [myDestLineID](#)
- [myDestLineID](#)
- [myDestLineStateCmd](#)
- [myDestLineStateCmd](#)
- [myDestPriority](#)
- [myDestPriority](#)
- [myDetailedModulation](#)

- [myDfltDeadReckoner](#) *Created and deleted by this ESR if needed*
- [myDfltPduFact](#)
- [myDfltSmoother](#)
- [myDimensions](#)
- [myDirectedRotRate](#)
- [myDomain](#) *Data members:*
- [myDomainFinalXi](#)
- [myDomainInitialXi](#) *Data members:*
- [myDomainNumPointsXi](#)
- [myDumRec](#)
- [myDumRec](#)
- [myERP](#)
- [myERP](#) *Data members:*
- [myESR](#)
- [myESR](#)
- [myElevationBeamwidth](#)
- [myElevationCenter](#)
- [myElevationCenter](#)
- [myElevationError](#)
- [myElevationSweep](#)
- [myElevationSweep](#)
- [myEmitterFunction](#)
- [myEmitterId](#)
- [myEmitterName](#)
- [myEmitterNumber](#)
- [myEmittingSystemId](#)
- **myEnable :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [myEngineRPM](#)
- [myEntCoords](#)
- [myEntities](#)
- [myEntityAppearanceList](#)
- [myEntityDecoder](#)
- [myEntityDecoder](#)
- [myEntity](#)
- [myEntityEncoder](#)
- [myEntityID](#) *Temporary objects for reference returns*
- [myEntityID](#) *Temporary objects for reference returns*

- [illegible]

- [myEntityStateRep](#)
- [myEntityType](#)
- [myEntityType](#)
- [myEntityType](#)
- [myEntityType](#)
- [myEntityType](#)
- [myEntityType](#)
- [myEntityType](#)
- [myEntityType](#)
- [myEntityType](#)
- [myEntityType](#)
- [myEntityType](#)
- [myEnvironmentType](#)
- [myEsPdu](#)
- [myEulerAngles](#)
- [myEulerAngles](#)
- [myEulerAngles](#)
- [myEulerAngles](#)
- [myEventID](#)
- [myEventID](#)
- [myEventID](#)
- [myEventId](#)
- [myEventId](#)
- [myExConn](#)
- [myExConn](#)
- [myExConn](#)
- [myExConn](#)
- [myEx](#)
- [myExternalBuffer](#)
- [myEz](#)
- [myFd](#)
- [myFieldOffset](#)
- [myFieldScale](#)
- [myFilename](#) *parameter*
- [myFilename](#) *parameter*
- [myFirst](#)
- [myFirstParam](#)
- [myForcelD](#)

- [myForceId](#)
- [myForceSendNeeded](#)
- [myFormation](#)
- [myFracThresh](#)
- [myFreqHi](#)
- [myFreqLo](#)
- [myFrequencyBandwidth](#)
- [myFrequency](#)
- [myFrequency](#)
- [myFrequencyHopInUse](#)
- [myFrequencyRange](#)
- [myFrozen](#)
- [myFuelStatus](#)
- [myFuelStatus](#)
- [myFuelStatus](#)
- [myFunc](#)
- [myFunction](#)
- [myGeocLocForGeocToUtm](#)
- [myGeocLocForUtmToGeoc](#)
- [myGeocToLocal](#)
- [myGeocToUtmMatrix](#)
- [myGroundMaintenanceStatus](#)
- [myGroupBitField](#)
- [myGuise](#)
- [myGunElevation](#)
- [myGunElevation](#)
- [myGunElevationRate](#)
- [myGunElevationRate](#)
- [myHashKey](#)
- [myHashKey](#)
- [myHashKey](#)
- [myHashKey](#)
- [myHashKey](#)
- [myHashValue](#)
- [myHashValuesComputed](#)
- [myHashlist](#)
- [myHashlist](#)
- [myHeadAzimuth](#)

- [myHeadElevation](#)
- [myHeadElevationRate](#)
- [myHeadScanRate](#)
- [myHeartbeat](#) *parameter*
- [myHighDensity](#)
- [myHorizontalDeflection](#) *Data members:*
- [myHorizontalDeviation](#)
- [myHorizontalDeviation](#)
- [myHorizontalError](#) *Data members:*
- [myHostId](#)
- [myHtau](#)
- [myId](#)
- [myId](#)
- [myId](#)
- [myId](#)
- [myId](#)
- [myId](#)
- [myIdString](#)
- [myInitialIndexXi](#)
- [myInputSource](#)
- [myIntBuffSize](#)
- [myInterleafFactor](#)
- [myInternalBuff](#)
- [myItem](#)
- [myItem](#)
- [myJammingModeSequence](#) *For Track Jam Beams*
- [myKey](#)
- [myKeys](#) *DtList of keys (DtBaseHashKeys)*
- [myKind](#)
- [myLast](#)
- [myLastParam](#)
- [myLastSimTimeUpdated](#)
- [myLayerHeader](#)
- [myLayerInfo](#)
- [myLayerLength](#)
- [myLayerNumber](#) *Data members:*
- [myLayers](#)
- [myLinearAcceleration](#)

- [myList](#)
- [myList](#) Regular DtList containing all of the items added to the hashlist, without regard to key information
- [myList](#)
- [myList](#)
- [myListIsMine](#)
- [myLocAtTimeOfValidity](#)
- [myLocalAccel](#)
- [myLocalAccel](#)
- [myLocalHtau](#)
- [myLocalLoc](#) used as return values by inspectors
- [myLocalLoc](#) used as return values by inspectors
- [myLocalOrient](#)
- [myLocalOrient](#)
- **myLocalTau :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [myLocalTauInv](#)
- [myLocalToGeoc](#)
- [myLocalVel](#)
- [myLocalVel](#)
- [myLocation](#)
- [myLocation](#)
- [myLocation](#)
- [myLocation](#)
- [myMajorModulationType](#)
- [myMarkingCharSet](#)
- [myMarkingCharSet](#)
- [myMarkingText](#)
- [myMarkingText](#) myMarkingText is an array of theMaxMarkingSize
- [myMode](#)
- [myModifications](#)
- [myModifier](#)
- [myModulationParamsAddr](#)
- [myModulationParamsLength](#)
- [myModulationSystem](#)
- [myMovementSpeed](#)
- [myMovementSpeed](#)
- [myMsgSize](#)
- [myMunType](#)

- [myMunType](#)
- [myName](#)
- [myNetMessage](#)
- [myNetRep](#)
- [myNetSize](#)
- [myNext](#)
- [myNext](#)
- [myNumArtParams](#)
- [myNumArtPartParams](#) *Number of articulated part parameters for this PDU*
- [myNumArtPartParams](#) *Number of articulated part parameters for this PDU*
- [myNumArtParts](#)
- [myNumAttPartParams](#) *Number of attached part parameters for this PDU*
- [myNumAttPartParams](#) *Number of attached part parameters for this PDU*
- [myNumBuckets](#)
- [myNumDataBytes](#)
- [myNumDataValues](#)
- [myNumDataValues](#)
- [myNum](#)
- [myNum](#)
- [myNum](#)
- [myNumParameterSets](#)
- [myNumParams](#)
- [myNumParmRecords](#) *Number of parameter records in this PDU*
- [myNumParts](#)
- [myNumPointsXi](#)
- [myNumberOfAggregates](#) *Data members:*
- [myNumberOfAppearanceRecords](#)
- [myNumberOfBeams](#)
- [myNumberOfEntities](#) *Data members:*
- [myNumberOfSilentAggregateTypes](#)
- [myNumberOfSilentEntityTypeTypes](#)
- [myNumberOfVariableDatumRecords](#)
- [myOKStatus](#)
- [myObjectAddedCallbacks](#)
- [myObjectHashlist](#)
- [myObjectId](#)
- [myObjectId](#)
- [myObjectList](#)

- [myObjectRemovedCallbacks](#)
- [myObjectType](#)
- [myObjectType](#)
- [myObjectType](#)
- [myObjs](#)
- [myObjs](#)
- [myOk](#)
- [myOk](#)
- [myOperational](#)
- [myOrderedShaftRPM](#)
- [myOrientation](#)
- [myOrientation](#)
- [myPRF](#)
- [myParamNum](#)
- [myParameter1](#) *Data members:*
- [myParameter2](#)
- [myParameter](#)
- [myParameter](#)
- [myParmIndex](#) *Data members:*
- [myPartAttachedTo](#)
- [myPartList](#)
- [myPartNum](#)
- [myPartType](#)
- [myPduCbMgr](#)
- [myPdu](#)
- [myPduFact](#)
- [myPerimeterRecord](#)
- [myPerimeterRecord](#)
- [myPgRF](#)
- [myPhase](#)
- [myPhi](#)
- [myPhi](#)
- [myPhi](#)
- [myPhi](#)
- [myPhi](#)
- [myPhi](#)
- [myPhi](#) *about intermediate y*
- [myPktBuff](#)

- [myPostDrainCbs](#)
- [myPower](#)
- [myPowerSetting](#) *Data members:*
- [myPrev](#)
- [myPrimaryAmmunition](#)
- [myPrimaryAmmunition](#)
- [myPrimaryAmmunition](#)
- [myPrimaryAmmunition](#)
- [myPsi](#)
- [myPsi](#)
- [myPsi](#)
- [myPsi](#)
- [myPsi](#)
- [myPsi](#) *Data members:*
- [myPsi](#)
- [myPsuedoNoiseSpectrumInUse](#)
- [myPulseWidth](#)
- [myPulseWidth](#)
- [myRadioEntityType](#)
- [myRealTimeLastUpdate](#)
- [myRecSet](#)
- [myReceivedPower](#)
- [myReceiverId](#)
- [myReceiverRadioNumber](#)
- [myReceiverState](#)
- [myRecordBuffer](#)
- [myRecordCount](#)
- [myRecordLength](#)
- [myRecordLength](#)
- [myRecordType](#)
- [myRecordType](#) *Data members:*
- [myRefRec](#)
- [myReferencePoint](#) *Data members:*
- [myReferenceSystem](#)
- [myRelOrAbs](#)
- [myRelativePosition](#)
- [myRelativeSpotLocation](#)
- [myRemoteSites](#) *DtList of DtInetAddrs (which do not need to be deleted)*

- [myRep](#)
- [myRep](#)
- [myRestStatus](#)
- [myRotThresh](#)
- [myRotThresh](#)
- [myRotVel](#)
- [myRotationError](#)
- [myRotationalVelocity](#)
- [mySampleType](#) *Data members:*
- [mySaveBuffCount](#)
- [mySaveRecCount](#) *These are used by recordSetOffset() to save the value and index of the last offset which was computed in case it can be used again*
- [myScale](#) *parameter*
- [myScanPattern](#)
- [mySecondOperational](#)
- [mySecondaryAmmunition](#)
- [mySecondaryAmmunition](#)
- [mySecondaryAmmunition](#)
- [mySecondaryAmmunition](#)
- [mySegmentDepth](#)
- [mySegmentHeight](#)
- [mySegmentLength](#)
- [mySegmentNumber](#) *Data members:*
- [mySegmentWidth](#)
- [mySendFinalInDtor](#)
- [mySenderId](#) *parameter*
- [mySerialNumber](#)
- [myShaftRPM_RateOfChange](#)
- [mySilentAggregates](#)
- [mySilentEntities](#)
- [mySimTimeLastUpdate](#)
- [mySimTimeLastUpdate](#)
- [mySimTimeOrientDataValid](#)
- [mySimTimePosDataValid](#)
- [mySimulationAddress](#)
- [mySimulationAddress](#)
- [mySimulationAddress](#)
- [mySite](#) *Data members:*
- [mySizeLength](#)

- [mySizeOffset](#)
- [mySmoothCapable](#)
- [mySmoothCapable](#)
- [mySmoother](#) *holds what was most recently passed to useSmoother*
- [mySockList](#)
- [mySockList](#)
- [mySocketMgr](#)
- [mySockfd](#)
- [mySorient](#)
- [mySpecList](#)
- [mySpecificAppearance](#)
- [mySpos](#)
- [mySpotLinearAcceleration](#)
- [myStateRepository](#)
- [myStateUpdateCallbacks](#)
- [myStation](#)
- [myStationList](#) *zero terminated list that should always be of length myNumParts + 1*
- [myStatus](#) *Data members:*
- [myStr](#)
- [myStr](#)
- [myStr](#)
- [myString](#)
- [myString](#)
- [myStringRep](#)
- [mySubAggs](#)
- [mySubcategory](#)
- [mySupplementalFuelStatus](#)
- [mySupplementalFuelStatus](#)
- [mySweepSync](#)
- [mySweepSync](#)
- [mySystemID](#)
- [mySystemId](#)
- **myTau :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [myTauInv](#)
- [myTheta](#)
- [myTheta](#)
- [myTheta](#)

- [myTheta](#)
- [myTheta](#)
- [myTheta](#)
- [myTheta](#) *about reference z*
- [myThreshMode](#)
- [myThresholder](#)
- [myTime](#) *parameter*
- [myTime](#) *parameter*
- [myTimeForSets](#)
- [myTimeHopInUse](#)
- [myTimeSet](#)
- [myTimeStampType](#)
- [myTimeThreshold](#)
- [myTimeThreshold](#)
- [myTimeout](#) *parameter*
- [myTimeoutInterval](#)
- [myTimeoutProcessingFlag](#)
- [myTraceRotThresh](#)
- [myTraceRotThresh](#)
- [myTrackJamTargets](#)
- [myTransThresh](#)
- [myTransThresh](#)
- [myTransmitId](#)
- [myTransmitState](#)
- [myTransmitterRadioNumber](#)
- [myTurretAzimuth](#)
- [myTurretAzimuth](#)
- [myTurretSlewRate](#)
- [myTurretSlewRate](#)
- [myType](#)
- [myType](#) *Data members:*
- [myTypeList](#) *NULL if myNumArtParts = 0, otherwise has length = myNumParts*
- [myUserData](#)
- [myUsingDfltAggDim](#)
- [myUsingDfltFrac](#)
- [myUsingDfltMode](#)
- [myUsingDfltRot](#)
- [myUsingDfltRot](#)

- [myUsingDfltTrans](#)
- [myUsingDfltTrans](#)
- [myUsr](#)
- [myUtmToGeocMatrix](#)
- [myVals](#) *myVals is an array of DtMAX_PARAMS_PER_PART + 1 items which are themselves pointers*
- [myVariableDatumRecords](#)
- [myVector](#)
- [myVector](#)
- [myVector](#)
- [myVector](#)
- [myVector](#)
- [myVector](#)
- [myVector](#)
- [myVelAtTimeOfValidity](#)
- [myVelocity](#)
- [myVelocity](#)
- [myVelocity](#)
- [myVerticalDeflection](#)
- [myVerticalDeviation](#)
- [myVerticalDeviation](#)
- [myVerticalError](#)
- [myWaterStatus](#)
- [myWorldLocation](#)
- [myX](#) *Data members:*
- [myX](#) *Data members:*
- [myX](#) *Data members:*
- [myX_Offset](#)
- [myX_Offset](#)
- [myX_Offset](#)
- [myX_Offset](#)
- [myX_Offset](#)
- [myXi](#)
- [myY](#)
- [myY](#)
- [myY](#)
- [myY_Offset](#)
- [myY_Offset](#)
- [myY_Offset](#)

- [myY_Offset](#)
- [myY_Offset](#)
- [myZ](#)
- [myZ](#)
- [myZ_Offset](#)
- [myZ_Offset](#)
- [myZ_Offset](#)
- [myZ_Offset](#)
- [myZ_Offset](#)
- [myZ_Offset](#)
- [mysock](#)
- [n](#) *radians per sec*
- [nSizeMismatchComplain](#)
- [nSizeMismatch](#)
- [needT1data](#)
- [needT2data](#)
- [negAckRetVal](#)
- [netSock](#)
- [nextPdu](#)
- **nomenclature :**
 - in [duplicate](#)
 - in [DtNetRadioEntityType](#)
- **nomenclatureVersion :**
 - in [duplicate](#)
 - in [DtNetRadioEntityType](#)
- [notifyPending](#)
- [o0](#)
- [ofd](#)
- **offset :**
 - in [DtCoordTransform](#) *these were protected, but it is often more efficient to set these by passing them to another function*
 - in [DtCoordTranslation](#)
- [ok](#)
- [ok](#)
- [optarg](#)
- [opterr](#)
- [optind](#)
- [ori](#)
- [os](#)
- [outBuf](#)

- [outBuff](#)
- [outSize](#)
- [p0](#)
- [p1](#)
- [param](#)
- [params](#)
- [pat](#)
- [pdu](#)
- [pduInterest](#)
- **phi :**
 - in [duplicate](#)
 - in [DtNetEulerAngles](#) *Counter clockwise about y*
- [pktBuff8](#)
- [pktBuff8](#)
- [pkts_rcvd](#)
- [ps](#)
- [psMsgTimeout](#)
- [psi](#)
- [pulseRepetitionFrequency](#)
- [pulseRepetitionFrequency](#)
- [pulseWidth](#)
- [pulseWidth](#)
- [rLoc](#)
- [read0](#)
- [read](#)
- [reader](#)
- [recvFd](#)
- [rep](#)
- **rep :**
 - in [DtNet32Vector](#)
 - in [DtNet64Vector](#)
- [reqId](#)
- [retId](#)
- [rotation](#) *"to" origin in base coords*
- [rvel](#)
- [sec](#)
- [semiMinor](#)
- [sendFd](#)
- [servId](#)

- [serverPort](#)
- [serverSockfd](#)
- [setFcn](#)
- [shmAddr](#)
- [sizeLeft](#)
- [skt_num](#)
- [sourceAddr](#)
- **supType :**
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [supType](#)
- **t0 :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [t1](#)
- **t2 :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [tar](#)
- [tar](#)
- [testOp](#)
- **theAngles :**
 - in [DtAbsoluteVCPdu](#)
 - in [DtMimicVCPdu](#)
 - in [DtTetherVCPdu](#)
- [theAnyInstancesFlag](#)
- [theBurst](#)
- [theBurst](#)
- [theDfltAggDimThresh](#)
- [theDfltFracThresh](#)
- [theDfltHtau](#)
- [theDfltRotThresh](#)
- [theDfltRotThresh](#)
- [theDfltSmoothPeriod](#)
- **theDfltTau :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [theDfltTauInv](#)
- [theDfltThreshMode](#)

- [theDfltTimeThreshold](#)
- [theDfltTimeThreshold](#)
- [theDfltTraceRotThresh](#)
- [theDfltTraceRotThresh](#)
- [theDfltTransThresh](#)
- [theDfltTransThresh](#)
- [theGuise](#)
- [theHourFracPerHour](#)
- [theHourPerHourFrac](#)
- [theld](#)
- [theld](#)
- [theld](#)
- [theld](#)
- **theld :**
 - in [DtMimicTrackVCPdu](#)
 - in [DtMimicVCPdu](#)
 - in [DtModelessVCPdu](#)
 - in [DtOrbitVCPdu](#)
- [theld](#)
- **theld :**
 - in [DtRepairCompletePdu](#)
 - in [DtRepairResponsePdu](#)
- **theld :**
 - in [DtResupplyCancelPdu](#)
 - in [DtResupplyOfferPdu](#)
 - in [DtResupplyReceivedPdu](#)
- [theld](#)
- **theld :**
 - in [DtTetherTrackVCPdu](#)
 - in [DtTetherVCPdu](#)
 - in [DtTrackVCPdu](#)
- [theld](#)
- **theMasterSmoothEnable :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [theMaxInsertionStringSize](#)
- [theMaxMarkingSize](#)
- [theMaxPduKinds](#)
- [theNumBucketsDefault](#)

- [theSeparator](#)
- [theStaticSize](#) *Size of PDU with zero data bytes*
- [theStaticSize](#)
- [theStaticSize](#)
- [theStaticSize](#) *Size of PDU with zero data bytes*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero data bytes*
- [theStaticSize](#)
- [theStaticSize](#)
- [theStaticSize](#)
- [theStaticSize](#)
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero data bytes*
- [theStaticSize](#)
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero data bytes*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero data bytes*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero data bytes*
- [theStaticSize](#) *Size of PDU with zero data bytes*
- [theStaticSize](#)
- [theStaticSize](#)
- [theStaticSize](#) *Size of PDU with zero data bytes*
- [theStaticSize](#)
- [theStaticSize](#)

- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theStaticSize](#) *Size of PDU with zero data bytes*
- [theStaticSize](#) *Size of PDU with zero parameter records*
- [theType](#)
- **theVec :**
 - in [DtAbsoluteVCPdu](#)
 - in [DtMimicTrackVCPdu](#)
 - in [DtMimicVCPdu](#)
 - in [DtTetherTrackVCPdu](#)
 - in [DtTetherVCPdu](#)
 - in [DtTrackVCPdu](#)
- [theta](#)
- [thetaDot](#)
- [theta](#) *Counter clockwise about z*
- [to](#)
- [transId](#)
- [type](#)
- [udpPort](#)
- [unbundleBuff8](#)
- [unbundling](#)
- **usr :**
 - in [DtSmoothOrient](#)
 - in [DtSmoothPos](#)
- [v0](#)
- [v1](#)
- [values](#)
- [vel](#)
- [vel](#)
- [vel](#)
- [vel](#)
- [verboseFromOther](#)
- [verboseNegAck](#)
- [verboseOtherType](#)
- [verbosePosAck](#)
- [verboseToOther](#)
- [version](#)
- [version](#)
- [vs](#)
- [wLoc](#)

- [world](#)
- [ws](#)
- [wy](#)
- [wz](#)

Macros

- [ACK_INTERACTION_H](#)
- [ACTION_REQUEST_INTERACTION_H](#)
- [ACTION_RESPONSE_H](#)
- [ADD](#) *Generic Vector Sum :Ans=A+s*B*
- [AGGREGATE_PUB_H](#)
- [AGGREGATE_SR_MARKING_TEXT_LENGTH](#) *Aggregate state PDU marking is 31 bytes long, RPR marking struct is 11*
- [COLLISION_H](#)
- [COMMENT_H](#)
- [CREATEENTINTER_H](#)
- [DATA_H](#)
- [DATA_QUERY_H](#)
- [DETONATION_INTERACTION_H](#)
- [DT_CANT_CAST_TO_CONST_2DARRAY](#)
- [DT_CANT_HAVE_NET_UNIONS](#)
- [DT_FD_ZERO_USES_MEMSET](#)
- [DT_FUNCTION_MUST_RETURN](#)
- [DT_HAVE_AINT](#)
- [DT_HAVE_ARPA_INET_H](#)
- [DT_HAVE_ATT_SELECT](#) *Both of these next two should NOT be true (but both MAY be false)*
- [DT_HAVE_BSD_SELECT](#)
- [DT_HAVE_BZERO_PROTO_IN_BSTRING_H](#)
- [DT_HAVE_BZERO_PROTO_IN_CC_LIBC_H](#)
- [DT_HAVE_BZERO_PROTO_IN_MEMORY_H](#)
- [DT_HAVE_COPYSIGN](#)
- [DT_HAVE_CPP_STRINGIZE](#)
- [DT_HAVE_DAEMON](#)
- [DT_HAVE_DRAND_48](#)
- [DT_HAVE_FILE_H](#)
- [DT_HAVE_GETHOSTID](#)
- [DT_HAVE_GETHOSTID_PROTO](#)
- [DT_HAVE_GETOPT_H](#)

- [DT_HAVE_GETTIMEOFDAY](#)
- [DT_HAVE_GETTIMEOFDAY_PROTO](#)
- [DT_HAVE_MESSAGE_SOCKETS](#)
- [DT_HAVE_NET_IF_H](#)
- [DT_HAVE_NUMERIC_CONSTANTS](#)
- [DT_HAVE_PARAM_H](#)
- [DT_HAVE_PROCESS_H](#)
- [DT_HAVE_SELECT_PROTO](#)
- [DT_HAVE_SEMUN](#)
- [DT_HAVE_SOCKET_PROTOS](#)
- [DT_HAVE_SOCKIO_H](#)
- [DT_HAVE_STD_H](#)
- [DT_HAVE_STRTOUL](#)
- [DT_HAVE_SYSV_IPC](#)
- [DT_HAVE_SYSV_IPC_PROTOS](#)
- [DT_HAVE_SYS_UIO_H](#)
- [DT_HAVE_UNISTD_H](#) *these were added while doing the MSVC Dos port*
- [DT_HAVE_WRITEV_PROTO](#)
- [DT_LITTLE_ENDIAN](#)
- [DT_NEED_DLL_DECLARATIONS](#) *If building a DLL MachineTypes*
- [DT_NO_CONSTPTRPTR](#)
- [DT_NO_MULTICAST](#)
- [DT_NO_RAND](#)
- [DT_NO_RANDOM](#)
- [DtABS](#)
- [DtADD_ENUM_STRING](#)
- [DtADD_PDU_CREATOR](#) *DtADD_PDU_CREATOR adds a PDU classes static create function to a PDU factory*
- [DtART_PART_SUCCESS](#)
- [Dtatoi](#)
- [Dtatoi](#) *some versions of strtol returns INT_MAX for input larger than 0x7ffffff*
- [DtAbs](#) *Returns the absolute value of x*
- [DtAbsNotifyObserver_H](#)
- [DtAcknowledgeDecoder_H](#)
- [DtAcknowledgeEncoder_H](#)
- [DtAcknowledgeInteraction_H](#)
- [DtAcknowledgePdu_H](#) *This file provides protocol independence for Acknowledge Interactions*
- [DtAcknowledgeRPdu_H](#)
- [DtAcknowledgeStaticSize](#)

- [DtActionRequestDecoder_H](#)
- [DtActionRequestEncoder_H](#)
- [DtActionRequestInteraction_H](#)
- [DtActionRequestPdu_H](#)
- [DtActionRequestRPdu_H](#) *This file provides protocol independence for ActionRequest Interactions*
- [DtActionRequestStaticSize](#)
- [DtActionRequestToObjDecoder_H](#)
- [DtActionRequestToObjEncoder_H](#)
- [DtActionRequestToObjInter_H](#)
- [DtActionResponseDecoder_H](#)
- [DtActionResponseEncoder_H](#)
- [DtActionResponseFromObjDecoder_H](#)
- [DtActionResponseFromObjEncoder_H](#)
- [DtActionResponseFromObjInter_H](#)
- [DtActionResponseInteraction_H](#)
- [DtActionResponseRPdu_H](#) *This file provides protocol independence for ActionResponse Interactions*
- [DtActionResponseStaticSize](#)
- [DtAggregateEntityDecoder_H](#)
- [DtAggregateEntityEncoder_H](#)
- [DtAggregatePublisher_H](#)
- [DtAggregateStateDecoder_H](#)
- [DtAggregateStateEncoder_H](#)
- [DtAggregateStatePdu_H](#)
- [DtAggregateStateRepository_H](#) *This file provides protocol independence for local aggregates*
- [DtAntennaPatternRecord_H](#)
- [DtAny](#)
- [DtAppSpecificRadioSignalDecoder_H](#)
- [DtAppSpecificRadioSignalEncoder_H](#)
- [DtApproximator_H](#)
- [DtArealObjectStatePdu_H](#)
- [DtArtPartList_H](#)
- [DtArtPartSpec_H](#)
- [DtArtPartThresholder_H](#)
- [DtArtPart_H](#)
- [DtAttachedPartList_H](#)
- [DtAttributeChangeReqDec_H](#)
- [DtAttributeChangeReqEnc_H](#)
- [DtAttributeChangeReqInter_H](#)

- [DtAttributeChangeResDec_H](#)
- [DtAttributeChangeResEnc_H](#)
- [DtAttributeChangeResInter_H](#)
- [DtBAD_BEAM_ADDR](#)
- [DtBAD_BEAM_NUM](#)
- [DtBAD_SYS_ADDR](#)
- [DtBAD_SYS_NUM](#)
- [DtBAD_TARGET_ADDR](#)
- [DtBAD_TARGET_NUM](#)
- [DtBaseEntityDecoder_H](#)
- [DtBaseEntityEncoder_H](#)
- [DtBaseEntityStateRepository_H](#)
- [DtBaseExerciseConn_H](#)
- [DtBaseHashKey_H](#)
- [DtBaseMsg_H](#)
- [DtBaseReflectedAggregate_H](#)
- [DtBaseReflectedEntity_H](#)
- [DtBasicFixedWingAircraftRecord_H](#)
- [DtBasicGroundCombatSoldierRecord_H](#)
- [DtBasicGroundCombatVehicleRecord_H](#)
- [DtBasicRotorWingAircraftRecord_H](#)
- [DtBeamAntennaPatternRecord_H](#)
- [DtBindToDestAddr](#) *setting this bit in the flags constructor argument will cause the NetSocket to ignore all traffic (incl*
- [DtCartesianView_H](#)
- [DtClark1866Spheroid](#)
- [DtCollisionDecoder_H](#)
- [DtCollisionElasticPdu_H](#)
- [DtCollisionEncoder_H](#)
- [DtCollisionInteraction_H](#)
- [DtCommentDecoder_H](#)
- [DtCommentEncoder_H](#)
- [DtCommentInteraction_H](#)
- [DtCommentPdu_H](#) *This file provides protocol independence for Comment Interactions*
- [DtCommentRPdu_H](#)
- [DtCommentStaticSize](#)
- [DtCreateEntityDecoder_H](#)
- [DtCreateEntityEncoder_H](#)
- [DtCreateEntityInteraction_H](#)

- [DtCreateEntityPdu_H](#) *This file provides protocol independence for CreateEntity Interactions*
- [DtCreateEntityRPdu_H](#)
- [DtCreateObjReqDecoder_H](#)
- [DtCreateObjReqEncoder_H](#)
- [DtCreateObjReqInter_H](#)
- [DtCreateObjectResultDecoder_H](#)
- [DtCreateObjectResultEncoder_H](#)
- [DtCreateObjectResultInter_H](#)
- [DtCreateStaticSize](#)
- [DtCsFlagsDefault](#)
- [DtCsQDefaultSize](#)
- [DtDATUM_BAD_ID](#)
- [DtDATUM_BAD_INDEX](#)
- [DtDATUM_BAD_LENGTH](#)
- [DtDATUM_VALID](#)
- [DtDEFAULT_PORT](#)
- [DtDELETE](#)
- [DtDInteraction_H](#)
- [DtDataDecoder_H](#)
- [DtDataEncoder_H](#)
- [DtDataInteraction_H](#)
- [DtDataPdu_H](#) *This file provides protocol independence for Comment Interactions*
- [DtDataQueryDecoder_H](#)
- [DtDataQueryEncoder_H](#)
- [DtDataQueryInteraction_H](#)
- [DtDataQueryPdu_H](#)
- [DtDataQueryRPdu_H](#) *This file provides protocol independence for ActionResponse Interactions*
- [DtDataRPdu_H](#)
- [DtDataStaticSize](#)
- [DtDatabaseIndexRadioSignalDecoder_H](#)
- [DtDatabaseIndexRadioSignalEncoder_H](#)
- [DtDatumMap_H](#)
- [DtDatumRecordSet_H](#)
- [DtDcm_H](#)
- [DtDeadReckoner_H](#)
- [DtDeg2Rad](#)
- [DtDesignatorPdu_H](#)
- [DtDesignatorPublisher_H](#)

- [DtDesignatorRepository_H](#)
- [DtDesignatorStaticSize](#)
- [DtDesignatorStaticSizeVERS4](#)
- [DtDesignatorStaticSizeVERS5](#)
- [DtDetonateStaticSize](#)
- [DtDetonationDecoder_H](#)
- [DtDetonationEncoder_H](#)
- [DtDetonationInteraction_H](#)
- [DtDirectedRotRate_H](#)
- [DtDisAggregatePublisher_H](#)
- [DtDisEntityPublisher_H](#)
- [DtDisExerciseConn_H](#)
- [DtDisPduFactory_H](#)
- [DtDiscoverObjectCbInfo_H](#)
- [DtECenti](#)
- [DtEDeca](#)
- [DtEDeci](#)
- [DtEGiga](#)
- [DtEHalf](#)
- [DtEHecto](#)
- [DtEKilo](#)
- [DtEMISSION_SUCCESS](#)
- [DtEMega](#)
- [DtEMicro](#)
- [DtEMilli](#)
- [DtENano](#)
- [DtENineTenths](#)
- [DtEOne](#)
- [DtEPico](#) *A selection of constants useful as tolerance and threshold levels, etc*
- [DtEXTEND_ATTR_ENUM](#) *This macro creates definitions that look like this: const DtAppearAttribute DtAppDensity = DtAppearAttribute(16 << DtAppDensityBitShift);*
- [DtEXTEND_ATTR_VAL_ENUM](#)
- [DtEXTEND_BIT_SHIFT_ENUM](#) *This macro creates definitions that look like this: const DtAppearAttributeBitShift DtAppDensityBitShift = DtAppearAttributeBitShift(16);*
- [DtEXTEND_ENUM](#) *This file should contain only enumeration definitions*
- [DtElectromagneticEmissionPdu](#)
- [DtElectromagneticEmissionPdu_H](#)
- [DtElectromagneticEmissionStaticSize](#)
- [DtEmbeddedSystemRepository_H](#)

- [DtEmissionPdu_H](#)
- [DtEmissionStaticSize](#)
- [DtEmitterBeamRepository_H](#)
- [DtEmitterSystemRepository_H](#)
- [DtEmptyFomMapper_H](#)
- [DtEncodedAudioRadioSignalDecoder_H](#)
- [DtEncodedAudioRadioSignalEncoder_H](#)
- [DtEnhancedFixedWingAircraftRecord_H](#)
- [DtEnhancedGroundCombatSoldierRecord_H](#)
- [DtEnhancedGroundCombatVehicleRecord_H](#)
- [DtEnhancedRotorWingAircraftRecord_H](#)
- [DtEntityEncoder_H](#)
- [DtEntityIsAntiPersonnel](#)
- [DtEntityPublisher_H](#)
- [DtEntityStateDecoder_H](#)
- [DtEntityStatePdu_H](#)
- [DtEntityStateRepository_H](#) *This file provides protocol independence for Local Entities*
- [DtEntityStateStaticSize](#)
- [DtEntityStateUpdatePdu_H](#)
- [DtEnvironmentEntityDecoder_H](#)
- [DtEnvironmentEntityEncoder_H](#)
- [DtEnvironmentTypeRecord_H](#)
- [DtEnvironmentalProcessPdu_H](#)
- [DtEq](#) *A macro to make use of these tolerances*
- [DtEventId_H](#)
- [DtEventReportDecoder_H](#)
- [DtEventReportEncoder_H](#)
- [DtEventReportInteraction_H](#)
- [DtEventReportPdu_H](#) *This file provides protocol independence for EventReport Interactions*
- [DtEventReportStaticSize](#)
- [DtEventReptRPdu_H](#)
- [DtExerciseConn_H](#)
- [DtFIXED](#) *For backwards compatibility:*
- [DtFREQ_LOWER](#)
- [DtFREQ_UPPER](#)
- [DtFedReader_H](#)
- [DtFedTarget_H](#)
- [DtFilename_H](#)

- [DtFireDecoder_H](#)
- [DtFireEncoder_H](#)
- [DtFireInteraction_H](#)
- [DtFirePdu_H](#) *This file provides protocol independence for Fire Interactions*
- [DtFireStaticSize](#)
- [DtFomMapper_H](#)
- [DtFom_H](#)
- [DtGenericObjectPublisher_H](#)
- [DtGenericStateRepository_H](#)
- [DtGeodeticCoord_H](#)
- [DtGlobalIdHashlist_H](#)
- [DtGlobalObjectDesignatorList_H](#)
- [DtGlobalObjectDesignator_H](#)
- [DtGridAxisRecordFixed_H](#)
- [DtGridAxisRecordVariable_H](#)
- [DtGridDataRecordType0_H](#)
- [DtGridDataRecordType1_H](#)
- [DtGridDataRecordType2_H](#)
- [DtGridDataRecord_H](#)
- [DtGriddedDataPdu_H](#)
- [DtGroundLogisticsVehicleRecord_H](#)
- [DtGroupSocket_H](#)
- [DtHInteraction_H](#)
- [DtHashItem_H](#)
- [DtHashlist_H](#)
- [DtHlaObjectFactory_H](#)
- [DtHlaObjectWithStateRep_H](#)
- [DtHlaObject_H](#)
- [DtHlaStateDecoder_H](#)
- [DtHlaStateEncoder_H](#)
- [DtIGNORE_AZIMUTH_RATE](#)
- [DtIGNORE_BODY_HEADING](#)
- [DtIGNORE_BODY_PITCH](#)
- [DtIGNORE_BODY_ROLL](#)
- [DtIGNORE_EAST](#)
- [DtIGNORE_EAST_OFFSET](#)
- [DtIGNORE_FORWARD](#)
- [DtIGNORE_GROUP_IDS](#)

- [DtIGNORE_HEADING](#)
- [DtIGNORE_HEADING_OFFSET](#)
- [DtIGNORE_HEIGHT](#)
- [DtIGNORE_HEIGHT_OFFSET](#)
- [DtIGNORE_NORTH](#)
- [DtIGNORE_NORTH_OFFSET](#)
- [DtIGNORE_PITCH](#)
- [DtIGNORE_PITCH_OFFSET](#)
- [DtIGNORE_PRIMARY_ID](#)
- [DtIGNORE_RADIAL_OFFSET](#)
- [DtIGNORE_RADIUS](#)
- [DtIGNORE_RIGHT](#)
- [DtIGNORE_ROLL](#)
- [DtIGNORE_SECONDARY_ID](#)
- [DtIGNORE_UP](#)
- [DtINCONSISTENT_BEAM_LENGTHS_SYS_LENGTH](#)
- [DtINCONSISTENT_NUM_TARGETS_BEAM_LENGTH](#)
- [DtINCONSISTENT_SYS_LENGTHS_PDU_LENGTH](#)
- [DtINCREMENT](#)
- [DtISOFF](#)
- [DtISON](#)
- [DtIffAtcNavAidsBeamData_H](#)
- [DtIffAtcNavAidsLayerHeader_H](#)
- [DtIffAtcNavAidsOperational_H](#)
- [DtIffAtcNavAidsParameter_H](#)
- [DtIffAtcNavAidsPdu_H](#)
- [DtIffAtcNavAidsSecondOperational_H](#)
- [DtIffAtcNavAidsSystemID_H](#)
- [DtIntHashKey_H](#) *This file is used as a convenience to group interaction concepts together*
- [DtInterClassDesc_H](#)
- [DtInteractionDecoderFactory_H](#)
- [DtInteractionDecoder_H](#)
- [DtInteractionEncoderFactory_H](#)
- [DtInteractionEncoder_H](#)
- [DtInteractionFactory_H](#)
- [DtInteractionManager_H](#)
- [DtInteractionWithData_H](#)
- [DtInteractionWithEncDec_H](#)

- [DtIntercomControlPdu_H](#)
- [DtIntercomEntityDestRec_H](#)
- [DtIntercomGroupAsgmtRec_H](#)
- [DtIntercomGroupDestRec_H](#)
- [DtIntercomParmRec_H](#)
- [DtIntercomSignalPdu_H](#)
- [DtInternationalSpheroid](#)
- [DtIsAirPlatform](#)
- [DtIsFixedWing](#)
- [DtIsGroupOfPdu_H](#)
- [DtIsLandLifeForm](#)
- [DtIsLandPlatform](#)
- [DtIsPartOfPdu_H](#)
- [DtIsRotaryWing](#)
- [DtIsTank](#)
- [DtLIMIT](#)
- [DtLaserPdu_H](#)
- [DtLaserStaticSize](#)
- [DtLeAppearancePdu_H](#)
- [DtLeArticulatedPartsPdu_H](#)
- [DtLeDeadReckoningParameters_H](#)
- [DtLeDetonationPdu_H](#)
- [DtLeEntityID_H](#)
- [DtLeEntityLinearVelocity_H](#)
- [DtLeEntityLocationEntityCoordinates_H](#)
- [DtLeEntityLocationWorldCoordinates_H](#)
- [DtLeEntityOrientation_H](#)
- [DtLeFirePdu_H](#)
- [DtLeOrientationError_H](#)
- [DtLePositionError_H](#)
- [DtLeTspiPdu_H](#)
- [DtLgrControlDecoder_H](#)
- [DtLgrControlEncoder_H](#)
- [DtLgrControlInteraction_H](#)
- [DtLgrControlPdu_H](#) *This file provides protocol independence for LgrControl Interactions*
- [DtLgrEndPdu_H](#)
- [DtLgrEofActionPdu_H](#)
- [DtLgrHeartbeatPdu_H](#)

- [DtLgrPausePdu_H](#)
- [DtLgrPbFilePdu_H](#)
- [DtLgrPlayActionPdu_H](#)
- [DtLgrPlayPdu_H](#)
- [DtLgrQuitPdu_H](#)
- [DtLgrRecFilePdu_H](#)
- [DtLgrRecordActionPdu_H](#)
- [DtLgrRecordPdu_H](#)
- [DtLgrSetTimePdu_H](#)
- [DtLgrSkipTimePdu_H](#)
- [DtLgrStartPdu_H](#)
- [DtLgrStopPdu_H](#)
- [DtLgrTimeScalePdu_H](#)
- [DtLgrTimeoutPdu_H](#)
- [DtLinearObjectStatePdu_H](#)
- [DtLinearSegmentParameter_H](#)
- [DtList_H](#)
- [DtLocalHlaObjectManager_H](#)
- [DtMAX](#)
- [DtMAX_PDU_KINDS](#)
- [DtMIN](#)
- [DtMaxEntityMarkingLength](#)
- [DtMaxExercisId](#)
- [DtMaxPacketSize](#) *IP + UDP headers are 20 + 8*
- [DtMessagePdu_H](#)
- [DtMessageStaticSize](#)
- [DtMilitaryEntityDecoder_H](#)
- [DtMilitaryEntityEncoder_H](#)
- [DtMilitaryPlatformEntityDecoder_H](#)
- [DtMilitaryPlatformEntityEncoder_H](#)
- [DtMinefieldDataPdu_H](#)
- [DtMinefieldPerimeterRecord_H](#)
- [DtMinefieldQueryPdu_H](#)
- [DtMinefieldResponseNackPdu_H](#)
- [DtMinefieldStatePdu_H](#)
- [DtMunitionEntityDecoder_H](#)
- [DtMunitionEntityEncoder_H](#)
- [DtNET_READ_BAD_PACKET](#)

- [DtNET_READ_FILTERED_PACKET](#)
- [DtNET_READ_NO_PACKETS](#)
- [DtNET_READ_SERVER_MSG](#)
- [DtNET_READ_SIZE_MISMATCH](#)
- [DtNET_READ_SUCCESS](#)
- [DtNO_SUCH_ART_PART](#)
- [DtNUMBER](#)
- [DtNeg](#) *Returns x if x is less than zero*
- [DtNetBoolean_H](#)
- [DtNetTimeStamp_H](#) *example device names are "ec0" and "le0"*
- [DtNIFatal](#) *Notification Levels*
- [DtNIInfo](#)
- [DtNIVerbose](#)
- [DtNIWarn](#)
- [DtNoArtParts](#)
- [DtNullArtPartSpec](#)
- [DtNullDatumParam](#)
- [DtNullFederateAmbassador_H](#) *Adapted from the DMSO RTI 1*
- [DtNullVarDatumSpec](#)
- [DtOFFSET](#)
- [DtObjClassDesc_H](#)
- [DtObjectIdHashKey_H](#)
- [DtObjectIdHashlist_H](#)
- [DtObjectReflectAttributeValuesCbInfo_H](#)
- [DtObjectType_H](#)
- [DtPROCESS_PDU_NO_PDU](#)
- [DtPROCESS_PDU_SUCCESS](#)
- [DtPROCESS_PDU_WRONG_EXID](#)
- [DtPduCallbackInfo_H](#)
- [DtPduCallbackManager_H](#)
- [DtPduDatumRecordSet_H](#)
- [DtPduFactory_H](#)
- [DtPduWithData_H](#) *DMO 2*
- [DtPdu_H](#)
- [DtPhysicalEntityDecoder_H](#)
- [DtPhysicalEntityEncoder_H](#)
- [DtPointObjectStatePdu_H](#)
- [DtPos](#) *Returns x if greater than zero*

- [DtPropulsionSystem_H](#)
- [DtQueryStaticSize](#)
- [DtQuiet](#)
- [DtRad2Deg](#)
- [DtRadioOther](#)
- [DtRadioReceiverRepository_H](#)
- [DtRadioTransmitterRepository_H](#)
- [DtRawBinaryRadioSignalDecoder_H](#)
- [DtRawBinaryRadioSignalEncoder_H](#)
- [DtRead](#)
- [DtReadWrite](#)
- [DtReceiveInteractionCbInfo_H](#)
- [DtReceiverPdu_H](#)
- [DtReceiverStaticSize](#)
- [DtRecordPdu_H](#)
- [DtRecordQueryPdu_H](#)
- [DtRecordSetPdu_H](#)
- [DtRecordSet_H](#)
- [DtReflectAttributeValuesCbInfo_H](#)
- [DtReflectAttributeValuesCbList_H](#)
- [DtReflectedAggregateList_H](#)
- [DtReflectedAggregate_H](#)
- [DtReflectedDesignatorList_H](#)
- [DtReflectedDesignator_H](#)
- [DtReflectedEmitterBeamList_H](#)
- [DtReflectedEmitterSystemList_H](#)
- [DtReflectedEmitterSystem_H](#)
- [DtReflectedEntityList_H](#)
- [DtReflectedEntity_H](#)
- [DtReflectedGenericObjectList_H](#)
- [DtReflectedGenericObject_H](#) *This file provides protocol independence for reflected emitter system lists*
- [DtReflectedHlaObjectManager_H](#) *This file provides protocol independence for Reflected Object Lists*
- [DtReflectedObjectCallbackInfo_H](#)
- [DtReflectedObjectList_H](#)
- [DtReflectedObject_H](#)
- [DtReflectedObject_H](#)
- [DtReflectedRadioReceiverList_H](#)
- [DtReflectedRadioTransmitterList_H](#)

- [DtReflectedRadioTransmitter_H](#)
- [DtRegControlCbInfo_H](#)
- [DtRegControlCbList_H](#)
- [DtRemoveEntityDecoder_H](#)
- [DtRemoveEntityEncoder_H](#)
- [DtRemoveEntityInteraction_H](#)
- [DtRemoveEntityPdu_H](#)
- [DtRemoveEntityRPdu_H](#)
- [DtRemoveObjectCbInfo_H](#) *This file provides protocol independence for RemoveEntity Interactions*
- [DtRemoveObjectRequestDecoder_H](#)
- [DtRemoveObjectRequestEncoder_H](#)
- [DtRemoveObjectRequestInter_H](#)
- [DtRemoveObjectResultDecoder_H](#)
- [DtRemoveObjectResultEncoder_H](#)
- [DtRemoveObjectResultInter_H](#)
- [DtRemoveStaticSize](#)
- [DtRepairCompleteDecoder_H](#)
- [DtRepairCompleteEncoder_H](#)
- [DtRepairCompleteInteraction_H](#)
- [DtRepairCompleteStaticSize](#)
- [DtRepairResponseDecoder_H](#)
- [DtRepairResponseEncoder_H](#)
- [DtRepairResponseInteraction_H](#)
- [DtRepairResponsePdu_H](#)
- [DtRepairResponseStaticSize](#)
- [DtResupplyCancelDecoder_H](#)
- [DtResupplyCancelEncoder_H](#)
- [DtResupplyCancelInteraction_H](#)
- [DtResupplyCancelPdu_H](#) *This file provides protocol independence for the Resupply Interactions: ResupplyCancel, ResupplyOffer, ResupplyReceived*
- [DtResupplyCancelStaticSize](#)
- [DtResupplyOfferDecoder_H](#)
- [DtResupplyOfferEncoder_H](#)
- [DtResupplyOfferInteraction_H](#)
- [DtResupplyOfferStaticSize](#)
- [DtResupplyRecInteraction_H](#)
- [DtResupplyReceivedDecoder_H](#)
- [DtResupplyReceivedEncoder_H](#)
- [DtResupplyReceivedStaticSize](#)

- [DtRprFomMapper_H](#)
- [DtRprHlaObjectFactory_H](#)
- [DtRprInteractionDecoderFactory_H](#)
- [DtRprInteractionEncoderFactory_H](#)
- [DtRprInteractionFactory_H](#)
- [DtRprStateDecoderFactory_H](#)
- [DtRprStateEncoderFactory_H](#)
- [DtSIZEOF](#)
- [DtSOCKET_FILTERED_PACKET](#)
- [DtSOCKET_NO_PACKET](#)
- [DtSOCKET_PACKET](#)
- [DtSOCKET_SERVER_MSG](#)
- [DtSQR](#)
- [DtSTATUS_NOT_OK](#)
- [DtSTATUS_OK](#)
- [DtSUPPLY_BAD_INDEX](#)
- [DtSUPPLY_VALID](#)
- [DtSameEntityKind](#)
- [DtServiceRequestDecoder_H](#)
- [DtServiceRequestEncoder_H](#)
- [DtServiceRequestInteraction_H](#)
- [DtServiceRequestPdu_H](#) *This file provides protocol independence for ActionResponse Interactions*
- [DtServiceRequestStaticSize](#)
- [DtSetDataDecoder_H](#)
- [DtSetDataEncoder_H](#)
- [DtSetDataInteraction_H](#)
- [DtSetDataPdu_H](#) *This file provides protocol independence for ActionResponse Interactions*
- [DtSetDataRPdu_H](#)
- [DtSetDataStaticSize](#)
- [DtSetRecordPdu_H](#)
- [DtSetVec](#) *Vector routines*
- [DtSgn](#) *Returns the sign of x*
- [DtSignalInteraction_H](#)
- [DtSignalPdu_H](#)
- [DtSignalStaticSize](#)
- [DtSilentAggregateList_H](#)
- [DtSilentEntityList_H](#)
- [DtSimanPdu_H](#)

- [DtSimpleRprFomMapper_H](#)
- [DtSmoothOriWithDr_H](#)
- [DtSmoothPosWithDr_H](#)
- [DtSmoothPos_H](#)
- [DtSmoother_H](#) *This file provides protocol independence for Signal Interactions*
- [DtSoldierDecoder_H](#)
- [DtSoldierEncoder_H](#)
- [DtSquare](#) *Returns the square of x*
- [DtStartResumeDecoder_H](#)
- [DtStartResumeEncoder_H](#)
- [DtStartResumeInteraction_H](#)
- [DtStartResumePdu_H](#) *This file provides protocol independence for StartResume Interactions*
- [DtStartResumeRPdu_H](#)
- [DtStartStaticSize](#)
- [DtStartStopInterCbInfo_H](#)
- [DtStartStopInterCbList_H](#)
- [DtStateDecoderFactory_H](#)
- [DtStateEncoderFactory_H](#)
- [DtStateRepository_H](#) *This file provides protocol independence for state messages*
- [DtStopFreezeDecoder_H](#)
- [DtStopFreezeEncoder_H](#)
- [DtStopFreezeInteraction_H](#)
- [DtStopFrzRPdu_H](#)
- [DtStopPdu_H](#) *This file provides protocol independence for StopFreeze Interactions*
- [DtStopStaticSize](#)
- [DtStringHashKey_H](#)
- [DtString_H](#)
- [DtSuppEmisEntStatePdu_H](#)
- [DtSystemHashKey_H](#)
- [DtTURNOFF](#)
- [DtTURNON](#)
- [DtTYPE_BYTE_ARRAY](#)
- [DtTYPE_FLOAT32](#)
- [DtTYPE_FLOAT64](#)
- [DtTYPE_INT32](#)
- [DtTYPE_NOT_DEFINED](#)
- [DtTYPE_STRING](#)
- [DtTYPE_UNSIGNED32](#)

- [DtTaitBryan_H](#)
- [DtTcpForwarder_H](#)
- [DtTcpSocketMgr_H](#)
- [DtTcpSocket_H](#)
- [DtThresholder_H](#)
- [DtTimeStampType_H](#)
- [DtTopoView_H](#)
- [DtTrackJamDesignatorList_H](#)
- [DtTrackJamDesignator_H](#)
- [DtTransferControlRequestPdu_H](#)
- [DtTransmitterPdu_H](#)
- [DtTransmitterStaticSize](#)
- [DtUSE_DEFAULT_IP_ADDR](#)
- [DtUSE_INTERNAL_BUFFER](#)
- [DtUSE_INTERNAL_BUFFER_REP](#)
- [DtUnderwaterAcousticAddtnlPassiveAct_H](#)
- [DtUnderwaterAcousticBeam_H](#)
- [DtUnderwaterAcousticEmitterSystem_H](#)
- [DtUnderwaterAcousticPdu_H](#)
- [DtUnderwaterAcousticShaftSpeed_H](#)
- [DtUnknownHlaObject_H](#)
- [DtUnknownInteraction_H](#)
- [DtUnknownPdu_H](#)
- [DtUnknownStaticSize](#)
- [DtUpdateControlCbInfo103_H](#)
- [DtUpdateControlCbInfo_H](#)
- [DtUpdateControlCbList103_H](#)
- [DtUpdateControlCbList_H](#)
- [DtUtmCoord_H](#)
- [DtUtmView_H](#)
- [DtVAR](#)
- [DtVariableDatumRecord_H](#)
- [DtVector_H](#)
- [DtVectoringNozzleSystem_H](#)
- [DtViewControlDecoder_H](#)
- [DtViewControlEncoder_H](#)
- [DtViewControlInteraction_H](#)
- [DtViewControlKind](#) *backwards compatible*

- [DtViewControlPdu_H](#) *This file provides protocol independence for ViewControl Interactions*
- [DtVrIFedTime_H](#) *This file is based on MyFedTime*
- [DtVrIFederateAmbassador_H](#)
- [DtWGS84Spheroid](#) *Here are a few of the most important spheroids; we should put all 13 in*
- [DtWrite](#)
- [ENTITY_PUB_H](#)
- [EVENTREPORT_H](#)
- [EXERCISE_CONNECTION_H](#)
- [Euler_H](#)
- [FIRE_INTERACTION_H](#) *This file provides protocol independence for exercise connections*
- [IDENTITY](#)
- [INTERACTION_H](#)
- [LGR_CONTROL_INTERACTION_H](#)
- [LIMIT](#)
- [MAX](#)
- [MIN](#)
- [M_1_PI](#)
- [M_2_PI](#)
- [M_2_SQRTPI](#)
- [M_E](#)
- [M_LN10](#)
- [M_LN2](#)
- [M_LOG10E](#)
- [M_LOG2E](#)
- [M_PI](#)
- [M_PI_2](#)
- [M_PI_4](#)
- [M_SQRT2](#)
- [MathTypes_H](#)
- [NULL](#)
- [PDUS_H](#)
- [REFLECTED_EMITTER_H](#) *This file provides protocol independence for Reflected Objects*
- [REFLECTED_EMITTER_LIST_H](#)
- [REMOTE_AGGREGATE_H](#) *This file provides protocol independence for remote aggregate lists*
- [REMOTE_AGGREGATE_LIST_H](#)
- [REMOTE_ENTITY_H](#) *This file provides protocol independence for reflected aggregates*
- [REMOTE_ENTITY_LIST_H](#) *This file provides protocol independence for reflected emitter systems*
- [REMOTE_OBJ_H](#) *This file provides protocol independence for Reflected Entities*

- [REMOTE_OBJ_LIST_H](#) *This file provides protocol independence for Remote Entity Lists*
- [REMOVE_ENTITY_H](#)
- [REPAIR_COMPLETE_H](#)
- [REPAIR_RESPONSE_H](#) *This file provides protocol independence for ActionResponse Interactions*
- [RESUPPLY_H](#) *This file provides protocol independence for ActionResponse Interactions*
- [REUSE_ALLOWED](#)
- [SCRIPT_H](#)
- [SERVICE_REQUEST_H](#)
- [SET_DATA_H](#)
- [SIGNAL_INTER_H](#)
- [SQR](#)
- [START_INTERACTION_H](#)
- [STATE_MESSAGE_H](#)
- [STOP_INTERACTION_H](#)
- [SUBTRACT](#)
- [TopoCoord_H](#)
- [VIEW_CONTROL_INTERACTION_H](#)
- [VLCONFIG_INCLUDED](#) *File - libsrc/md/c-NT*
- [VL_TIME_H](#)
- [ZERO](#) *generic initialize matrix routines*
- [ACKPRED_H](#)
- [CLIENTSOCKET_H](#)
- [COLLPDU_H](#)
- [DATUMIDTYPE_H](#) *This file provides protocol independence for Collision Interactions*
- [DISENUMS_H](#)
- [ENTITYTYPES_H](#)
- [ENUM2STRING_H](#)
- [HOSTSTRUCTS_H](#)
- [LibMatrix_h](#)
- [MachineTypes_h](#)
- [NETSTRUCTS_H](#)
- [NetSocket_h](#) *class NetSocket NetSocket provides several overloaded constructors that allow the user to specify UDP port, device, and destination address in any combination except port and destination address together*
- [NetTypes](#)
- [PAIRSOCK_H](#)
- [PRINT_UTIL_H](#)
- [ProcessControl_h](#)
- [Socket_h](#) *DtSocket is a misnomer This provides net-level Packet IO Sometimes, it is implemented with sockets, sometimes not*

- [_VLPRINT_H](#)
- [_VLUTIL_H](#)
- [_declspec](#)
- [_detonationPdu_h](#) *This file provides protocol independence for Detonation Interactions*
- [_parseEnum_h](#)
- [antiPersonell](#)
- [dDtStateMsg_H](#)
- [define](#)
- [gFindOwnId](#)
- [gGuiseSameAsType](#)
- [hDtStateMsg_H](#)
- [looptlist](#) *Macro to loop through the list*
- [looptlist](#)
- [looptlist2](#)
- [netPdu](#)

Enums, Unions, Structs

- [DtAH1WSpecific](#)
- [DtAcknowledgeFlag](#)
- [DtActionId](#)
- [DtAegisCruiserSpecific](#)
- [DtAggregateFormation](#)
- [DtAggregateState](#) *Aggregate State PDU enums*
- [DtAntennaPatternType](#)
- [DtAppearAttrVal](#) *Create an enumeration type, but since many of the names will have the same value, define them all as consts below*
- [DtAppearAttribute](#) *if you put spaces between the "<<" and the surrounding strings, enumstr gets confused and thinks the shift is another value*
- [DtAppearAttributeBitShift](#)
- [DtArtParamType](#) *Articulated parts*
- [DtArtPartThresholdType](#)
- [DtArtPartType](#)
- [DtAttachedPartStatus](#)
- [DtAttributeChangeResult](#)
- [DtBeamFunc](#)
- [DtBeamType](#)
- [DtBoolean](#) *For convenience*
- [DtCV59Specific](#)
- [DtCamouflageType](#)

- [DtCapability](#)
- [DtCharacterSet](#)
- [DtCollisionType](#)
- [DtCountryCode](#)
- [DtCreateObjectResult](#)
- [DtCryptoMode](#)
- [DtCulturalFeatureDomain](#)
- [DtCulturalFeatureLand](#)
- [DtDamageState](#) *This is for the tank ditch and concertina wire breach appearance Areal object specific appearance*
- [DtDatumParam](#)
- [DtDatumRecordType](#)
- [DtDatumRetCode](#)
- [DtDeadReckonTypes](#) *Section 7*
- [DtDensityType](#)
- [DtDesignatorCode](#)
- [DtDesignatorCodeName](#)
- [DtDetailedModulationAmplitudeAngle](#)
- [DtDetailedModulationAngle](#)
- [DtDetailedModulationCombination](#)
- [DtDetailedModulationType](#)
- [DtDetailedModulationUnmodulated](#)
- [DtDetonationResult](#)
- [DtDetonatorFuze](#)
- [DtEmissionFunc](#)
- [DtEmissionPduType](#)
- [DtEmitterSysName](#)
- [DtEmitterSysRetCode](#) *error codes for looking up individual beams*
- [DtEncodingClass](#)
- [DtEncodingType](#)
- [DtEntityKind](#)
- [DtEnvironmentalAir](#)
- [DtEnvironmentalDomain](#)
- [DtEnvironmentalModelType](#) *Environmental Process PDU enums*
- [DtEnvironmentalStatusBit](#)
- [DtEventType](#)
- [DtF16](#)
- [DtFFG7Specific](#)
- [DtFLIR_SensorTypeSubcategory](#)

- [DtFedReadStatus](#)
- [DtFilterType](#)
- [DtForceType](#) *Section 4*
- [DtGridAxisType](#)
- [DtGridDataRecType](#) *Gridded data PDU enums*
- [DtGroupedEntityCategory](#) *IsGroupOf PDU enums*
- [DtHatchState](#)
- [DtIffAtcNavAidsAltParm4](#)
- [DtIffAtcNavAidsApplicableModes](#)
- [DtIffAtcNavAidsChangeOptionsBit](#)
- [DtIffAtcNavAidsLayerSpecificInfo](#)
- [DtIffAtcNavAidsModifier](#)
- [DtIffAtcNavAidsParameterBit](#)
- [DtIffAtcNavAidsParm1](#)
- [DtIffAtcNavAidsParm2](#)
- [DtIffAtcNavAidsStatusBit](#)
- [DtIffAtcNavAidsSysMode](#)
- [DtIffAtcNavAidsSysName](#) *IFF/ATC/NAVAIDS PDU enums*
- [DtIffAtcNavAidsSysType](#)
- [DtInitErrorAction](#)
- [DtIntercomCmdType](#)
- [DtIntercomCommClass](#)
- [DtIntercomCommType](#)
- [DtIntercomCtlRecType](#)
- [DtIntercomCtlType](#) *Intercom Control PDU enums*
- [DtIntercomDestLineStateCmd](#)
- [DtIntercomTransLineState](#)
- [DtLE_AppearanceField1Flag](#)
- [DtLE_AppearanceField2Flag](#)
- [DtLE_DetonField2Flag](#)
- [DtLE_DetonFieldFlag](#)
- [DtLE_FireFieldFlag](#)
- [DtLE_TSPI_FieldFlag](#)
- [DtLHASpecific](#)
- [DtLaserSensorTypeSubcategory](#)
- [DtLgrCommand](#)
- [DtLgrDirection](#) *Play forwards or backwards when in LGR_PLAY mode*
- [DtLgrFilePosition](#) *Location status*

- [DtLgrMode](#) *General operation modes*
- [DtLgrProcess](#) *Receiving application may filter lcPdu based on this start-up flag*
- [DtLgrStyle](#)
- [DtLifeFormDomain](#)
- [DtLifeFormLand](#)
- [DtLifeformState](#)
- [DtLightState](#)
- [DtLongWordPortion](#)
- [DtMagneticSensorTypeSubcategory](#)
- [DtMinefieldAlgae](#)
- [DtMinefieldAntiHandlingDevice](#)
- [DtMinefieldDataFilterField](#) *Minefield PDU enums*
- [DtMinefieldFusing](#)
- [DtMinefieldPaintScheme](#)
- [DtMinefieldProtocol](#)
- [DtModificationField](#)
- [DtMultispectralSensorTypeSubcategory](#)
- [DtMunitionAntiAir](#)
- [DtMunitionAntiArmor](#)
- [DtMunitionAntiPersonnel](#)
- [DtMunitionAntiRadar](#)
- [DtMunitionAntiShip](#)
- [DtMunitionAntiSubmarine](#)
- [DtMunitionBattlefieldSupport](#)
- [DtMunitionDomain](#)
- [DtMunitionUKAntiPersonnelUnguided](#) *UK*
- [DtMunitionUKBattlefieldSupportUnguided](#)
- [DtMunitionUSAntiAirGuided](#) *US*
- [DtMunitionUSAntiArmorGuided](#)
- [DtMunitionUSAntiArmorUnguided](#)
- [DtMunitionUSAntiPersonnelUnguided](#)
- [DtMunitionUSAntiRadarGuided](#)
- [DtMunitionUSAntiSubmarineGuided](#)
- [DtMunitionUSBattlefieldSupportGuided](#)
- [DtMunitionUSBattlefieldSupportUnguided](#)
- [DtMunitionUSSRAntiAirGuided](#) *USSR*
- [DtMunitionUSSRAntiPersonnelUnguided](#)
- [DtNature](#) *IsPartOf PDU enums*

- [DtNumberOfIndividualLifeForms](#)
- [DtObjectKind](#) *Environmental process PDU enums*
- [DtOpticalSensorTypeSubcategory](#)
- [DtOrderType](#)
- [DtPduKind](#)
- [DtPhysicalSensorTypeSubcategory](#)
- [DtPlatformAir](#)
- [DtPlatformAirFranceUtilityHelicopter](#) *France*
- [DtPlatformAirUSAttack](#)
- [DtPlatformAirUSAttackHelicopter](#)
- [DtPlatformAirUSCargo](#)
- [DtPlatformAirUSFighter](#)
- [DtPlatformAirUSObservationHelicopter](#)
- [DtPlatformAirUSSRAttack](#)
- [DtPlatformAirUSSRAttackHelicopter](#)
- [DtPlatformAirUSSRBomber](#)
- [DtPlatformAirUSSRFighter](#)
- [DtPlatformAirUSSSpecialOperations](#)
- [DtPlatformAirUSUnmannedAirVehicle](#)
- [DtPlatformDomain](#)
- [DtPlatformLand](#)
- [DtPlatformLandAnySmallWheeledUtilityVehicle](#) *Any Country*
- [DtPlatformLandUKTank](#) *UK*
- [DtPlatformLandUSArmoredFightingVehicle](#)
- [DtPlatformLandUSArmoredUtilityVehicle](#)
- [DtPlatformLandUSSRArmoredFightingVehicle](#)
- [DtPlatformLandUSSRLargeWheeledUtilityVehicle](#)
- [DtPlatformLandUSSRSelfPropelledArtillery](#)
- [DtPlatformLandUSSRSelfTowedArtillery](#)
- [DtPlatformLandUSSRTank](#) *USSR*
- [DtPlatformLandUSSSelfPropelledArtillery](#)
- [DtPlatformLandUSSSmallWheeledUtilityVehicle](#)
- [DtPlatformLandUSTank](#) *US*
- [DtPlatformSubsurfaceUSStrategicMissileSubmarine](#)
- [DtPlatformSurface](#)
- [DtPlatformSurfaceUSAmphibiousAssaultShip](#)
- [DtPlatformSurfaceUSCarrier](#)
- [DtPlatformSurfaceUSGuidedMissileCruiser](#)

- [DtPlatformSurfaceUSGuidedMissileFrigate](#)
- [DtPolarDirection](#)
- [DtPosition](#)
- [DtProtocolFamily](#)
- [DtRADAR_SensorTypeSubcategory](#)
- [DtRadioCryptoSystem](#)
- [DtRadioDetailedModulationAmplitude](#)
- [DtRadioInputSource](#)
- [DtRadioMajorModulation](#)
- [DtRadioReceiverState](#)
- [DtRadioSpreadSpectrum](#)
- [DtRadioSystem](#)
- [DtRadioTransmitState](#)
- [DtReferenceSys](#)
- [DtRemoveObjectResult](#)
- [DtRepairResult](#)
- [DtRepairType](#) *Section 2*
- [DtRequestStatus](#)
- [DtRequiredReliabilityService](#) *Reliable service PDU enums*
- [DtResponseFlag](#)
- [DtRestStatus](#)
- [DtSA10Specific](#)
- [DtSA8Specific](#)
- [DtSONARSensorTypeSubcategory](#)
- [DtSensorTypeCategory](#)
- [DtServiceType](#)
- [DtSizeQualifier](#)
- [DtStationName](#)
- [DtStopReason](#)
- [DtSubProtocol](#) *Live entity protocol PDU enums*
- [DtSupplyCategories](#)
- [DtT72Specific](#)
- [DtTacticalDataLinkType](#)
- [DtTestOperator](#)
- [DtTimeStampType](#)
- [DtTrackingJamming](#)
- [DtTrailState](#)
- [DtTransferType](#) *Transfer Control Request PDU enums*

- [DtTransportType](#)
- [DtUnderwaterAcousticActiveEmissionParmIndex](#)
- [DtUnderwaterAcousticAdditionalPassiveActivityParmIndex](#)
- [DtUnderwaterAcousticFunction](#)
- [DtUnderwaterAcousticHullMountedMaskerStatus](#)
- [DtUnderwaterAcousticPassiveParmIndex](#)
- [DtUnderwaterAcousticPropulsionPlantConfig](#)
- [DtUnderwaterAcousticScanPattern](#)
- [DtUnderwaterAcousticSystemName](#)
- [DtUnderwaterAcousticUpdate](#) *Underwater acoustic PDU enums*
- [DtUserProtocol](#)
- [DtVectorOffset](#)
- [DtViewerCommand](#)
- [DtWarheadType](#)
- [DtWeaponState](#)
- [generic_mm_multflag](#) *Generic DcmDcm Multiply routine*

Typedefs

- [DtBufferPtr](#)
- [DtEntityKind](#)
- [DtMunition](#)
- [DtQ0](#) *Quaternion routines*
- [DtSimulator](#)
- [DtSite](#) *for backwards compatibility*
- [DtSite](#)
- [DtX](#)
- [East](#)
- [artOrAttached](#)
- [deg](#)
- [hour](#)
- [kind](#)
- [myArtOrAttached](#)
- [myValue](#) *Used only for internal representation of Art Param information*
- [partType](#)
- [psi](#)
- [semiMajor](#) *Spheroid*
- [spheroid](#)
- [t](#)

- [WX](#)

[Hierarchy of classes](#)



VR-Link Class Documentation

MÄK Technologies: www.mak.com

VR-Link 3.4 DIS Class Documentation

[Alphabetic index](#)

Hierarchy of Classes

- [DtAbsNotifyObserver](#)
- [DtAckPredOpt](#)
- [DtAggregatePublisher](#)
- [DtAggregateStateDecoder](#)
- [DtAggregateStateEncoder](#)
- [DtAntennaPatternRecord](#)
 - [DtBeamAntennaPatternRecord](#)
- [DtApproximator](#)
 - [DtDeadReckoner](#)
 - [DtSmoother](#)
- [DtArtPart](#)
- [DtArtPartList](#)
- [DtArtPartThresholder](#)
- [DtAttachedPartList](#)
- [DtBaseExerciseConn](#)
 - [DtExerciseConn](#)
- [DtBaseHashKey](#)
 - [DtSystemHashKey](#)
 - [DtStringHashKey](#)
 - [DtObjectIdHashKey](#)
 - [DtIntHashKey](#)
- [DtBaseMsg](#)
 - [DtPdu](#)
 - [DtUnknownPdu](#)
 - [DtUnderwaterAcousticPdu](#)
 - [DtTransmitterPdu](#)
 - [DtSuppEmisEntStatePdu](#)
 - [DtStateMsg](#)
 - [DtEntityStatePdu](#)
 - [DtEmissionPdu](#)
 - [DtElectromagneticEmissionPdu](#)
 - [DtAggregateStatePdu](#)

- [DtServiceRequestPdu](#)
- [DtResupplyReceivedPdu](#)
- [DtResupplyOfferPdu](#)
- [DtResupplyCancelPdu](#)
- [DtRepairResponsePdu](#)
- [DtRepairCompletePdu](#)
- [DtRecordSetPdu](#)
 - [DtTransferControlRequestPdu](#)
 - [DtSetRecordPdu](#)
 - [DtRecordPdu](#)
- [DtRecordQueryPdu](#)
- [DtReceiverPdu](#)
- [DtPointObjectStatePdu](#)
- [DtMinefieldStatePdu](#)
- [DtMinefieldResponseNackPdu](#)
- [DtMinefieldQueryPdu](#)
- [DtMinefieldDataPdu](#)
- [DtLinearObjectStatePdu](#)
- [DtLeTspiPdu](#)
- [DtLeFirePdu](#)
- [DtLeDetonationPdu](#)
- [DtLeArticulatedPartsPdu](#)
- [DtLeAppearancePdu](#)
- [DtLaserPdu](#)
- [DtIsPartOfPdu](#)
- [DtIsGroupOfPdu](#)
- [DtIntercomSignalPdu](#)
- [DtIntercomControlPdu](#)
- [DtInteraction](#)
 - [DtViewControlPdu](#)
 - [DtUnknownVCPdu](#)
 - [DtTrackVCPdu](#)
 - [DtTetherVCPdu](#)
 - [DtTetherTrackVCPdu](#)
 - [DtOrbitVCPdu](#)
 - [DtModelessVCPdu](#)
 - [DtMimicVCPdu](#)
 - [DtMimicTrackVCPdu](#)

- [DtGroupVCPdu](#)
- [DtCompassVCPdu](#)
- [DtAbsoluteVCPdu](#)
- [DtSimanPdu](#)
 - [DtStopFreezeRPdu](#)
 - [DtStopFreezePdu](#)
 - [DtStartResumeRPdu](#)
 - [DtStartResumePdu](#)
 - [DtRemoveEntityRPdu](#)
 - [DtRemoveEntityPdu](#)
 - [DtPduWithData](#)
 - [DtSetDataRPdu](#)
 - [DtSetDataPdu](#)
 - [DtEventReptRPdu](#)
 - [DtEventReportPdu](#)
 - [DtDataRPdu](#)
 - [DtDataPdu](#)
 - [DtCommentRPdu](#)
 - [DtCommentPdu](#)
 - [DtActionResponseRPdu](#)
 - [DtActionResponsePdu](#)
 - [DtActionRequestRPdu](#)
 - [DtActionRequestPdu](#)
 - [DtMessagePdu](#)
 - [DtDataQueryRPdu](#)
 - [DtDataQueryPdu](#)
 - [DtCreateEntityRPdu](#)
 - [DtCreateEntityPdu](#)
 - [DtAcknowledgeRPdu](#)
 - [DtAcknowledgePdu](#)
- [DtSignalPdu](#)
- [DtLgrControlPdu](#)
 - [DtLgrTimeoutPdu](#)
 - [DtLgrTimeScalePdu](#)
 - [DtLgrStopPdu](#)
 - [DtLgrStartPdu](#)
 - [DtLgrSkipTimePdu](#)
 - [DtLgrSetTimePdu](#)

- [DtLgrRecordPdu](#)
 - [DtLgrRecordActionPdu](#)
 - [DtLgrRecFilePdu](#)
 - [DtLgrQuitPdu](#)
 - [DtLgrPlayPdu](#)
 - [DtLgrPlayActionPdu](#)
 - [DtLgrPbFilePdu](#)
 - [DtLgrPausePdu](#)
 - [DtLgrHeartbeatPdu](#)
 - [DtLgrEofActionPdu](#)
 - [DtLgrEndPdu](#)
- [DtFirePdu](#)
- [DtDetonationPdu](#)
- [DtCollisionPdu](#)
- [DtIffAtcNavAidsPdu](#)
- [DtGriddedDataPdu](#)
- [DtEnvironmentalProcessPdu](#)
- [DtEntityStateUpdatePdu](#)
- [DtDesignatorPdu](#)
- [DtCollisionElasticPdu](#)
- [DtArealObjectStatePdu](#)
- [DtDatumRecordSet](#)
 - [DtPduDatumRecordSet](#)
- [DtBasicFixedWingAircraftRecord](#)
 - [DtEnhancedFixedWingAircraftRecord](#)
- [DtBasicGroundCombatSoldierRecord](#)
 - [DtEnhancedGroundCombatSoldierRecord](#)
- [DtBasicGroundCombatVehicleRecord](#)
 - [DtEnhancedGroundCombatVehicleRecord](#)
- [DtBasicRotorWingAircraftRecord](#)
 - [DtEnhancedRotorWingAircraftRecord](#)
- [DtBurstDescriptor](#)
- [DtCartesianView](#)
 - [DtTopoView](#)
- [DtCoordTransform](#)
- [DtCoordTranslation](#)
- [DtDatumMap](#)
- [DtDatumMapEntry](#)

- [DtDmeGlobalFcns](#)
- [DtDcm](#)
- [DtDirectedRotRate](#)
- [DtEntityIdentifier](#)
- [DtEntityPublisher](#)
- [DtEntityStateDecoder](#)
- [DtEntityStateEncoder](#)
- [DtEntityType](#)
 - [DtRadioEntityType](#)
 - [DtEnvironmentTypeRecord](#)
- [DtEventId](#)
- [DtFedReader](#)
- [DtFedTarget](#)
- [DtFilterTest](#)
 - [DtCsFilterTest](#)
- [DtFundamentalParameterData](#)
- [DtFundamentalParameterData](#)
- [DtGlobalIdHashlist](#)
- [DtGlobalObjectDesignatorList](#)
- [DtGridAxisRecordFixed](#)
 - [DtGridAxisRecordVariable](#)
- [DtGridDataRecord](#)
 - [DtGridDataRecordType2](#)
 - [DtGridDataRecordType1](#)
 - [DtGridDataRecordType0](#)
- [DtGroundLogisticsVehicleRecord](#)
- [DtHashItem](#)
- [DtHashlist](#)
- [DtIffAtcNavAidsBeamData](#)
- [DtIffAtcNavAidsLayerHeader](#)
- [DtIffAtcNavAidsOperational](#)
- [DtIffAtcNavAidsParameter](#)
- [DtIffAtcNavAidsSecondOperational](#)
- [DtIffAtcNavAidsSystemID](#)
- [DtIntercomParmRec](#)
 - [DtIntercomGroupDestRec](#)
 - [DtIntercomGroupAsgmtRec](#)
 - [DtIntercomEntityDestRec](#)

- [DtLeDeadReckoningParameters](#)
- [DtLeEntityID](#)
- [DtLeEntityLinearVelocity](#)
- [DtLeEntityLocationEntityCoordinates](#)
- [DtLeEntityLocationWorldCoordinates](#)
- [DtLeEntityOrientation](#)
- [DtLeOrientationError](#)
- [DtLePositionError](#)
- [DtLinearSegmentParameter](#)
- [DtList](#)
 - [DtScript](#)
 - [DtIntrusiveList](#)
- [DtListItem](#)
- [DtListedObj](#)
 - [DtReflectedObjectCallbackInfo](#)
 - [DtReflectedObject](#)
 - [DtReflectedEntity](#)
 - [DtReflectedEmitterSystem](#)
 - [DtReflectedAggregate](#)
 - [DtBaseReflectedEntity](#)
 - [DtBaseReflectedAggregate](#)
- [DtMinefieldPerimeterRecord](#)
- [DtModulationType](#)
- [DtNet32Vector](#)
- [DtNet64Vector](#)
- [DtNetBurstDescriptor](#)
- [DtNetEntityIdentifier](#)
- [DtNetEntityMarking](#)
- [DtNetEntityType](#)
- [DtNetEulerAngles](#)
- [DtNetEventId](#)
- [DtNetFloat32](#)
- [DtNetFloat64](#)
- [DtNetInt16](#)
- [DtNetInt32](#)
- [DtNetRadioEntityType](#)
- [DtNetSimulationAddress](#)
- [DtNetTimeStamp](#)

- [DtNetU16](#)
- [DtNetU32](#)
- [DtNullFederateAmbassador](#)
- [DtObjectIdHashlist](#)
- [DtObjectType](#)
- [DtPduCallbackInfo](#)
- [DtPduCallbackManager](#)
- [DtPduFactory](#)
 - [DtDisPduFactory](#)
- [DtPropulsionSystem](#)
- [DtRecordSet](#)
- [DtReflectedObjectList](#)
 - [DtReflectedEntityList](#)
 - [DtReflectedEmitterSystemList](#)
 - [DtReflectedAggregateList](#)
- [DtSelectParam](#)
- [DtSilentAggregateList](#)
- [DtSilentEntityList](#)
- [DtSmoothOrient](#)
 - [DtSmoothOriWithDr](#)
- [DtSmoothPos](#)
 - [DtSmoothPosWithDr](#)
- [DtSocket](#)
 - [DtTcpSocket](#)
 - [DtSocketPairSocket](#)
 - [DtNetSocket](#)
 - [DtGroupSocket](#)
 - [DtClientSocket](#)
- [DtStateRepository](#)
 - [DtEmitterBeamRepository](#)
 - [DtEmbeddedSystemRepository](#)
 - [DtRadioTransmitterRepository](#)
 - [DtRadioReceiverRepository](#)
 - [DtEmitterSystemRepository](#)
 - [DtDesignatorRepository](#)
 - [DtBaseEntityStateRepository](#)
 - [DtEntityStateRepository](#)
 - [DtAggregateStateRepository](#)

- [DtString](#)
 - [DtFilename](#)
- [DtTaitBryan](#)
- [DtTcpForwarder](#)
- [DtTcpSocketMgr](#)
- [DtThresholder](#)
- [DtTrackJamDesignator](#)
- [DtTrackJamDesignatorList](#)
- [DtTrackJamTarget](#)
- [DtTrackJamTarget](#)
- [DtUnderwaterAcousticAddtnlPassiveAct](#)
- [DtUnderwaterAcousticBeam](#)
- [DtUnderwaterAcousticEmitterSystem](#)
- [DtUnderwaterAcousticShaftSpeed](#)
- [DtUtmView](#)
- [DtVarDatumSpec](#)
- [DtVariableDatumRecord](#)
- [DtVector](#)
 - [DtUtmCoord](#)
 - [DtGeodeticCoord](#)
- [DtVectoringNozzleSystem](#)

[Alphabetic index](#)

In file absnotifyobs.h:

class DtAbsNotifyObserver

class DtAbsNotifyObserver: DtAbsNotifyObserver is an abstract base class

Public Methods

- virtual **~DtAbsNotifyObserver ()**
destructor
- [DtAbsNotifyObserver](#)& **operator=** (const [DtAbsNotifyObserver](#)& orig)
assignment operator
- virtual [DtBoolean](#) **notifyFromObserved** (DtObservee* observed, const DtObservee::DtWhyNotifyObserver& why)
Called by DtObservee when something has happened to the observed object
- **DtAbsNotifyObserver ()**
default constructor

Protected Methods

- **DtAbsNotifyObserver** (const [DtAbsNotifyObserver](#)& orig)
copy constructor

Protected

- Constructors are protected since this is an abstract base class
-

Documentation

class DtAbsNotifyObserver:

DtAbsNotifyObserver is an abstract base class. Instances of DtAbsNotifyObserver are used by DtObservee to notify an observer object that something has happened to the observed object. Observers should subclass from DtAbsNotifyObserver to receive notification for their class.

- virtual **~DtAbsNotifyObserver ()**
destructor
- [DtAbsNotifyObserver](#)& **operator=** (const [DtAbsNotifyObserver](#)& orig)
assignment operator
- virtual [DtBoolean](#) **notifyFromObserved** (DtObservee* observed, const DtObservee::DtWhyNotifyObserver& why)

Called by DtObservee when something has happened to the observed object. Default does nothing. This should be overridden.

Constructors are protected since this is an abstract base class

DtAbsNotifyObserver()

default constructor

DtAbsNotifyObserver(const [DtAbsNotifyObserver](#)& orig)

copy constructor

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file machinetypes.h:

enum DtBoolean

For convenience

 **DtFalse**

 **DtTrue**

Documentation

For convenience

 **DtFalse**

 **DtTrue**

In file smutil.h:

class DtAckPredOpt

options for DtAcknowledgePredicate

Public Fields

- [DtEntityIdentifier](#) to
- [DtEntityIdentifier](#) from
- [DtAcknowledgeFlag](#) ackFlag
- int negAckRetVal
- char verboseToOther
- char verboseFromOther
- char verboseOtherType
- char verboseNegAck
- char verbosePosAck

Public Methods

- [DtAckPredOpt](#) ()
-

Documentation

options for DtAcknowledgePredicate

- [DtAckPredOpt](#) ()
- [DtEntityIdentifier](#) to
- [DtEntityIdentifier](#) from
- [DtAcknowledgeFlag](#) ackFlag
- int negAckRetVal
- char verboseToOther
- char verboseFromOther
- char verboseOtherType

char verboseNegAck

char verbosePosAck

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file hoststructs.h:

class DtEntityIdentifier

Public Fields

 DtSimulationAddress **DtSimulator**
 DtU16 **DtEntity**

Public Methods

 **DtEntityIdentifier** (const char* str)
 **DtEntityIdentifier** (int site, int host, int entityNum)
 **DtEntityIdentifier** ([DtNetEntityIdentifier](#) netId)
 **DtEntityIdentifier** ()
 void **init** (int site, int host, int entityNum)
 void **setFromNet** ([DtNetEntityIdentifier](#) netId)
 **operator DtNetEntityIdentifier** () const
 int **site** () const
 int **host** () const
 int **entityNum** () const
 void **setSite** (int site)
 void **setHost** (int host)
 void **setEntityNum** (int entityNum)
 int **sameHost** (const [DtEntityIdentifier](#) &otherGuy) const
 int **sameEntity** (const [DtEntityIdentifier](#) &otherGuy) const
 int **matchPattern** (const [DtEntityIdentifier](#) &pat) const
 int **operator==** (const [DtEntityIdentifier](#) &otherGuy) const
 int **operator!=** (const [DtEntityIdentifier](#) &otherGuy) const
 char* **string** () const
 static const [DtEntityIdentifier](#)& **nullId** ()

Documentation

```
DtEntityIdentifier(const char* str)
DtEntityIdentifier(int site, int host, int entityNum)
DtEntityIdentifier(DtNetEntityIdentifier netId)
DtEntityIdentifier()
void init(int site, int host, int entityNum)
void setFromNet(DtNetEntityIdentifier netId)
operator DtNetEntityIdentifier () const
int site() const
int host() const
int entityNum() const
void setSite(int site)
void setHost(int host)
void setEntityNum(int entityNum)
int sameHost(const DtEntityIdentifier &otherGuy) const
int sameEntity(const DtEntityIdentifier &otherGuy) const
int matchPattern(const DtEntityIdentifier &pat) const
int operator==(const DtEntityIdentifier &otherGuy) const
int operator!=(const DtEntityIdentifier &otherGuy) const
char* string() const
static const DtEntityIdentifier& nullId()
DtSimulationAddress DtSimulator
DtU16 DtEntity
```

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file netstructs.h:

class DtNetEntityIdentifier

Public Fields

 [DtNetSimulationAddress](#) DtSimulator

 DtNetU16 DtEntity

Public Methods

 [DtEntityIdentifier](#) operator = ([DtEntityIdentifier](#))

 [DtNetEntityIdentifier](#) operator = ([DtNetEntityIdentifier](#))

Documentation

 [DtNetSimulationAddress](#) DtSimulator

 DtNetU16 DtEntity

 [DtEntityIdentifier](#) operator = ([DtEntityIdentifier](#))

 [DtNetEntityIdentifier](#) operator = ([DtNetEntityIdentifier](#))

This class has no child classes.

In file netstructs.h:

class DtNetSimulationAddress

Public Fields

 DtNetU16 **DtSite**

 DtNetU16 **DtHost**

Public Methods

 DtSimulationAddress **operator =** (DtSimulationAddress)

 [DtNetSimulationAddress](#) **operator =** ([DtNetSimulationAddress](#))

Documentation

 DtNetU16 DtSite

 DtNetU16 DtHost

 DtSimulationAddress operator = (DtSimulationAddress)

 [DtNetSimulationAddress](#) operator = ([DtNetSimulationAddress](#))

This class has no child classes.

In file disenums.h:

enum DtAcknowledgeFlag

-
-  DtAckCreate
 -  DtAckRemove
 -  DtAckStart
 -  DtAckStop
-

Documentation

-  DtAckCreate
-  DtAckRemove
-  DtAckStart
-  DtAckStop

In file dagpub.h:

class DtAggregatePublisher

class DtAggregatePublisher: Instances of DtAggregatePublisher are used to take care of the details associated with publishing the state of an aggregate

Public Methods

	DtAggregatePublisher (DtExerciseConn * conn, const DtObjectId& id = gFindOwnId()) <i>constructor</i>
	DtAggregatePublisher (const DtEntityType & type, DtExerciseConn * conn, const DtObjectId& id = gFindOwnId()) <i>constructor</i>
	DtAggregatePublisher (const DtEntityType & type, DtExerciseConn * conn, DtDeadReckonTypes algorithm, DtForceType forceId, const DtObjectId& id = gFindOwnId()) <i>constructor</i>
 virtual	~DtAggregatePublisher () <i>destructor</i>
	DtAggregatePublisher (const DtAggregatePublisher & orig) <i>copy constructor</i>
 DtAggregatePublisher &	operator= (const DtAggregatePublisher & orig) <i>assignment operator</i>
 virtual void	tick () <i>Tick the publisher</i>
 virtual DtAggregateStateRepository *	aggregateStateRep () <i>Get the aggregate state repository for this aggregate publisher</i>
 virtual const DtObjectId&	id () const <i>Get the entity's identifier</i>
 virtual const DtGlobalObjectDesignator&	globalId () const <i>Get the global object designator - same as id for DIS</i>
 virtual void	setTimeThreshold (DtTime time) <i>Set/get threshold for timeouts</i>
 virtual DtTime	timeThreshold () const
 virtual DtExerciseConn *	exerciseConn () <i>Get our exercise connection</i>
 static void	setDfltTimeThreshold (DtTime time) <i>Set/Get default time threshold</i>
 static DtTime	dfltTimeThreshold ()
 static const DtObjectId&	findOwnId () <i>returns an id of (-1,-1,-1)</i>
 inline DtExerciseConn *	exerciseConn ()

Protected Fields

 <code>DtObjectId</code>	<code>myId</code>
 <code>DtExerciseConn*</code>	<code>myExConn</code>
 <code>DtAggregateStateRepository*</code>	<code>myAggregateStateRep</code>
 <code>DtAggregateStateRepository*</code>	<code>myAsSeenByRemote</code>
 <code>DtAggregateStateDecoder*</code>	<code>myAggregateDecoder</code>
 <code>DtAggregateStateEncoder*</code>	<code>myAggregateEncoder</code>
 <code>DtTime</code>	<code>mySimTimeLastUpdate</code>
 <code>DtTime</code>	<code>myTimeThreshold</code>
 <code>static DtTime</code>	<code>theDfltTimeThreshold</code>

Protected Methods

-  `void` `init` (`DtExerciseConn*` `exer`, `const` `DtEntityIdentifier&` `vehId`)
not virtual since called from constructor
-

Documentation

class `DtAggregatePublisher`: Instances of `DtAggregatePublisher` are used to take care of the details associated with publishing the state of an aggregate

-  `DtAggregatePublisher`(`DtExerciseConn*` `conn`, `const` `DtObjectId&` `id` = `gFindOwnId()`)
constructor
-  `DtAggregatePublisher`(`const` `DtEntityType&` `type`, `DtExerciseConn*` `conn`, `const` `DtObjectId&` `id` = `gFindOwnId()`)
constructor
-  `DtAggregatePublisher`(`const` `DtEntityType&` `type`, `DtExerciseConn*` `conn`, `DtDeadReckonTypes` `algorithm`, `DtForceType` `forceId`, `const` `DtObjectId&` `id` = `gFindOwnId()`)
constructor
-  `virtual` `~DtAggregatePublisher`()
destructor
-  `DtAggregatePublisher`(`const` `DtAggregatePublisher&` `orig`)
copy constructor
-  `DtAggregatePublisher&` `operator=`(`const` `DtAggregatePublisher&` `orig`)
assignment operator
-  `virtual` `void` `tick`()
Tick the publisher

- `virtual DtAggregateStateRepository* aggregateStateRep()`
Get the aggregate state repository for this aggregate publisher
- `virtual const DtObjectId& id() const`
Get the entity's identifier
- `virtual const DtGlobalObjectDesignator& globalId() const`
Get the global object designator - same as id for DIS
- `virtual void setTimeThreshold(DtTime time)`
Set/get threshold for timeouts
- `virtual DtTime timeThreshold() const`
- `virtual DtExerciseConn* exerciseConn()`
Get our exercise connection
- `static void setDfltTimeThreshold(DtTime time)`
Set/Get default time threshold
- `static DtTime dfltTimeThreshold()`
- `static const DtObjectId& findOwnId()`
returns an id of (-1,-1,-1)
- `void init(DtExerciseConn* exer, const DtEntityIdentifier& vehId)`
not virtual since called from constructor
- `DtObjectId myId`
- `DtExerciseConn* myExConn`
- `DtAggregateStateRepository* myAggregateStateRep`
- `DtAggregateStateRepository* myAsSeenByRemote`
- `DtAggregateStateDecoder* myAggregateDecoder`
- `DtAggregateStateEncoder* myAggregateEncoder`
- `DtTime mySimTimeLastUpdate`
- `DtTime myTimeThreshold`
- `static DtTime theDfltTimeThreshold`
- `inline DtExerciseConn* exerciseConn()`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file dexerciseconn.h:

class DtExerciseConn: public DtBaseExerciseConn

Inheritance:

DtExerciseConn - [DtBaseExerciseConn](#)

Public Fields

static int **nSizeMismatchComplain**

Public Methods

	DtExerciseConn (int port, int exercise_id, int site_id, int application_num) <i>Constructors</i>
	DtExerciseConn (int port, DtInetAddr destAddr, int exercise_id, int site_id, int application_num)
	DtExerciseConn (<u>DtSocket</u> *skt, int exercise_id, int site_id, int application_num)
	~DtExerciseConn () <i>Destructor</i>
	addPduCallback (<u>DtPduKind</u> pduKind, DtPduCallbackFcn fcn, void *usr) <i>Register/deregister callbacks by PDU kind</i>
	removePduCallback (<u>DtPduKind</u> pduKind, DtPduCallbackFcn fcn, void *usr)
	drainInput (DtTime timeout=-1, int nReadPerTimeCheck=100) <i>Normal way of processing input from the network</i>
	readAndProcess (void) <i>Called by drainInput, so application code rarely needs to use this</i>
	processPdu (<u>DtPdu</u> *pdu) <i>Called by readAndProcess (which is called by drainInput)</i>
	netRead (int *retCode = NULL) <i>Note: netRead() creates a new DtPdu on the heap</i>
	addInterestInPduKind (<u>DtPduKind</u> kind) <i>Adds/subtracts interest in a PDU kind</i>
	subtractInterestInPduKind (<u>DtPduKind</u> kind)
	readUntil (DtPredicate predicate, void *arg, DtTime timeout, DtTime sleepTime=0.1) <i>Repeatedly call readAndProcess until predicate holds on the PDU that is being read, or until timeout seconds have elapsed</i>
	addInterestInMcastAddr (DtInetAddr addr) <i>Add/Subtract interest in a multicast address</i>

virtual void	subtractInterestInMcastAddr (DtInetAddr addr)
virtual void	send (const DtStateMsg &msg) <i>Send a message - call down to send(DtPdu &pdu</i>
virtual void	send (const DtInteraction &msg)
virtual void	sendStamped (const DtStateMsg &msg) <i>Send a stamped message - call down to sendStamped(DtPdu &pdu</i>
virtual void	sendStamped (const DtInteraction &msg)
virtual int	send (const DtPdu &pdu, DtInetAddr addr = DtUSE_DEFAULT_IP_ADDR) <i>Send a PDU to the specified address, or the DtExerciseConn's default destination address if addr is DtUSE_DEFAULT_IP_ADDR</i>
virtual int	sendStamped (const DtPdu &pdu, DtInetAddr addr = DtUSE_DEFAULT_IP_ADDR) <i>Send a PDU to the specified address, or the DtExerciseConn's default destination address if addr is DtUSE_DEFAULT_IP_ADDR</i>
virtual DtInetAddr	destinationAddr (const DtPdu &pdu) <i>Returns the current destination address for the PDU</i>
virtual void	setExerciseId (int exercise_id) <i>Set/Get the current exercise ID</i>
virtual DtExerciseId	exerciseId ()
virtual DtSimulationAddress	applicationId () <i>Returns the current application ID (site & host)</i>
virtual const DtEntityIdentifier &	nextId () <i>Returns the next valid entity ID formed by calling makeEntityId with the current value of a entity number counter</i>
virtual DtEntityIdentifier	makeEntityId (int entityNum) <i>Returns an entity ID consisting of the DtExerciseConn's site and host numbers, and entityNum as the entity number</i>
virtual const DtEventID&	nextEventId () <i>Return an event ID consisting of the DtExerciseConn's site and host numbers, and the current value of an event number counter</i>
virtual DtSocket *	socket () <i>Returns the current DtSocket that we are using for communication</i>
virtual DtPduFactory *	pduFactory () <i>Get the PDU factory used by netRead to create the right kind of derived DtPDU</i>
virtual void	setPduFactory (DtPduFactory * fact) <i>Rather than just modifying the ExerciseConn's current PDU factory, you can tell the ExerciseConn to use a different PDU factory object</i>
virtual void	enableFiltering () <i>Enable/disable PDU filtering</i>
virtual void	disableFiltering ()
virtual void	setTimeStampType (DtTimeStampType relOrAbs) <i>Set/Get time stamp type (DtTimeStampAbsolute or DtTimeStampRelative)</i>
virtual DtTimeStampType	timeStampType () const
virtual void	setTimeStampType (int relOrAbs) <i>This function has been obsoleted by the version above that use the DtTimeStampType enumeration, but is included for backwards compatibility</i>
static const char*	vrlinkVersion () <i>Returns a string indicating the VR-Link version number</i>
inline	DtExerciseConn (int po, int ex, int si, int ap) ----- INLINES -----
inline	DtExerciseConn (DtSocket *so, int ex, int si, int ap)

inline DtSimulationAddress	applicationId ()
inline DtExerciseld	exerciseld ()
inline DtPduFactory *	pduFactory ()
inline int	readAndProcess ()
inline DtSocket *	socket ()

Public

	<i>----- READING PDUs -----</i>
	<i>----- SENDING PDUs -----</i>
	<i>----- ID MANAGEMENT -----</i>
	<i>----- MISC -----</i>

Protected Fields

DtExerciseld	exid
DtSimulationAddress	appid
int	availEventNum
int	deleteSocket
DtSocket *	mysock
DtPduCallbackManager	myPduCbMgr
int	pduInterest [DtMAX_PDU_KINDS + 1]
int	nSizeMismatch
DtBoolean	filtering
DtHashlist	mcastAddrs
DtPduFactory *	myPduFact
DtPduFactory *	myDfltPduFact
DtEntityIdentifier	myCurrentId
DtEventID	myCurrentEventId
DtTimeStampType	myRelOrAbs

Protected Methods

- **DtExerciseConn** (const [DtExerciseConn](#)& orig)
copy constructor - not really implemented
 - [DtExerciseConn](#)& **operator=** (const [DtExerciseConn](#)& orig)
assignment operator - not really implemented
 - virtual DtTime **currentTimeForStamping** ()
Return time to send with outgoing messages (default is sim time if absolute timestamping, real time otherwise)
 - void **init0** (int port, int exerId, int siteld, int applNum, int v)
Called by constructor
 - void **init0** ([DtSocket](#) *s, int exerId, int siteld, int applNum, int v)
 - void **init** ([DtSocket](#) *s, int deleteSkt, int exerId, int siteld, int applNum, int v)
 - void **zeroPdulInterestList** ()
-

Inherited from [DtBaseExerciseConn](#):

Public Methods

- virtual void **addPostDrainCallback**(DtPostDrainCb cb, void *usr)
- virtual void **removePostDrainCallback**(DtPostDrainCb cb, void *usr)

Protected Fields

- [DtList](#) myPostDrainCbs

Protected Methods

- virtual void **executePostDrainCbs**()
-

Documentation

- `DtExerciseConn(int port, int exercise_id, int site_id, int application_num)`
Constructors
- `DtExerciseConn(int port, DtInetAddr destAddr, int exercise_id, int site_id, int application_num)`
- `DtExerciseConn(DtSocket *skt, int exercise_id, int site_id, int application_num)`
- `virtual ~DtExerciseConn()`
Destructor
-

----- READING PDUs -----

virtual void addPduCallback([DtPduKind](#) pduKind, DtPduCallbackFcn fcn, void *usr)

Register/deregister callbacks by PDU kind. Called by each derived DtPdu's addCallback function, which should be called instead of these by user code.

virtual void removePduCallback([DtPduKind](#) pduKind, DtPduCallbackFcn fcn, void *usr)

virtual int drainInput(DtTime timeout=-1, int nReadPerTimeCheck=100)

Normal way of processing input from the network. Repeatedly calls readAndProcess to read PDUs and execute user callbacks. If timeout is -1, return only when there are no PDUs left to read. Otherwise, read for a maximum of timeout seconds. nReadPerTimeCheck is the number of PDUs to read in between checks to see if timeout has expired.

virtual int readAndProcess(void)

Called by drainInput, so application code rarely needs to use this. Reads and processes (via netRead and processPdu) a single PDU.

virtual int processPdu([DtPdu](#) *pdu)

Called by readAndProcess (which is called by drainInput). The default behavior is to execute any user callbacks that have been registered on the pdu's kind.

virtual [DtPdu](#)* netRead(int *retCode = NULL)

Note: netRead() creates a new DtPdu on the heap. Caller, (usually readAndProcess()), is responsible for deleting it when through.

virtual void addInterestInPduKind([DtPduKind](#) kind)

Adds/subtracts interest in a PDU kind. PDUs in which no interest has been registered are filtered out below the level of netRead (possibly by the packet server). All PDU kinds for which user callbacks have been registered automatically are "interesting", so there is rarely a need to application code to call these functions.

virtual void subtractInterestInPduKind([DtPduKind](#) kind)

virtual [DtPduKind](#) readUntil(DtPredicate predicate, void *arg, DtTime timeout, DtTime sleepTime=0.1)

Repeatedly call readAndProcess until predicate holds on the PDU that is being read, or until timeout seconds have elapsed. sleepTime indicates the amount of time to sleep in between polling the network when there are no packets to read.

virtual void addInterestInMcastAddr(DtInetAddr addr)

Add/Subtract interest in a multicast address. Valid multicast addresses are between 224.0.0.0 and 239.255.255.255.

virtual void subtractInterestInMcastAddr(DtInetAddr addr)

----- SENDING PDUs -----

virtual void send(const [DtStateMsg](#) &msg)

Send a message - call down to send(DtPdu &pdu....)

virtual void send(const [DtInteraction](#) &msg)

virtual void sendStamped(const [DtStateMsg](#) &msg)

Send a stamped message - call down to sendStamped(DtPdu &pdu....)

- `virtual void sendStamped(const DtInteraction &msg)`
- `virtual int send( const DtPdu &pdu, DtInetAddr addr = DtUSE_DEFAULT_IP_ADDR)`
Send a PDU to the specified address, or the DtExerciseConn's default destination address if addr is DtUSE_DEFAULT_IP_ADDR. PDU must be completely filled out, as it is not touched by this call.
- `virtual int sendStamped( const DtPdu &pdu, DtInetAddr addr = DtUSE_DEFAULT_IP_ADDR)`
Send a PDU to the specified address, or the DtExerciseConn's default destination address if addr is DtUSE_DEFAULT_IP_ADDR. The outgoing PDU is stamped with the current timeStamp (obtained using currentTimeForStamping()), and the current exerciseId.

- `virtual DtInetAddr destinationAddr(const DtPdu &pdu)`
Returns the current destination address for the PDU. Defaults to DtUSE_DEFAULT_IP_ADDR.

----- ID MANAGEMENT -----

- `virtual void setExerciseId(int exercise_id)`
Set/Get the current exercise ID
- `virtual DtExerciseId exerciseId()`
- `virtual DtSimulationAddress applicationId()`
Returns the current application ID (site & host)
- `virtual const DtEntityIdentifier& nextId()`
Returns the next valid entity ID formed by calling makeEntityId with the current value of a entity number counter
- `virtual DtEntityIdentifier makeEntityId(int entityNum)`
Returns an entity ID consisting of the DtExerciseConn's site and host numbers, and entityNum as the entity number
- `virtual const DtEventID& nextEventId()`
Return an event ID consisting of the DtExerciseConn's site and host numbers, and the current value of an event number counter

----- MISC -----

- `virtual DtSocket* socket()`
Returns the current DtSocket that we are using for communication
- `virtual DtPduFactory* pduFactory()`
Get the PDU factory used by netRead to create the right kind of derived DtPDU. You can use DtPduFactory member function, including its assignment operator to configure it to "know" about various PDU classes. For example, you can use `*(exConn.pduFactory()) = DtDisPduFactory();` to configure the PDU factory to know about all DIS PDUs and the VR-Link classes that represent them.
- `virtual void setPduFactory(DtPduFactory* fact)`
Rather than just modifying the ExerciseConn's current PDU factory, you can tell the ExerciseConn to use a different PDU factory object. The object passed belongs to the ExerciseConn, and should not be deleted until after a subsequent call to setPduFactory, or the ExerciseConn is deleted.
- `virtual void enableFiltering()`
Enable/disable PDU filtering

- `virtual void disableFiltering()`
- `virtual void setTimeStampType(DtTimeStampType relOrAbs)`
Set/Get time stamp type (DtTimeStampAbsolute or DtTimeStampRelative)
- `virtual DtTimeStampType timeStampType() const`
- `virtual void setTimeStampType(int relOrAbs)`
This function has been obsoleted by the version above that use the DtTimeStampType enumeration, but is included for backwards compatibility
- `static const char* vrlinkVersion()`
Returns a string indicating the VR-Link version number
- `static int nSizeMismatchComplain`
- `DtExerciseConn(const DtExerciseConn& orig)`
copy constructor - not really implemented
- `DtExerciseConn& operator=(const DtExerciseConn& orig)`
assignment operator - not really implemented
- `virtual DtTime currentTimeForStamping()`
Return time to send with outgoing messages (default is sim time if absolute timestamping, real time otherwise)
- `void init0(int port, int exerId, int siteId, int applNum, int v)`
Called by constructor
- `void init0(DtSocket *s, int exerId, int siteId, int applNum, int v)`
- `void init(DtSocket *s, int deleteSkt, int exerId, int siteId, int applNum, int v)`
- `DtExerciseId exid`
- `DtSimulationAddress appid`
- `int availEventNum`
- `int deleteSocket`
- `DtSocket* mysock`
- `DtPduCallbackManager myPduCbMgr`
- `int pduInterest[DtMAX_PDU_KINDS + 1]`
- `void zeroPduInterestList()`
- `int nSizeMismatch`
- `DtBoolean filtering`
- `DtHashlist mcastAddrs`
- `DtPduFactory* myPduFact`
- `DtPduFactory* myDfltPduFact`

• [DtEntityIdentifier](#) myCurrentId

• DtEventID myCurrentEventId

• [DtTimeStampType](#) myRelOrAbs

• inline DtExerciseConn(int po, int ex, int si, int ap)
----- INLINES -----

• inline DtExerciseConn([DtSocket](#) *so, int ex, int si, int ap)

• inline DtSimulationAddress applicationId()

• inline DtExerciseId exerciseId()

• inline [DtPduFactory](#)* pduFactory()

• inline int readAndProcess()

• inline [DtSocket](#)* socket()

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file baseexconn.h:

class DtBaseExerciseConn

Inheritance:

DtBaseExerciseConn

Public Methods

virtual	~DtBaseExerciseConn () <i>destructor</i>
DtBaseExerciseConn &	operator= (const DtBaseExerciseConn & orig) <i>assignment operator</i>
virtual void	send (const DtStateMsg & msg) <i>Send a state message to the network</i>
virtual void	send (const DtInteraction & msg) <i>Send an interaction to the network</i>
virtual void	sendStamped (const DtStateMsg & msg) <i>Send a state message to the network which has been time stamped with the current time</i>
virtual void	sendStamped (const DtInteraction & msg) <i>Send an interaction to the network which has been time stamped with the current time</i>
virtual int	drainInput (DtTime timeout=-1, int nReadPerTimeCheck=100) <i>Read and process messages from the network until either no more messages, or for timeout seconds, whichever happens first</i>
virtual void	addPostDrainCallback (DtPostDrainCb cb, void *usr) <i>add/remove callbacks to be executed from within drainInput right before it returns</i>
virtual void	removePostDrainCallback (DtPostDrainCb cb, void *usr)
	DtBaseExerciseConn () <i>default constructor</i>

Protected Fields

[DtList](#) myPostDrainCbs

Protected Methods

[DtBaseExerciseConn](#) (const [DtBaseExerciseConn](#)& orig)
copy constructor

- virtual void **executePostDrainCbs** ()
should be called by derived class' drainInputs

Protected

- Constructors are protected since this is an abstract base class
-

Documentation

- virtual ~DtBaseExerciseConn()
destructor
- [DtBaseExerciseConn](#)& operator=(const [DtBaseExerciseConn](#)& orig)
assignment operator
- virtual void send(const [DtStateMsg](#)& msg)
Send a state message to the network
- virtual void send(const [DtInteraction](#)& msg)
Send an interaction to the network
- virtual void sendStamped(const [DtStateMsg](#)& msg)
Send a state message to the network which has been time stamped with the current time
- virtual void sendStamped(const [DtInteraction](#)& msg)
Send an interaction to the network which has been time stamped with the current time
- virtual int drainInput(DtTime timeout=-1, int nReadPerTimeCheck=100)
Read and process messages from the network until either no more messages, or for timeout seconds, whichever happens first. Will read nReadPerTimeCheck messages at a time before checking for timeout. Returns number of incoming events which have been processed.
- virtual void addPostDrainCallback(DtPostDrainCb cb, void *usr)
add/remove callbacks to be executed from within drainInput right before it returns
- virtual void removePostDrainCallback(DtPostDrainCb cb, void *usr)
- Constructors are protected since this is an abstract base class
- DtBaseExerciseConn()
default constructor
- DtBaseExerciseConn(const [DtBaseExerciseConn](#)& orig)
copy constructor
- virtual void executePostDrainCbs()
should be called by derived class' drainInputs
- [DtList](#) myPostDrainCbs

Direct child classes:

[DtExerciseConn](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file dstatemsg.h:

class DtStateMsg: public DtPdu

class DtStateMsg: Instances of DtStateMsg are used to represent state data communicated via the network in a DIS (as opposed to HLA) environment

Inheritance:

DtStateMsg - DtPdu - DtBaseMsg

Public Methods

- **DtStateMsg** ()
default constructor
 - virtual **~DtStateMsg** ()
destructor
 - **DtStateMsg** (const DtStateMsg& orig)
copy constructor
 - DtStateMsg& **operator=** (const DtStateMsg& orig)
assignment operator
-

Inherited from DtPdu:

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual DtPduKind **kind**() const
- virtual DtProtocolFamily **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, DtTimeStampType type = DtTimeStampRelative)

- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- virtual void **printData**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtPduKind](#) **internalGetPduKind**() const
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)

- virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtStateMsg: Instances of DtStateMsg are used to represent state data communicated via the network in a DIS (as opposed to HLA) environment

- DtStateMsg ()**
default constructor

- `virtual ~DtStateMsg()`
destructor
 - `DtStateMsg(const DtStateMsg& orig)`
copy constructor
 - `DtStateMsg& operator=(const DtStateMsg& orig)`
assignment operator
-

Direct child classes:

[DtEntityStatePdu](#)
[DtEmissionPdu](#)
[DtElectromagneticEmissionPdu](#)
[DtAggregateStatePdu](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file pdu.h:

class DtPdu: public DtBaseMsg

Inheritance:

DtPdu - DtBaseMsg

Public Methods

 virtual	~DtPdu () <i>Destructor</i>
	DtPdu (const DtPdu &pdu, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Copy constructor</i>
 DtPdu &	operator= (const DtPdu &pdu) <i>Assignment operator</i>
 virtual void*	packet () const <i>Older function that returns a pointer to the network representation</i>
 virtual int	status () const <i>Get PDU status</i>
 virtual void	setVersion (int vers) <i>Set/Get protocol version from PDU header</i>
 virtual int	protocolVersion () const
 virtual void	setExerciseld (int id) <i>Set/Get exerciseID in PDU header</i>
 virtual int	exerciseld () const
 virtual DtPduKind	kind () const <i>Returns PDU kind from PDU header</i>
 virtual DtProtocolFamily	protocolFamily () const <i>Returns protocol family from PDU header</i>
 virtual void	setTimeStamp (DtTime time, DtTimeStampType type = DtTimeStampRelative) <i>Set the PDU header's time stamp to reflect time and type</i>
 virtual DtTime	timeStamp () const <i>Return the time (in seconds past an hour) encoded in the PDU header's time stamp</i>
 virtual DtTimeStampType	timeStampType () const <i>Return the type of the time stamp, as indicated in PDU header</i>
 virtual DtTime	guessTimeValid (DtTime referenceTime) const <i>Since timestamp only represents number of seconds past the hour, guessTimeValid chooses the hour that would make the timestamp closest to reference time, and returns a time that is timestamp seconds past that hour</i>
 virtual void	print () const <i>Print the contents of the PDU by calling printHeader and printData</i>

virtual void	printHeader () const <i>Print the PDU's header</i>
virtual void	printData () const <i>Print the PDU's data (not including header)</i>
inline int	exerciseld () const
inline DtPduKind	kind () const
inline DtNetPduHeader*	netHeader () const
inline void*	packet () const
inline DtProtocolFamily	protocolFamily () const
inline int	protocolVersion () const
inline void	setExerciseld (int id)
inline void	setStatus (int okStatus)
inline void	setTimeStamp (DtTime time, DtTimeStampType relOrAbs)
inline void	setVersion (int vers)
inline int	status () const
inline DtTime	timeStamp () const
inline DtTimeStampType	timeStampType () const

Protected Fields

int myOKStatus

Protected Methods

	DtPdu () <i>Constructor</i>
virtual void	setStatus (int okStatus) <i>Sets the PDU's status to okStatus</i>
virtual DtNetPduHeader*	netHeader () const <i>Returns a pointer to the PDU's header</i>
virtual void	initPdu (int size, DtBufferPtr buffer) <i>Must be called after default ctor before PDU is completely constructed</i>
virtual void	initPdu (const void* initial, DtBufferPtr buffer) <i>Called by from-net-rep ctor</i>
virtual DtPduKind	internalGetPduKind () const <i>Returns the PDU kind associated with the derived PDU class</i>
virtual DtProtocolFamily	kindToFamily (DtPduKind kind) const <i>Maps PDU kind to ProtocolFamily</i>
virtual void	sizeChanged (int size) <i>Called by changeSize</i>
virtual void	clearData () <i>zeros out all bytes after the PDU header</i>
virtual void	initHeader (DtPduKind kind, int size) <i>Initializes PDU header</i>

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- virtual ~DtPdu()
Destructor
- DtPdu(const [DtPdu](#) &pdu, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Copy constructor
- [DtPdu](#)& operator=(const [DtPdu](#) &pdu)

Assignment operator

• `virtual void* packet() const`

Older function that returns a pointer to the network representation. The newer `DtBaseMsg::netRep` should generally be used instead.

• `virtual int status() const`

Get PDU status. `DtSTATUS_OK` means it's good. Other values indicate a bad PDU. Return codes are derived PDU-dependent.

• `virtual void setVersion(int vers)`

Set/Get protocol version from PDU header

• `virtual int protocolVersion() const`

• `virtual void setExerciseId(int id)`

Set/Get exerciseID in PDU header

• `virtual int exerciseId() const`

• `virtual DtPduKind kind() const`

Returns PDU kind from PDU header

• `virtual DtProtocolFamily protocolFamily() const`

Returns protocol family from PDU header

• `virtual void setTimeStamp(DtTime time, DtTimeStampType type = DtTimeStampRelative)`

Set the PDU header's time stamp to reflect time and type. Time is in seconds, but only the time modulo an hour actually gets into the time stamp.

• `virtual DtTime timeStamp() const`

Return the time (in seconds past an hour) encoded in the PDU header's time stamp

• `virtual DtTimeStampType timeStampType() const`

Return the type of the time stamp, as indicated in PDU header

• `virtual DtTime guessTimeValid(DtTime referenceTime) const`

Since timestamp only represents number of seconds past the hour, `guessTimeValid` chooses the hour that would make the timestamp closest to reference time, and returns a time that is timestamp seconds past that hour

• `virtual void print() const`

Print the contents of the PDU by calling `printHeader` and `printData`

• `virtual void printHeader() const`

Print the PDU's header

• `virtual void printData() const`

Print the PDU's data (not including header). Base `DtPdu` version just prints hex.

• `DtPdu()`

Constructor. PDU is not fully constructed until `initPdu` is called.

• `virtual void setStatus(int okStatus)`

Sets the PDU's status to `okStatus`

• `virtual DtNetPduHeader* netHeader() const`

Returns a pointer to the PDU's header

virtual void initPdu(int size, DtBufferPtr buffer)

Must be called after default ctor before PDU is completely constructed. Fills in PDU header, and initializes DtBaseMsg.

virtual void initPdu(const void* initial, DtBufferPtr buffer)

Called by from-net-rep ctor. Initializes DtBaseMsg.

virtual [DtPduKind](#) internalGetPduKind() const

Returns the PDU kind associated with the derived PDU class

virtual [DtProtocolFamily](#) kindToFamily([DtPduKind](#) kind) const

Maps PDU kind to ProtocolFamily

virtual void sizeChanged(int size)

Called by changeSize. Sets the size in the header to size.

virtual void clearData()

zeros out all bytes after the PDU header

virtual void initHeader([DtPduKind](#) kind, int size)

Initializes PDU header. Called by initPdu.

int myOKStatus

inline int exerciseId() const

inline [DtPduKind](#) kind() const

inline DtNetPduHeader* netHeader() const

inline void* packet() const

inline [DtProtocolFamily](#) protocolFamily() const

inline int protocolVersion() const

inline void setExerciseId(int id)

inline void setStatus(int okStatus)

inline void setTimeStamp(DtTime time, [DtTimeStampType](#) relOrAbs)

inline void setVersion(int vers)

inline int status() const

inline DtTime timeStamp() const

inline [DtTimeStampType](#) timeStampType() const

Direct child classes:

[DtUnknownPdu](#)

[DtUnderwaterAcousticPdu](#)

[DtTransmitterPdu](#)

[DtSuppEmisEntStatePdu](#)

[DtStateMsg](#)
[DtServiceRequestPdu](#)
[DtResupplyReceivedPdu](#)
[DtResupplyOfferPdu](#)
[DtResupplyCancelPdu](#)
[DtRepairResponsePdu](#)
[DtRepairCompletePdu](#)
[DtRecordSetPdu](#)
[DtRecordQueryPdu](#)
[DtReceiverPdu](#)
[DtPointObjectStatePdu](#)
[DtMinefieldStatePdu](#)
[DtMinefieldResponseNackPdu](#)
[DtMinefieldQueryPdu](#)
[DtMinefieldDataPdu](#)
[DtLinearObjectStatePdu](#)
[DtLeTspiPdu](#)
[DtLeFirePdu](#)
[DtLeDetonationPdu](#)
[DtLeArticulatedPartsPdu](#)
[DtLeAppearancePdu](#)
[DtLaserPdu](#)
[DtIsPartOfPdu](#)
[DtIsGroupOfPdu](#)
[DtIntercomSignalPdu](#)
[DtIntercomControlPdu](#)
[DtInteraction](#)
[DtIffAtcNav aidsPdu](#)
[DtGriddedDataPdu](#)
[DtEnvironmentalProcessPdu](#)
[DtEntityStateUpdatePdu](#)
[DtDesignatorPdu](#)
[DtCollisionElasticPdu](#)
[DtArealObjectStatePdu](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file basemsg.h:

class DtBaseMsg

Inheritance:

DtBaseMsg

Public Methods

- virtual ~DtBaseMsg ()
Destructor
- DtBaseMsg (const [DtBaseMsg](#)& msg, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Copy constructor
- [DtBaseMsg](#)& operator= (const [DtBaseMsg](#)& msg)
Assignment
- virtual const char* netRep () const
Get network representation
- virtual int length () const
Get size of message
- inline const char* netRep () const

Protected Fields

- void* myNetMessage
- void* myInternalBuff
- int myIntBuffSize
- int myMsgSize

Protected Methods

- DtBaseMsg ()
Constructor Object is not fully constructed until initMsg is called
- void initMsg (int size, DtBufferPtr bufferPtr)
- void initMsg (const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void changeSize (int size)
Changes the size of the message

- virtual void **insertBytes** (int offset, int numBytes)
Inserts numBytes zeroed-out bytes at the indicated offset into the net rep
 - virtual void **insertBytes** (void *address, int numBytes)
Inserts numBytes zeroed-out bytes at the indicated address
 - virtual void **deleteBytes** (int offset, int numBytes)
Deletes numBytes bytes at the indicated offset in the net rep
 - virtual void **deleteBytes** (void *address, int numBytes)
Deletes numBytes bytes starting at the indicated address
 - virtual void **clearAll** ()
zeros out all bytes in the network representation
 - virtual void* **newInternalBuffer** (int size)
Sets myInternalBuff to point to a malloc'ed buffer of size bytes
 - virtual void **sizeChanged** (int size)
Called by changeSize
-

Documentation

- virtual ~DtBaseMsg()
Destructor
- DtBaseMsg(const [DtBaseMsg](#)& msg, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Copy constructor
- [DtBaseMsg](#)& operator=(const [DtBaseMsg](#)& msg)
Assignment
- virtual const char* netRep() const
Get network representation
- virtual int length() const
Get size of message
- DtBaseMsg()
Constructor Object is not fully constructed until initMsg is called
- void initMsg(int size, DtBufferPtr bufferPtr)
- void initMsg(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void changeSize(int size)
Changes the size of the message. If we are using an external buffer, assumes that the full buffer size is big enough to accommodate the new msg size. If using an internal buffer, the call may result in allocating a new internal buffer, and copying data from the old to the new buffer.
- virtual void insertBytes(int offset, int numBytes)
Inserts numBytes zeroed-out bytes at the indicated offset into the net rep. Uses changeSize to increase the size of the buffer.
- virtual void insertBytes(void *address, int numBytes)
Inserts numBytes zeroed-out bytes at the indicated address. Assumes that address falls within the net rep. Uses changeSize to increase the size of the buffer.

- **virtual void deleteBytes(int offset, int numBytes)**
Deletes numBytes bytes at the indicated offset in the net rep. Uses change size to decrease the size of the buffer.
 - **virtual void deleteBytes(void *address, int numBytes)**
Deletes numBytes bytes starting at the indicated address. Assumes that address falls within the net rep. Uses changeSize to decrease the size of the buffer.
 - **virtual void clearAll()**
zeros out all bytes in the network representation
 - **virtual void* newInternalBuffer(int size)**
Sets myInternalBuff to point to a malloc'ed buffer of size bytes. Sets myIntBuffSize to size. Returns myInternalBuff. Size must be a positive number.
 - **virtual void sizeChanged(int size)**
Called by changeSize. Derived versions can provide definitions that set the size field in network rep.
 - **void* myNetMessage**
 - **void* myInternalBuff**
 - **int myIntBuffSize**
 - **int myMsgSize**
 - **inline const char* netRep() const**
-

Direct child classes:

[DtPdu](#)

[DtDatumRecordSet](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file datumrecset.h:

class DtDatumRecordSet: public DtBaseMsg

Inheritance:

DtDatumRecordSet - DtBaseMsg

Public Methods

	DtDatumRecordSet (DtBufferPtr bufferPtr = DtUSE_INTERNAL_BUFFER) <i>Constructor</i>
	DtDatumRecordSet (const void* netRep, int size, DtBufferPtr bufferPtr = DtUSE_INTERNAL_BUFFER) <i>Constructor</i>
 virtual	~DtDatumRecordSet () <i>Destructor</i>
	DtDatumRecordSet (const DtDatumRecordSet & orig) <i>Copy ctor</i>
 DtDatumRecordSet &	operator= (const DtDatumRecordSet & orig) <i>Assignment operator</i>
 virtual int	numFixedFields () const <i>returns number of fixed fields</i>
 virtual int	numVarFields () const <i>returns number of variable length fields</i>
 virtual DtDatumParam	varDatumId (int index) const <i>Returns the datum param found in the index'th variable datum record</i>
 virtual DtDatumParam	fixedDatumId (int index) const <i>Returns the datum param found in the index'th fixed datum record</i>
 virtual DtDatumParam	datumId (DtDatumRecordType fixedOrVar, int index) const <i>Returns the datum param found in the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFixed or DtVar</i>
 virtual int	datumSizeBits (DtDatumRecordType fixedOrVar, int index) const <i>Returns the number of bits of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFixed or DtVar</i>
 virtual int	datumSizeBytes (DtDatumRecordType fixedOrVar, int index) const <i>Returns the number of bytes of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFixed or DtVar</i>
 virtual void	getIndex (DtDatumParam param, DtDatumRecordType * fixedOrVar, int* index, DtDatumRetCode * retcode = NULL) const <i>Looks through all fixed and variable records for one containing param as the parameter ID</i>

virtual DtFloat32	datumValFloat32 (DtDatumRecordType fixedOrVar, int index, DtDatumRetCode * retcode = NULL) const <i>***** INSPECTING DATUM VALUES BY INDEX ***** If you know the data type of the value contained in a particular record, use one of the following to return it</i>
virtual DtFloat64	datumValFloat64 (DtDatumRecordType fixedOrVar, int index, DtDatumRetCode * retcode = NULL) const
virtual unsigned long	datumValUnsigned32 (DtDatumRecordType fixedOrVar, int index, DtDatumRetCode * retcode = NULL) const
virtual long	datumValInt32 (DtDatumRecordType fixedOrVar, int index, DtDatumRetCode * retcode = NULL) const
virtual const char*	datumValByteArray (DtDatumRecordType fixedOrVar, int index) const
virtual DtFloat32	datumValFloat32 (DtDatumParam param, DtDatumRetCode * retcode = NULL) const <i>***** INSPECTING DATUM VALUES BY PARAM ID ***** If you know the data type of the value contained in a particular record, use one of the following to return it</i>
virtual DtFloat64	datumValFloat64 (DtDatumParam param, DtDatumRetCode * retcode = NULL) const
virtual unsigned long	datumValUnsigned32 (DtDatumParam param, DtDatumRetCode * retcode = NULL) const
virtual long	datumValInt32 (DtDatumParam param, DtDatumRetCode * retcode = NULL) const
virtual const char*	datumValByteArray (DtDatumParam param, DtDatumRetCode * retcode = NULL) const
virtual void	init (const DtDatumParam * fixedParams, const DtVarDatumSpec * specList) <i>Initialize to reflect the fixed parameters and variable parameters and sizes indicated</i>
virtual void	setNumFixedFields (int newNum) <i>Sets the number of fixed or variable fields to newNum</i>
virtual void	setNumVarFields (int newNum)
virtual void	setDatumParam (DtDatumRecordType fixedOrVar, int index, DtDatumParam param, DtDatumRetCode * retcode = NULL) <i>Set the datum ID in the index'th fixed or variable datum record to param, depending on whether fixedOrVar is DtFixed or DtVar</i>
virtual void	setVarDataBytes (int index, int length, DtDatumRetCode * retcode = NULL) <i>Allocates length bytes for the index'th variable datum record</i>
virtual void	setDatumValFloat64 (DtDatumRecordType fixedOrVar, int index, DtFloat64 val, DtDatumRetCode * retcode = NULL) <i>***** SETTING DATA VALUES BY INDEX ***** Use one of the following to set the data value in a fixed or variable datum record, depending on the data type of your data</i>
virtual void	setDatumValFloat32 (DtDatumRecordType fixedOrVar, int index, DtFloat32 val, DtDatumRetCode * retcode = NULL)
virtual void	setDatumValInt32 (DtDatumRecordType fixedOrVar, int index, long val, DtDatumRetCode * retcode = NULL)
virtual void	setDatumValUnsigned32 (DtDatumRecordType fixedOrVar, int index, unsigned long val, DtDatumRetCode * retcode = NULL)
virtual void	setDatumValByteArray (DtDatumRecordType fixedOrVar, int index, const char* val, DtDatumRetCode * retcode = NULL)
virtual void	setDatumValFloat64 (DtDatumParam param, DtFloat64 val, DtDatumRetCode * retcode = NULL) <i>***** SETTING DATA VALUES BY PARAM ID ***** Use one of the following to set the data value in a fixed or variable datum record, depending on the data type of your data</i>
virtual void	setDatumValFloat32 (DtDatumParam param, DtFloat32 val, DtDatumRetCode * retcode = NULL)
virtual void	setDatumValInt32 (DtDatumParam param, long val, DtDatumRetCode * retcode = NULL)

virtual void	setDatumValUnsigned32 (DtDatumParam param, unsigned long val, DtDatumRetCode * retcode = NULL)
virtual void	setDatumValByteArray (DtDatumParam param, const char* val, DtDatumRetCode * retcode = NULL)
virtual void	printData () const <i>prints the data</i>
virtual DtNetDataPduTail*	netDataRep () const <i>Returns a pointer to the network representation, as might appear in a data PDU</i>
virtual DtNetFixDatum*	nthFixedDataRecord (int index) const <i>Return a pointer to the index'th fixed data record in the network representation</i>
virtual DtNetVarDatum*	nthVarDataRecord (int index) const <i>Return a pointer to the index'th variable data record in the network representation</i>
virtual void	setFromNetFixed (const char* fixedFields, int numFixedFields) <i>Sets the fixed data from a network representation of a fixed datum record array</i>
virtual void	setFromNetVar (const char* varFields, int numVarFields, int totalVarFieldsLength) <i>Sets the variable data from a network representation consisting of the variable length records concatenated together</i>
virtual void	sizeChanged (int size) <i>Really, should not be pure virtual in DtBaseMsg, but since it is, we need a null definition here</i>
inline DtNetDataPduTail*	netDataRep () const

Public

***** MUTATOR FUNCTIONS *****

Protected Methods

virtual void* **datumValPtr** ([DtDatumRecordType](#) fixedOrVar, int index) const
Returns a pointer to the index'th fixed or variable record in the network representation, depending on whether fixedOrVar is DtFixed or DtVar

Inherited from [DtBaseMsg](#):

Public Methods

virtual const char* **netRep**() const
virtual int **length**() const
inline const char* **netRep**() const

Protected Fields

void* **myNetMsg**
void* **myInternalBuff**

• `int myIntBuffSize`

• `int myMsgSize`

Protected Methods

• `void initMsg(int size, DtBufferPtr bufferPtr)`

• `void initMsg(const void *initial, int size, DtBufferPtr bufferPtr)`

• `virtual void changeSize(int size)`

• `virtual void insertBytes(int offset, int numBytes)`

• `virtual void insertBytes(void *address, int numBytes)`

• `virtual void deleteBytes(int offset, int numBytes)`

• `virtual void deleteBytes(void *address, int numBytes)`

• `virtual void clearAll()`

• `virtual void* newInternalBuffer(int size)`

Documentation

• `DtDatumRecordSet(DtBufferPtr bufferPtr = DtUSE_INTERNAL_BUFFER)`

Constructor

• `DtDatumRecordSet(const void* netRep, int size, DtBufferPtr bufferPtr = DtUSE_INTERNAL_BUFFER)`

Constructor

• `virtual ~DtDatumRecordSet()`

Destructor

• `DtDatumRecordSet(const DtDatumRecordSet& orig)`

Copy ctor

• `DtDatumRecordSet& operator=(const DtDatumRecordSet& orig)`

Assignment operator

• `virtual int numFixedFields() const`

returns number of fixed fields

• `virtual int numVarFields() const`

returns number of variable length fields

• `virtual DtDatumParam varDatumId(int index) const`

Returns the datum param found in the index'th variable datum record. Index is between 1 and numRecords. Returns DtDatumParam(0) if index is out of bounds.

• `virtual DtDatumParam fixedDatumId(int index) const`

Returns the datum param found in the index'th fixed datum record. Index is between 1 and numRecords. Returns DtDatumParam(0) if index is out of bounds.

virtual [DtDatumParam](#) datumId([DtDatumRecordType](#) fixedOrVar, int index) const

Returns the datum param found in the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFixed or DtVar. Index is between 1 and numRecords. Returns DtDatumParam(0) if index is out of bounds.

virtual int datumSizeBits([DtDatumRecordType](#) fixedOrVar, int index) const

Returns the number of bits of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFixed or DtVar. If DtFixed, the answer is always 32.

virtual int datumSizeBytes([DtDatumRecordType](#) fixedOrVar, int index) const

Returns the number of bytes of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFixed or DtVar. If DtFixed, the answer is always 4.

virtual void getIndex([DtDatumParam](#) param, [DtDatumRecordType](#)* fixedOrVar, int* index, [DtDatumRetCode](#)* retcode = NULL) const

Looks through all fixed and variable records for one containing param as the parameter ID. If it finds one, fills supplied fixedOrVar pointer with DtFixed or DtVar, and index with the index to indicate which record was found, and fills retcode with DtDatumValid. If param was not found, retcode is filled with DtDatumBadId, and index and fixedOrVar are invalid.

virtual DtFloat32 datumValFloat32([DtDatumRecordType](#) fixedOrVar, int index, [DtDatumRetCode](#)* retcode = NULL) const

***** INSPECTING DATUM VALUES BY INDEX ***** If you know the data type of the value contained in a particular record, use one of the following to return it. fixedOrVar (either DtFixed or DtVar) and index (between 1 and numRecords) indicate which record you are inspecting. datumValByteArray is used when the data type is a character string, or something other than a long, unsigned long, float or double. It returns a pointer to a character string containing the exact bytes in the network representation that represents the value being inspected. The supplied retcode pointer will be filled with DtDatumValid if the value was successfully inspected, and DtDatumBadLength if either the index was invalid, or the size of the data indicated that it could not be interpreted as the chosen data type.

virtual DtFloat64 datumValFloat64([DtDatumRecordType](#) fixedOrVar, int index, [DtDatumRetCode](#)* retcode = NULL) const

virtual unsigned long datumValUnsigned32([DtDatumRecordType](#) fixedOrVar, int index, [DtDatumRetCode](#)* retcode = NULL) const

virtual long datumValInt32([DtDatumRecordType](#) fixedOrVar, int index, [DtDatumRetCode](#)* retcode = NULL) const

virtual const char* datumValByteArray([DtDatumRecordType](#) fixedOrVar, int index) const

virtual DtFloat32 datumValFloat32([DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL) const

***** INSPECTING DATUM VALUES BY PARAM ID ***** If you know the data type of the value contained in a particular record, use one of the following to return it. The particular record inspected will be the first one containing param as the datum ID. datumValByteArray is used when the data type is a character string, or something other than a long, unsigned long, float or double. It returns a pointer to a character string containing the exact bytes in the network representation that represents the value being inspected. The supplied retcode pointer will be filled with DtDatumValid if the value was successfully inspected, and DtDatumBadLength if either param was not found, or the size of the data indicated that it could not be interpreted as the chosen data type.

virtual DtFloat64 datumValFloat64([DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL) const

virtual unsigned long datumValUnsigned32([DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL) const

virtual long datumValInt32([DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL) const

virtual const char* datumValByteArray([DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL) const

***** MUTATOR FUNCTIONS *****

virtual void init(const [DtDatumParam](#)* fixedParams, const [DtVarDatumSpec](#)* specList)

Initialize to reflect the fixed parameters and variable parameters and sizes indicated. fixedParams is a NULL-terminated array of DtDatumParams. specList is an array of DtVarDatumSpecs terminated with the DtNullVarDatumSpec constant. This call can be made instead of calls to setNumVarDatumRecords, setNumFixedDatumRecord, and repeated calls to setDatumParam and setVarDataBytes.

virtual void setNumFixedFields(int newNum)

Sets the number of fixed or variable fields to newNum. Previously existing fixed fields with indices greater than newNum are deleted. If newNum exceeds the number of existing fields, trailing new fields are created with zeros as datum params, sizes (for var only) and values.

virtual void setNumVarFields(int newNum)

virtual void setDatumParam([DtDatumRecordType](#) fixedOrVar, int index, [DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL)

Set the datum ID in the index'th fixed or variable datum record to param, depending on whether fixedOrVar is DtFixed or DtVar. Supplied retcode pointer is filled with DtDatumValid if successful, or DtDatumBadIndex if the index is not valid.

virtual void setVarDataBytes(int index, int length, [DtDatumRetCode](#)* retcode = NULL)

Allocates length bytes for the index'th variable datum record. Should be called before using one of the setDatumVal functions to actually set the data. Supplied retcode pointer is set to DtDatumValid if successful, or DtDatumBadIndex if the index is not valid.

virtual void setDatumValFloat64([DtDatumRecordType](#) fixedOrVar, int index, DtFloat64 val, [DtDatumRetCode](#)* retcode = NULL)

***** SETTING DATA VALUES BY INDEX ***** Use one of the following to set the data value in a fixed or variable datum record, depending on the data type of your data. fixedOrVar (either DtFixed or DtVar) and index (between 1 and numRecords) indicate which record you are modifying. setDatumValBytesArray is used when the data type is a character string, or something other than a long, unsigned long, float or double. It takes a pointer to a character string containing the exact bytes that should be used in the network representation. The supplied retcode pointer will be filled with DtDatumValid if the value was successfully set, DtDatumBadIndex if the index is invalid, and DtDatumBadLength if the size of the data indicated that it could not be set from as the chosen data type.

virtual void setDatumValFloat32([DtDatumRecordType](#) fixedOrVar, int index, DtFloat32 val, [DtDatumRetCode](#)* retcode = NULL)

virtual void setDatumValInt32([DtDatumRecordType](#) fixedOrVar, int index, long val, [DtDatumRetCode](#)* retcode = NULL)

virtual void setDatumValUnsigned32([DtDatumRecordType](#) fixedOrVar, int index, unsigned long val, [DtDatumRetCode](#)* retcode = NULL)

virtual void setDatumValByteArray([DtDatumRecordType](#) fixedOrVar, int index, const char*

val, [DtDatumRetCode](#)* retcode = NULL)

virtual void setDatumValFloat64([DtDatumParam](#) param, DtFloat64 val, [DtDatumRetCode](#)* retcode = NULL)

***** SETTING DATA VALUES BY PARAM ID ***** Use one of the following to set the data value in a fixed or variable datum record, depending on the data type of your data. The particular record modified will be the first one containing param as its parameter ID. setDatumValByteArray is used when the data type is a character string, or something other than a long, unsigned long, float or double. It takes a pointer to a character string containing the exact bytes that should be used in the network representation. The supplied retcode pointer will be filled with DtDatumValid if the value was successfully set, DtDatumBadId if param is invalid, and DtDatumBadLength if the size of the data indicated that it could not be set from the chosen data type.

virtual void setDatumValFloat32([DtDatumParam](#) param, DtFloat32 val, [DtDatumRetCode](#)* retcode = NULL)

virtual void setDatumValInt32([DtDatumParam](#) param, long val, [DtDatumRetCode](#)* retcode = NULL)

virtual void setDatumValUnsigned32([DtDatumParam](#) param, unsigned long val, [DtDatumRetCode](#)* retcode = NULL)

virtual void setDatumValByteArray([DtDatumParam](#) param, const char* val, [DtDatumRetCode](#)* retcode = NULL)

virtual void printData() const
prints the data

virtual DtNetDataPduTail* netDataRep() const

Returns a pointer to the network representation, as might appear in a data PDU. Consists of numberOfFixedFields, numberOfVarFields, then the fixed and variable fields.

virtual DtNetFixDatum* nthFixedDataRecord(int index) const

Return a pointer to the index'th fixed data record in the network representation. Index is between 1 and numRecords.

virtual DtNetVarDatum* nthVarDataRecord(int index) const

Return a pointer to the index'th variable data record in the network representation. Index is between 1 and numRecords.

virtual void setFromNetFixed(const char* fixedFields, int numFixedFields)

Sets the fixed data from a network representation of a fixed datum record array. Also pass the number of fixed fields.

virtual void setFromNetVar(const char* varFields, int numVarFields, int totalVarFieldsLength)

Sets the variable data from a network representation consisting of the variable length records concatenated together. Also pass the number of fields, and the total length of the variable data records.

virtual void sizeChanged(int size)

Really, should not be pure virtual in DtBaseMsg, but since it is, we need a null definition here

virtual void* datumValPtr([DtDatumRecordType](#) fixedOrVar, int index) const

Returns a pointer to the index'th fixed or variable record in the network representation, depending on whether fixedOrVar is DtFixed or DtVar

inline DtNetDataPduTail* netDataRep() const

Direct child classes:

[DtPduDatumRecordSet](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtDatumParam

- DtDatumMass
- DtDatumForceld
- DtDatumKind
- DtDatumDomain
- DtDatumCountry
- DtDatumCategory
- DtDatumSubCategory
- DtDatumSpecific
- DtDatumExtra
- DtDatumAltKind
- DtDatumAltDomain
- DtDatumAltCountry
- DtDatumAltCategory
- DtDatumAltSubCategory
- DtDatumAltSpecific
- DtDatumAltExtra
- DtDatumXpos
- DtDatumYpos
- DtDatumZpos
- DtDatumXvel
- DtDatumYvel
- DtDatumZvel
- DtDatumPhi
- DtDatumTheta
- DtDatumPsi
- DtDatumDrAlgorithm

-  DtDatumXaccel
 -  DtDatumYaccel
 -  DtDatumZaccel
 -  DtDatumXangvel
 -  DtDatumYangvel
 -  DtDatumZangvel
 -  DtDatumAppearance
 -  DtDatumMarkingCS
 -  DtDatumMarking
 -  DtDatumNumArtParam
 -  DtDatumArtParamId
 -  DtDatumArtParamType
 -  DtDatumArtParamValue
 -  DtDatumAmountAmmoSupplies
 -  DtDatumAmmoQuantity
 -  DtDatumFuelQuantity
 -  DtDatumRadarSysStatus
 -  DtDatumRadioCommSysStatus
 -  DtDatumStationaryRadioTime
 -  DtDatumMovingRadioTime
 -  DtDatumBodyPartDamagedRatio
 -  DtDatumTerrainDatabase
 -  DtDatumLoggerFile
 -  DtDatumCapability
-

Documentation

-  DtDatumMass
-  DtDatumForceId
-  DtDatumKind
-  DtDatumDomain
-  DtDatumCountry

- DtDatumCategory
- DtDatumSubCategory
- DtDatumSpecific
- DtDatumExtra
- DtDatumAltKind
- DtDatumAltDomain
- DtDatumAltCountry
- DtDatumAltCategory
- DtDatumAltSubCategory
- DtDatumAltSpecific
- DtDatumAltExtra
- DtDatumXpos
- DtDatumYpos
- DtDatumZpos
- DtDatumXvel
- DtDatumYvel
- DtDatumZvel
- DtDatumPhi
- DtDatumTheta
- DtDatumPsi
- DtDatumDrAlgorithm
- DtDatumXaccel
- DtDatumYaccel
- DtDatumZaccel
- DtDatumXangvel
- DtDatumYangvel
- DtDatumZangvel
- DtDatumAppearance
- DtDatumMarkingCS
- DtDatumMarking
- DtDatumNumArtParam
- DtDatumArtParamId

- DtDatumArtParamType
- DtDatumArtParamValue
- DtDatumAmountAmmoSupplies
- DtDatumAmmoQuantity
- DtDatumFuelQuantity
- DtDatumRadarSysStatus
- DtDatumRadioCommSysStatus
- DtDatumStationaryRadioTime
- DtDatumMovingRadioTime
- DtDatumBodyPartDamagedRatio
- DtDatumTerrainDatabase
- DtDatumLoggerFile
- DtDatumCapability

[Alphabetic index](#) [Hierarchy of classes](#)

In file datumrecset.h:

enum DtDatumRecordType

 DtFixed

 DtVar

Documentation

 DtFixed

 DtVar

[Alphabetic index](#) [Hierarchy of classes](#)

In file datumrecset.h:

enum DtDatumRetCode

-
-  DtDatumValid
 -  DtDatumBadLength
 -  DtDatumBadId
 -  DtDatumBadIndex
-

Documentation

-  DtDatumValid
-  DtDatumBadLength
-  DtDatumBadId
-  DtDatumBadIndex

In file datumrecset.h:

class DtVarDatumSpec

This class has public data members for backwards compatibility with pre-VR-Link3

Public Fields

 [DtDatumParam](#) param

 int length

Documentation

This class has public data members for backwards compatibility with pre-VR-Link3.3 versions.

 [DtDatumParam](#) param

 int length

This class has no child classes.

In file pdudatrecset.h:

class DtPduDatumRecordSet: public DtDatumRecordSet

An instance of this class is a DtDatumRecordSet that uses a part of a DtPduWithData as its network representation

Inheritance:

DtPduDatumRecordSet - [DtDatumRecordSet](#) - [DtBaseMsg](#)

Public Methods

- **DtPduDatumRecordSet** ([DtPduWithData](#)* pdu)
Constructor
- virtual **~DtPduDatumRecordSet** ()
Destructor
- **DtPduDatumRecordSet** (const [DtPduDatumRecordSet](#)& orig)
Copy ctor
- [DtPduDatumRecordSet](#)& **operator=** (const [DtPduDatumRecordSet](#)& orig)
Assignment operator

Protected Fields

- [DtPduWithData](#)* myPdu

Protected Methods

- virtual void **changeSize** (int size)
Changes the size of the datumRecSet, by asking my PDU to change size, then reinitializing to use the right piece of the PDU's buffer
- virtual void **insertBytes** (int offset, int numBytes)
Inserts numBytes zeroed-out bytes at the indicated offset into the net rep, by asking my PDU to insert bytes, then reinitializing to use the right piece of the PDU's buffer
- virtual void **insertBytes** (void* addr, int numBytes)
Inserts numBytes zeroed-out bytes at the indicated addr within the net rep, by asking my PDU to insert bytes, then reinitializing to use the right piece of the PDU's buffer
- virtual void **deleteBytes** (int offset, int numBytes)
Deletes numBytes bytes at the indicated offset in the net rep, by asking my PDU to insert bytes, then reinitializing to use the right piece of the PDU's buffer

- virtual void **deleteBytes** (void* addr, int numBytes)
Deletes numBytes bytes at the indicated addr within the net rep, by asking my PDU to insert bytes, then reinitializing to use the right piece of the PDU's buffer
-

Inherited from [DtDatumRecordSet](#):

Public Methods

- virtual int **numFixedFields**() const
- virtual int **numVarFields**() const
- virtual [DtDatumParam](#) **varDatumId**(int index) const
- virtual [DtDatumParam](#) **fixedDatumId**(int index) const
- virtual [DtDatumParam](#) **datumId**([DtDatumRecordType](#) fixedOrVar, int index) const
- virtual int **datumSizeBits**([DtDatumRecordType](#) fixedOrVar, int index) const
- virtual int **datumSizeBytes**([DtDatumRecordType](#) fixedOrVar, int index) const
- virtual void **getIndex**([DtDatumParam](#) param, [DtDatumRecordType](#)* fixedOrVar, int* index, [DtDatumRetCode](#)* retcode = NULL) const
- virtual DtFloat32 **datumValFloat32**([DtDatumRecordType](#) fixedOrVar, int index, [DtDatumRetCode](#)* retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**([DtDatumRecordType](#) fixedOrVar, int index, [DtDatumRetCode](#)* retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**([DtDatumRecordType](#) fixedOrVar, int index, [DtDatumRetCode](#)* retcode = NULL) const
- virtual long **datumValInt32**([DtDatumRecordType](#) fixedOrVar, int index, [DtDatumRetCode](#)* retcode = NULL) const
- virtual const char* **datumValByteArray**([DtDatumRecordType](#) fixedOrVar, int index) const
- virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL) const
- virtual long **datumValInt32**([DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL) const
- virtual const char* **datumValByteArray**([DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL) const
- virtual void **init**(const [DtDatumParam](#)* fixedParams, const [DtVarDatumSpec](#)* specList)
- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**([DtDatumRecordType](#) fixedOrVar, int index, [DtDatumParam](#) param, [DtDatumRetCode](#)* retcode = NULL)

- virtual void **setVarDataBytes**(int index, int length, [DtDatumRetCode](#)* retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumRecordType](#) fixedOrVar, int index, DtFloat64 val, [DtDatumRetCode](#)* retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumRecordType](#) fixedOrVar, int index, DtFloat32 val, [DtDatumRetCode](#)* retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumRecordType](#) fixedOrVar, int index, long val, [DtDatumRetCode](#)* retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumRecordType](#) fixedOrVar, int index, unsigned long val, [DtDatumRetCode](#)* retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumRecordType](#) fixedOrVar, int index, const char* val, [DtDatumRetCode](#)* retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, [DtDatumRetCode](#)* retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, [DtDatumRetCode](#)* retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, [DtDatumRetCode](#)* retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, [DtDatumRetCode](#)* retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char* val, [DtDatumRetCode](#)* retcode = NULL)
- virtual void **printData**() const
- virtual DtNetDataPduTail* **netDataRep**() const
- virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const
- virtual DtNetVarDatum* **nthVarDataRecord**(int index) const
- virtual void **setFromNetFixed**(const char* fixedFields, int numFixedFields)
- virtual void **setFromNetVar**(const char* varFields, int numVarFields, int totalVarFieldsLength)
- virtual void **sizeChanged**(int size)
- inline DtNetDataPduTail* **netDataRep**() const

Public



Protected Methods

- virtual void* **datumValPtr**([DtDatumRecordType](#) fixedOrVar, int index) const
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

An instance of this class is a DtDatumRecordSet that uses a part of a DtPduWithData as its network representation. Normally, a DtDatumRecord manages the memory associated with its network representation through its DtBaseMsg base class. However, since a DtPduWithData's netRep memory is managed by that DtPduWithData, DtPduDatumRecordSets must defer ITS memory management responsibilities to the DtPduWithData. This is accomplished by overriding changeSize, addBytes and deleteBytes here, with versions that call the DtPduWithData's versions of the functions. No one should need to use this class, except for DtPduWithData.

- DtPduDatumRecordSet** ([DtPduWithData](#)* pdu)
Constructor
- virtual ~DtPduDatumRecordSet** ()
Destructor
- DtPduDatumRecordSet** (const [DtPduDatumRecordSet](#)& orig)
Copy ctor
- [DtPduDatumRecordSet](#)**& operator=(const [DtPduDatumRecordSet](#)& orig)
Assignment operator

● **virtual void changeSize(int size)**

Changes the size of the datumRecSet, by asking my PDU to change size, then reinitializing to use the right piece of the PDU's buffer

● **virtual void insertBytes(int offset, int numBytes)**

Inserts numBytes zeroed-out bytes at the indicated offset into the net rep, by asking my PDU to insert bytes, then reinitializing to use the right piece of the PDU's buffer

● **virtual void insertBytes(void* addr, int numBytes)**

Inserts numBytes zeroed-out bytes at the indicated addr within the net rep, by asking my PDU to insert bytes, then reinitializing to use the right piece of the PDU's buffer

● **virtual void deleteBytes(int offset, int numBytes)**

Deletes numBytes bytes at the indicated offset in the net rep, by asking my PDU to insert bytes, then reinitializing to use the right piece of the PDU's buffer

● **virtual void deleteBytes(void* addr, int numBytes)**

Deletes numBytes bytes at the indicated addr within the net rep, by asking my PDU to insert bytes, then reinitializing to use the right piece of the PDU's buffer

● [DtPduWithData](#)* myPdu

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file pduwithdata.h:

class DtPduWithData: public DtSimanPdu

Inheritance:

DtPduWithData - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtPduWithData () <i>Constructor</i>
 virtual int	numFixedFields () const <i>returns number of fixed fields</i>
 virtual int	numVarFields () const <i>returns number of variable length fields</i>
 virtual DtDatumParam	varDatumId (int index) const <i>Returns the datum param found in the index'th variable datum record</i>
 virtual DtDatumParam	fixedDatumId (int index) const <i>Returns the datum param found in the index'th fixed datum record</i>
 virtual DtDatumParam	datumId (int fixedOrVar, int index) const <i>Returns the datum param found in the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFIXED or DtVAR</i>
 virtual int	datumSizeBits (int fixedOrVar, int index) const <i>Returns the number of bits of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFIXED or DtVAR</i>
 virtual int	datumSizeBytes (int fixedOrVar, int index) const <i>Returns the number of bytes of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFIXED or DtVAR</i>
 virtual void	getIndex (DtDatumParam param, int *fixedOrVar, int *index, int *retcode = NULL) const <i>Looks through all fixed and variable records for one containing param as the parameter ID</i>
 virtual DtFloat32	datumValFloat32 (int fixedOrVar, int index, int *retcode = NULL) const <i>***** INSPECTING DATUM VALUES BY INDEX ***** If you know the data type of the value contained in a particular record, use one of the following to return it</i>
 virtual DtFloat64	datumValFloat64 (int fixedOrVar, int index, int *retcode = NULL) const
 virtual unsigned long	datumValUnsigned32 (int fixedOrVar, int index, int *retcode = NULL) const
 virtual long	datumValInt32 (int fixedOrVar, int index, int *retcode = NULL) const
 virtual const char*	datumValByteArray (int fixedOrVar, int index) const
 virtual DtFloat32	datumValFloat32 (DtDatumParam param, int *retcode = NULL) const <i>***** INSPECTING DATUM VALUES BY PARAM ID ***** If you know the data type of the value contained in a particular record, use one of the following to return it</i>

virtual DtFloat64	datumValFloat64 (DtDatumParam param, int *retcode = NULL) const
virtual unsigned long	datumValUnsigned32 (DtDatumParam param, int *retcode = NULL) const
virtual long	datumValInt32 (DtDatumParam param, int *retcode = NULL) const
virtual const char*	datumValByteArray (DtDatumParam param, int *retcode = NULL) const
virtual void	initDataSection (const DtDatumParam *fixedParams, const DtVarDatumSpec *specList) <i>Initialize the data section of the PDU to reflect the fixed parameters and variable parameters and sizes indicated</i>
virtual void	setNumFixedFields (int newNum) <i>Sets the number of fixed or variable fields to newNum</i>
virtual void	setNumVarFields (int newNum)
virtual void	setDatumParam (int fixedOrVar, int index, DtDatumParam param, int *retcode = NULL) <i>Set the datum ID in the index'th fixed or variable datum record to param, depending on whether fixedOrVar is DtFIXED or DtVAR</i>
virtual void	setVarDataBytes (int index, int length, int *retcode = NULL) <i>Allocates length bytes for the index'th variable datum record</i>
virtual void	setDatumValFloat64 (int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL) ***** SETTING DATA VALUES BY INDEX ***** <i>Use one of the following to set the data value in a fixed or variable datum record, depending on the data type of your data</i>
virtual void	setDatumValFloat32 (int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
virtual void	setDatumValInt32 (int fixedOrVar, int index, long val, int *retcode = NULL)
virtual void	setDatumValUnsigned32 (int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
virtual void	setDatumValByteArray (int fixedOrVar, int index, const char *val, int *retcode = NULL)
virtual void	setDatumValFloat64 (DtDatumParam param, DtFloat64 val, int *retcode = NULL) ***** SETTING DATA VALUES BY PARAM ID ***** <i>Use one of the following to set the data value in a fixed or variable datum record, depending on the data type of your data</i>
virtual void	setDatumValFloat32 (DtDatumParam param, DtFloat32 val, int *retcode = NULL)
virtual void	setDatumValInt32 (DtDatumParam param, long val, int *retcode = NULL)
virtual void	setDatumValUnsigned32 (DtDatumParam param, unsigned long val, int *retcode = NULL)
virtual void	setDatumValByteArray (DtDatumParam param, const char *val, int *retcode = NULL)
virtual void	printData () const <i>Implements DtPdu's virtual function to print the PDU's data</i>
virtual void	printDataSection () const <i>Helper function that prints only the contents of the data records</i>
virtual DtDatumRecordSet *	datumRecSet () <i>Returns our datum record set helper object</i>

Public

***** MUTATOR FUNCTIONS *****

Protected Fields

• [DtDatumRecordSet](#)* myRecSet

Protected Methods

- virtual const [DtDatumRecordSet](#)* **datumRecSet** () const
Const version of above
- virtual DtNetDataPduTail* **dataSection** () const
Returns a pointer to the data section of the PDU - the part that starts with the numberOfFixedFields field
- virtual DtNetFixDatum* **nthFixedDataRecord** (int index) const
Return a pointer to the index'th fixed data record in the network representation
- virtual DtNetVarDatum* **nthVarDataRecord** (int index) const
Return a pointer to the index'th variable data record in the network representation
- virtual void* **datumValPtr** (int fixedOrVar, int index) const
Returns a pointer to the index'th fixed or variable record in the network representation, depending on whether fixedOrVar is DtFIXED or DtVAR
- virtual **~DtPduWithData** ()
Destructor
-

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

• [DtEntityIdentifier](#) cache

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtPduKind](#) **internalGetPduKind**() const
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)

- virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtPduWithData()**
Constructor
- virtual int numFixedFields() const**
returns number of fixed fields
- virtual int numVarFields() const**
returns number of variable length fields
- virtual [DtDatumParam](#) varDatumId(int index) const**
Returns the datum param found in the index'th variable datum record. Index is between 1 and numRecords. Returns DtDatumParam(0) if index is out of bounds.
- virtual [DtDatumParam](#) fixedDatumId(int index) const**
Returns the datum param found in the index'th fixed datum record. Index is between 1 and numRecords. Returns DtDatumParam(0) if index is out of bounds.
- virtual [DtDatumParam](#) datumId(int fixedOrVar, int index) const**
Returns the datum param found in the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFIXED or DtVAR. Index is between 1 and numRecords. Returns DtDatumParam(0) if index is out of bounds.
- virtual int datumSizeBits(int fixedOrVar, int index) const**
Returns the number of bits of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFIXED or DtVAR. If DtFIXED, the answer is always 32.
- virtual int datumSizeBytes(int fixedOrVar, int index) const**
Returns the number of bytes of data contained in the value fields of the index'th fixed or variable datum record, depending on whether fixedOrVar is DtFIXED or DtVAR. If DtFIXED, the answer is always 4.
- virtual void getIndex([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const**
Looks through all fixed and variable records for one containing param as the parameter ID. If it finds one, fills supplied fixedOrVar pointer with DtFIXED or DtVAR, and index with the index to indicate which record was found, and fills retcode with DtDATUM_VALID. If param was not found, retcode is filled with DtDATUM_BAD_ID, and index and fixedOrVar are invalid.
- virtual DtFloat32 datumValFloat32(int fixedOrVar, int index, int *retcode = NULL) const**
***** INSPECTING DATUM VALUES BY INDEX ***** If you know the data type of the value contained in a particular record, use one of the following to return it. fixedOrVar (either DtFIXED or DtVAR) and index (between 1

and numRecords) indicate which record you are inspecting. datumValByteArray is used when the data type is a character string, or something other than a long, unsigned long, float or double. It returns a pointer to a character string containing the exact bytes in the PDU's network representation that represent the value being inspected. The supplied retcode pointer will be filled with DtDATUM_VALID if the value was successfully inspected, and DtDATUM_BAD_LENGTH if either the index was invalid, or the size of the data indicated that it could not be interpreted as the chosen data type.

- `virtual DtFloat64 datumValFloat64(int fixedOrVar, int index, int *retcode = NULL) const`
- `virtual unsigned long datumValUnsigned32(int fixedOrVar, int index, int *retcode = NULL) const`
- `virtual long datumValInt32(int fixedOrVar, int index, int *retcode = NULL) const`
- `virtual const char* datumValByteArray(int fixedOrVar, int index) const`
- `virtual DtFloat32 datumValFloat32(DtDatumParam param, int *retcode = NULL) const`

***** INSPECTING DATUM VALUES BY PARAM ID ***** If you know the data type of the value contained in a particular record, use one of the following to return it. The particular record inspected will be the first one containing param as the datum ID. datumValByteArray is used when the data type is a character string, or something other than a long, unsigned long, float or double. It returns a pointer to a character string containing the exact bytes in the PDU's network representation that represent the value being inspected. The supplied retcode pointer will be filled with DtDATUM_VALID if the value was successfully inspected, and DtDATUM_BAD_LENGTH if either param was not found, or the size of the data indicated that it could not be interpreted as the chosen data type.

- `virtual DtFloat64 datumValFloat64(DtDatumParam param, int *retcode = NULL) const`
- `virtual unsigned long datumValUnsigned32(DtDatumParam param, int *retcode = NULL) const`
- `virtual long datumValInt32(DtDatumParam param, int *retcode = NULL) const`
- `virtual const char* datumValByteArray(DtDatumParam param, int *retcode = NULL) const`

***** MUTATOR FUNCTIONS *****

- `virtual void initDataSection(const DtDatumParam *fixedParams, const DtVarDatumSpec *specList)`

Initialize the data section of the PDU to reflect the fixed parameters and variable parameters and sizes indicated. fixedParams is a NULL-terminated array of DtDatumParams. specList is an array of DtVarDatumSpecs terminated with the DtNullVarDatumSpec constant. This call can be made instead of calls to setNumVarDatumRecords, setNumFixedDatumRecord, and repeated calls to setDatumParam and setVarDataBytes.

- `virtual void setNumFixedFields(int newNum)`
Sets the number of fixed or variable fields to newNum. Previously existing fixed fields with indices greater than newNum are deleted. If newNum exceeds the number of existing fields, trailing new fields are created with zeros as datum params, sizes (for var only) and values.
- `virtual void setNumVarFields(int newNum)`
- `virtual void setDatumParam(int fixedOrVar, int index, DtDatumParam param, int *retcode = NULL)`
Set the datum ID in the index'th fixed or variable datum record to param, depending on whether fixedOrVar is DtFIXED or DtVAR. Supplied retcode pointer is filled with DtDATUM_VALID if successful, or DtDATUM_BAD_INDEX if the index is not valid.
- `virtual void setVarDataBytes(int index, int length, int *retcode = NULL)`

Allocates length bytes for the index'th variable datum record. Should be called before using one of the setDatumVal functions to actually set the data. Supplied retcode pointer is set to DtDATUM_VALID if successful, or DtDATUM_BAD_INDEX if the index is not valid.

virtual void setDatumValFloat64(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)

***** SETTING DATA VALUES BY INDEX ***** Use one of the following to set the data value in a fixed or variable datum record, depending on the data type of your data. fixedOrVar (either DtFIXED or DtVAR) and index (between 1 and numRecords) indicate which record you are modifying. setDatumValByteArray is used when the data type is a character string, or something other than a long, unsigned long, float or double. It takes a pointer to a character string containing the exact bytes that should be used in the PDU's network representation. The supplied retcode pointer will be filled with DtDATUM_VALID if the value was successfully set, DtDATUM_BAD_INDEX if the index is invalid, and DtDATUM_BAD_LENGTH if the size of the data indicated that it could not be set from as the chosen data type.

virtual void setDatumValFloat32(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)

virtual void setDatumValInt32(int fixedOrVar, int index, long val, int *retcode = NULL)

virtual void setDatumValUnsigned32(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)

virtual void setDatumValByteArray(int fixedOrVar, int index, const char *val, int *retcode = NULL)

virtual void setDatumValFloat64([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)

***** SETTING DATA VALUES BY PARAM ID ***** Use one of the following to set the data value in a fixed or variable datum record, depending on the data type of your data. The particular record modified will be the first one containing param as its parameter ID. setDatumValByteArray is used when the data type is a character string, or something other than a long, unsigned long, float or double. It takes a pointer to a character string containing the exact bytes that should be used in the PDU's network representation. The supplied retcode pointer will be filled with DtDATUM_VALID if the value was successfully set, DtDATUM_BAD_ID if param is invalid, and DtDATUM_BAD_LENGTH if the size of the data indicated that it could not be set from the chosen data type.

virtual void setDatumValFloat32([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)

virtual void setDatumValInt32([DtDatumParam](#) param, long val, int *retcode = NULL)

virtual void setDatumValUnsigned32([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)

virtual void setDatumValByteArray([DtDatumParam](#) param, const char *val, int *retcode = NULL)

virtual void printData() const

Implements DtPdu's virtual function to print the PDU's data

virtual void printDataSection() const

Helper function that prints only the contents of the data records

virtual [DtDatumRecordSet](#)* datumRecSet()

Returns our datum record set helper object

virtual const [DtDatumRecordSet](#)* datumRecSet() const

Const version of above

virtual DtNetDataPduTail* dataSection() const

Returns a pointer to the data section of the PDU - the part that starts with the numberOfFixedFields field

● **virtual DtNetFixDatum* nthFixedDataRecord(int index) const**

Return a pointer to the index'th fixed data record in the network representation. Index is between 1 and numRecords.

● **virtual DtNetVarDatum* nthVarDataRecord(int index) const**

Return a pointer to the index'th variable data record in the network representation. Index is between 1 and numRecords.

● **virtual void* datumValPtr(int fixedOrVar, int index) const**

Returns a pointer to the index'th fixed or variable record in the network representation, depending on whether fixedOrVar is DtFIXED or DtVAR

● **virtual ~DtPduWithData()**

Destructor

● [**DtDatumRecordSet**](#)* myRecSet

Direct child classes:

[DtSetDataRPdu](#)

[DtSetDataPdu](#)

[DtEventReptRPdu](#)

[DtEventReportPdu](#)

[DtDataRPdu](#)

[DtDataPdu](#)

[DtCommentRPdu](#)

[DtCommentPdu](#)

[DtActionResponseRPdu](#)

[DtActionResponsePdu](#)

[DtActionRequestRPdu](#)

[DtActionRequestPdu](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file simanpdu.h:

class DtSimanPdu: public DtInteraction

Inheritance:

DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId** () const
- virtual const [DtEntityIdentifier](#)& **receiverId** () const
- virtual void **setSenderId** (const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId** (const [DtEntityIdentifier](#) &id)
- virtual void **printData** () const

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from DtInteraction:

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from DtPdu:

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const

- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtPduKind](#) **internalGetPduKind**() const
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)
- virtual void **insertBytes**(void *address, int numBytes)
- virtual void **deleteBytes**(int offset, int numBytes)

- virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- virtual const [DtEntityIdentifier](#)& senderId() const
 - virtual const [DtEntityIdentifier](#)& receiverId() const
 - virtual void setSenderId(const [DtEntityIdentifier](#) &id)
 - virtual void setReceiverId(const [DtEntityIdentifier](#) &id)
 - virtual void printData() const
 - [DtEntityIdentifier](#) cache
-

Direct child classes:

[DtStopFreezeRPdu](#)
[DtStopFreezePdu](#)
[DtStartResumeRPdu](#)
[DtStartResumePdu](#)
[DtRemoveEntityRPdu](#)
[DtRemoveEntityPdu](#)
[DtPduWithData](#)
[DtMessagePdu](#)
[DtDataQueryRPdu](#)
[DtDataQueryPdu](#)
[DtCreateEntityRPdu](#)
[DtCreateEntityPdu](#)
[DtAcknowledgeRPdu](#)
[DtAcknowledgePdu](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file dinteraction.h:

class DtInteraction: public DtPdu

class DtInteraction: Instances of DtInteraction are used for representing DIS events

Inheritance:

DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtInteraction ()**
default constructor
 - virtual **~DtInteraction ()**
destructor
 - **DtInteraction** (const [DtInteraction](#)& orig)
copy constructor
 - [DtInteraction](#)& **operator=** (const [DtInteraction](#)& orig)
assignment operator
-

Inherited from DtPdu:

Public Methods

- virtual void* **packet()** const
- virtual int **status()** const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion()** const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld()** const
- virtual [DtPduKind](#) **kind()** const
- virtual [DtProtocolFamily](#) **protocolFamily()** const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp()** const

- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- virtual void **printData**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtPduKind](#) **internalGetPduKind**() const
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()

virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

virtual const char* **netRep**() const

virtual int **length**() const

inline const char* **netRep**() const

Protected Fields

void* **myNetMessage**

void* **myInternalBuff**

int **myIntBuffSize**

int **myMsgSize**

Protected Methods

void **initMsg**(int size, DtBufferPtr bufferPtr)

void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)

virtual void **changeSize**(int size)

virtual void **insertBytes**(int offset, int numBytes)

virtual void **insertBytes**(void *address, int numBytes)

virtual void **deleteBytes**(int offset, int numBytes)

virtual void **deleteBytes**(void *address, int numBytes)

virtual void **clearAll**()

virtual void* **newInternalBuffer**(int size)

Documentation

class DtInteraction: Instances of DtInteraction are used for representing DIS events

DtInteraction()
default constructor

virtual ~DtInteraction()
destructor

 `DtInteraction(const DtInteraction& orig)`

copy constructor

 `DtInteraction& operator=(const DtInteraction& orig)`

assignment operator

Direct child classes:

[DtViewControlPdu](#)

[DtSimanPdu](#)

[DtSignalPdu](#)

[DtLgrControlPdu](#)

[DtFirePdu](#)

[DtDetonationPdu](#)

[DtCollisionPdu](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtPduKind

- DtOtherPduKind
- DtEntityStatePduKind
- DtFirePduKind
- DtDetonationPduKind
- DtCollisionPduKind
- DtServiceRequestPduKind
- DtResupplyOfferPduKind
- DtResupplyReceivedPduKind
- DtResupplyCancelPduKind
- DtRepairCompletePduKind
- DtRepairResponsePduKind
- DtCreateEntityPduKind
- DtRemoveEntityPduKind
- DtStartResumePduKind
- DtStopFreezePduKind
- DtAcknowledgePduKind
- DtActionRequestPduKind
- DtActionResponsePduKind
- DtDataQueryPduKind
- DtSetDataPduKind
- DtDataPduKind
- DtEventReportPduKind
- DtCommentPduKind
- DtMessagePduKind
- DtElectromagneticEmissionPduKind
- DtDesignatorPduKind

- DtTransmitterPduKind
- DtSignalPduKind
- DtReceiverPduKind
- DtIffAtcNav aidsPduKind
- DtUnderwaterAcousticPduKind
- DtSuppEmisEntStatePduKind
- DtIntercomSignalPduKind
- DtIntercomControlPduKind
- DtAggregateStatePduKind
- DtIsGroupOfPduKind
- DtTransferControlRequestPduKind
- DtIsPartOfPduKind
- DtMinefieldStatePduKind
- DtMinefieldQueryPduKind
- DtMinefieldDataPduKind
- DtMinefieldResponseNackPduKind
- DtEnvironmentalProcessPduKind
- DtGriddedDataPduKind
- DtPointObjectStatePduKind
- DtLinearObjectStatePduKind
- DtArealObjectStatePduKind
- DtLeTspiPduKind
- DtLeAppearancePduKind
- DtLeArticulatedPartsPduKind
- DtLeFirePduKind
- DtLeDetonationPduKind
- DtCreateEntityRPduKind
- DtRemoveEntityRPduKind
- DtStartResumeRPduKind
- DtStopFreezeRPduKind
- DtAcknowledgeRPduKind
- DtActionRequestRPduKind

- DtActionResponseRPduKind
 - DtDataQueryRPduKind
 - DtSetDataRPduKind
 - DtDataRPduKind
 - DtEventReptRPduKind
 - DtCommentRPduKind
 - DtRecordQueryPduKind
 - DtSetRecordPduKind
 - DtRecordPduKind
 - DtCollisionElasticPduKind
 - DtEntityStateUpdatePduKind
-

Documentation

- DtOtherPduKind
- DtEntityStatePduKind
- DtFirePduKind
- DtDetonationPduKind
- DtCollisionPduKind
- DtServiceRequestPduKind
- DtResupplyOfferPduKind
- DtResupplyReceivedPduKind
- DtResupplyCancelPduKind
- DtRepairCompletePduKind
- DtRepairResponsePduKind
- DtCreateEntityPduKind
- DtRemoveEntityPduKind
- DtStartResumePduKind
- DtStopFreezePduKind
- DtAcknowledgePduKind
- DtActionRequestPduKind
- DtActionResponsePduKind

- DtDataQueryPduKind
- DtSetDataPduKind
- DtDataPduKind
- DtEventReportPduKind
- DtCommentPduKind
- DtMessagePduKind
- DtElectromagneticEmissionPduKind
- DtDesignatorPduKind
- DtTransmitterPduKind
- DtSignalPduKind
- DtReceiverPduKind
- DtIffAtcNavAidsPduKind
- DtUnderwaterAcousticPduKind
- DtSuppEmisEntStatePduKind
- DtIntercomSignalPduKind
- DtIntercomControlPduKind
- DtAggregateStatePduKind
- DtIsGroupOfPduKind
- DtTransferControlRequestPduKind
- DtIsPartOfPduKind
- DtMinefieldStatePduKind
- DtMinefieldQueryPduKind
- DtMinefieldDataPduKind
- DtMinefieldResponseNackPduKind
- DtEnvironmentalProcessPduKind
- DtGriddedDataPduKind
- DtPointObjectStatePduKind
- DtLinearObjectStatePduKind
- DtArealObjectStatePduKind
- DtLeTspiPduKind
- DtLeAppearancePduKind
- DtLeArticulatedPartsPduKind

- DtLeFirePduKind
- DtLeDetonationPduKind
- DtCreateEntityRPduKind
- DtRemoveEntityRPduKind
- DtStartResumeRPduKind
- DtStopFreezeRPduKind
- DtAcknowledgeRPduKind
- DtActionRequestRPduKind
- DtActionResponseRPduKind
- DtDataQueryRPduKind
- DtSetDataRPduKind
- DtDataRPduKind
- DtEventReptRPduKind
- DtCommentRPduKind
- DtRecordQueryPduKind
- DtSetRecordPduKind
- DtRecordPduKind
- DtCollisionElasticPduKind
- DtEntityStateUpdatePduKind

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtProtocolFamily

-  DtFamilyOther
 -  DtFamilyEntityInteraction
 -  DtFamilyWarfare
 -  DtFamilyLogistics
 -  DtFamilyRadioCommunication
 -  DtFamilySimulationManagement
 -  DtFamilyDistributedEmissionRegeneration
 -  DtFamilyEntityManagement
 -  DtFamilyMinefield
 -  DtFamilySyntheticEnvironment
 -  DtFamilySimulationManagementWithReliability
 -  DtFamilyLiveEntity
 -  DtFamilyNonRealTime
-

Documentation

-  DtFamilyOther
-  DtFamilyEntityInteraction
-  DtFamilyWarfare
-  DtFamilyLogistics
-  DtFamilyRadioCommunication
-  DtFamilySimulationManagement
-  DtFamilyDistributedEmissionRegeneration
-  DtFamilyEntityManagement
-  DtFamilyMinefield
-  DtFamilySyntheticEnvironment

- DtFamilySimulationManagementWithReliability
- DtFamilyLiveEntity
- DtFamilyNonRealTime

[Alphabetic index](#) [Hierarchy of classes](#)

In file tstamptype.h:

enum DtTimeStampType

-
-  DtTimeStampRelative
 -  DtTimeStampAbsolute
-

Documentation

-  DtTimeStampRelative
-  DtTimeStampAbsolute

In file vcpdu.h:

class DtViewControlPdu: public DtInteraction

Inheritance:

DtViewControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static void	addCallback (DtExerciseConn *conn, DtViewControlInteractionCb cb, void *usr)
static void	removeCallback (DtExerciseConn *conn, DtViewControlInteractionCb cb, void *usr)
const DtGlobalObjectDesignator&	viewerId () const
const DtGlobalObjectDesignator&	senderId () const
DtViewerCommand	viewerCommand () const
unsigned long	ignoreFlags () const
void	setViewerId (const DtGlobalObjectDesignator& id)
void	setSenderId (const DtGlobalObjectDesignator& id)
void	setIgnoreFlags (unsigned long flags)
virtual void	setFromString (const char *)
virtual char*	string () const
void	printData () const

Protected Fields

[DtGlobalObjectDesignator](#) **retId**

Protected Methods

[DtPduKind](#) **internalGetPduKind** () const

void **setViewerCommand** ([DtViewerCommand](#) com)

Inherited from [DtInteraction](#):

Public Methods

• [DtInteraction](#)& **operator**=(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

• virtual void* **packet**() const

• virtual int **status**() const

• virtual void **setVersion**(int vers)

• virtual int **protocolVersion**() const

• virtual void **setExerciseld**(int id)

• virtual int **exerciseld**() const

• virtual [DtPduKind](#) **kind**() const

• virtual [DtProtocolFamily](#) **protocolFamily**() const

• virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

• virtual DtTime **timeStamp**() const

• virtual [DtTimeStampType](#) **timeStampType**() const

• virtual DtTime **guessTimeValid**(DtTime referenceTime) const

• virtual void **print**() const

• virtual void **printHeader**() const

• inline int **exerciseld**() const

• inline [DtPduKind](#) **kind**() const

• inline DtNetPduHeader* **netHeader**() const

• inline void* **packet**() const

• inline [DtProtocolFamily](#) **protocolFamily**() const

• inline int **protocolVersion**() const

• inline void **setExerciseld**(int id)

• inline void **setStatus**(int okStatus)

- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - static void addCallback([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
 - static void removeCallback([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
 - const DtGlobalObjectDesignator& viewerId() const
 - const DtGlobalObjectDesignator& senderId() const
 - [DtViewerCommand](#) viewerCommand() const
 - unsigned long ignoreFlags() const
 - void setViewerId(const DtGlobalObjectDesignator& id)
 - void setSenderId(const DtGlobalObjectDesignator& id)
 - void setIgnoreFlags(unsigned long flags)
 - virtual void setFromString(const char *)
 - virtual char* string() const
 - void printData() const
 - [DtPduKind](#) internalGetPduKind() const
 - DtGlobalObjectDesignator retId
 - void setViewerCommand([DtViewerCommand](#) com)
-

Direct child classes:

[DtUnknownVCPdu](#)
[DtTrackVCPdu](#)
[DtTetherVCPdu](#)
[DtTetherTrackVCPdu](#)
[DtOrbitVCPdu](#)
[DtModelessVCPdu](#)
[DtMimicVCPdu](#)
[DtMimicTrackVCPdu](#)
[DtGroupVCPdu](#)
[DtCompassVCPdu](#)
[DtAbsoluteVCPdu](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtViewerCommand

-  DtViewerAbsolute
 -  DtViewerTether
 -  DtViewerCompass
 -  DtViewerMimic
 -  DtViewerTurret
 -  DtViewerOrbit
 -  DtViewerTrack
 -  DtViewerTetherTrack
 -  DtViewerMimicTrack
 -  DtViewerGroup
 -  DtViewerModeless
-

Documentation

-  DtViewerAbsolute
-  DtViewerTether
-  DtViewerCompass
-  DtViewerMimic
-  DtViewerTurret
-  DtViewerOrbit
-  DtViewerTrack
-  DtViewerTetherTrack
-  DtViewerMimicTrack
-  DtViewerGroup
-  DtViewerModeless

In file vcpdu.h:

class DtUnknownVCPdu: public DtViewControlPdu

Inheritance:

DtUnknownVCPdu - DtViewControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

● **DtUnknownVCPdu** (const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Inherited from DtViewControlPdu:

Public Methods

● static DtPdu* **create**(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

● static void **addCallback**(DtExerciseConn *conn, DtViewControlInteractionCb cb, void *usr)

● static void **removeCallback**(DtExerciseConn *conn, DtViewControlInteractionCb cb, void *usr)

● const DtGlobalObjectDesignator& **viewerId**() const

● const DtGlobalObjectDesignator& **senderId**() const

● DtViewerCommand **viewerCommand**() const

● unsigned long **ignoreFlags**() const

● void **setViewerId**(const DtGlobalObjectDesignator& id)

● void **setSenderId**(const DtGlobalObjectDesignator& id)

● void **setIgnoreFlags**(unsigned long flags)

● virtual void **setFromString**(const char *)

● virtual char* **string**() const

● void **printData**() const

Protected Fields

• DtGlobalObjectDesignator **retId**

Protected Methods

• [DtPduKind](#) **internalGetPduKind()** const

• void **setViewerCommand**([DtViewerCommand](#) com)

Inherited from [DtInteraction](#):

Public Methods

• [DtInteraction](#)& **operator**=(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

• virtual void* **packet**() const

• virtual int **status**() const

• virtual void **setVersion**(int vers)

• virtual int **protocolVersion**() const

• virtual void **setExerciseld**(int id)

• virtual int **exerciseld**() const

• virtual [DtPduKind](#) **kind**() const

• virtual [DtProtocolFamily](#) **protocolFamily**() const

• virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

• virtual DtTime **timeStamp**() const

• virtual [DtTimeStampType](#) **timeStampType**() const

• virtual DtTime **guessTimeValid**(DtTime referenceTime) const

• virtual void **print**() const

• virtual void **printHeader**() const

• inline int **exerciseld**() const

• inline [DtPduKind](#) **kind**() const

- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

 `DtUnknownVCPdu(const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vcpdu.h:

class DtTrackVCPdu: public DtViewControlPdu

Inheritance:

DtTrackVCPdu - DtViewControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtTrackVCPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtTrackVCPdu (const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 DtConstVector	worldLocation () const
 const DtGlobalObjectDesignator&	trackId () const
 void	setWorldLocation (DtConstVector loc)
 void	setTrackId (const DtGlobalObjectDesignator& id)
 void	setFromString (const char *str)
 char*	string () const
 void	printData () const

Protected Fields

 DtVector	theVec
 DtGlobalObjectDesignator	theId

Inherited from DtViewControlPdu:

Public Methods

	static DtPdu * create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	static void addCallback (DtExerciseConn *conn, DtViewControlInteractionCb cb, void *usr)

- static void **removeCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- const DtGlobalObjectDesignator& **viewerId**() const
- const DtGlobalObjectDesignator& **senderId**() const
- [DtViewerCommand](#) **viewerCommand**() const
- unsigned long **ignoreFlags**() const
- void **setViewerId**(const DtGlobalObjectDesignator& id)
- void **setSenderId**(const DtGlobalObjectDesignator& id)
- void **setIgnoreFlags**(unsigned long flags)

Protected Fields

- DtGlobalObjectDesignator **retId**

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
 - void **setViewerCommand**([DtViewerCommand](#) com)
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const

- virtual [DtProtocolFamily](#) **protocolFamily()** const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp()** const
- virtual [DtTimeStampType](#) **timeStampType()** const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader()** const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)

- virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtTrackVCPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtTrackVCPdu(const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

- `DtConstVector worldLocation() const`
- `const DtGlobalObjectDesignator& trackId() const`
- `void setWorldLocation(DtConstVector loc)`
- `void setTrackId(const DtGlobalObjectDesignator& id)`
- `void setFromString(const char *str)`
- `char* string() const`
- `void printData() const`
- [DtVector](#) theVec
- `DtGlobalObjectDesignator theId`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vvector.h:

class DtVector

class DtVector: No virtual functions, because we want the size of a DtVector to 3 * 8 bytes - so that you can make the same assumptions about it that we used to make about its size and layout

Inheritance:

DtVector

Public Methods

	DtVector () <i>default constructor</i>
	DtVector (double x, double y, double z) <i>construct from three doubles</i>
	~DtVector () <i>destructor</i>
	DtVector (const DtVector & vec) <i>copy constructor</i>
DtVector &	operator= (const DtVector & vec) <i>assignment operator</i>
	operator double* () <i>Convert the vector to a pointer to double</i>
	operator const double* () const
double&	operator[] (int ind) <i>Get the value for the specified subscript</i>
double	operator[] (int ind) const
double&	operator[] (DtVectorOffset ind) <i>Get the value for the specified subscript</i>
double	operator[] (DtVectorOffset ind) const
int	operator== (const DtVector & vec) const <i>Get whether another vector equals this vector</i>
const char*	string () const <i>Get a string representation of the vector</i>
static DtConstVector	zero () <i>Get a zero'ed vector</i>
static DtConstVector	i () <i>Get a (1,0,0) vector</i>
static DtConstVector	j () <i>Get a (0,1,0) vector</i>

- static DtConstVector **k** ()
Get a (0,0,1) vector
- static DtConstVector **ones** ()
Get a vector set to all ones
- inline **DtVector** ()
- inline **DtVector** (double a, double b, double c)
- inline **DtVector** (const [DtVector](#)& vec)
- inline **operator const double*** () const
- inline **operator double*** ()
- inline [DtVector](#)& **operator=** (const [DtVector](#)& vec)
- inline int **operator==** (const [DtVector](#)& vec) const
- inline double& **operator[]** ([DtVectorOffset](#) ind)
Inline DtVector functions
- inline double **operator[]** ([DtVectorOffset](#) ind) const
- inline double& **operator[]** (int ind)
- inline double **operator[]** (int ind) const
- inline **~DtVector** ()

Protected Fields

- double **myRep** [3]
-

Documentation

class DtVector:

No virtual functions, because we want the size of a DtVector to 3 * 8 bytes - so that you can make the same assumptions about it that we used to make about its size and layout. This is especially important in DtDcm, where we use an array of 3 DtVectors as the myRep. If DtVector had any virtual functions, there would be 4 bytes between vector reps.

- DtVector** ()
default constructor
- DtVector**(double x, double y, double z)
construct from three doubles
- ~DtVector** ()
destructor
- DtVector**(const [DtVector](#)& vec)
copy constructor
- [DtVector](#)& **operator=**(const [DtVector](#)& vec)
assignment operator

- `operator double* ()`
Convert the vector to a pointer to double
- `operator const double* () const`
- `double& operator[] (int ind)`
Get the value for the specified subscript. (int version)
- `double operator[] (int ind) const`
- `double& operator[] (DtVectorOffset ind)`
Get the value for the specified subscript. (enum version)
- `double operator[] (DtVectorOffset ind) const`
- `int operator==(const DtVector& vec) const`
Get whether another vector equals this vector
- `const char* string() const`
Get a string representation of the vector
- `static DtConstVector zero()`
Get a zero'ed vector
- `static DtConstVector i()`
Get a (1,0,0) vector
- `static DtConstVector j()`
Get a (0,1,0) vector
- `static DtConstVector k()`
Get a (0,0,1) vector
- `static DtConstVector ones()`
Get a vector set to all ones
- `double myRep[3]`
- `inline DtVector()`
- `inline DtVector(double a, double b, double c)`
- `inline DtVector(const DtVector& vec)`
- `inline operator const double* () const`
- `inline operator double* ()`
- `inline DtVector& operator=(const DtVector& vec)`
- `inline int operator==(const DtVector& vec) const`
- `inline double& operator[] (DtVectorOffset ind)`
Inline DtVector functions
- `inline double operator[] (DtVectorOffset ind) const`

- `inline double& operator[] (int ind)`
- `inline double operator[] (int ind) const`
- `inline ~DtVector()`

Direct child classes:

[DtUtmCoord](#)

[DtGeodeticCoord](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file vlvector.h:

enum DtVectorOffset

 DtX

 DtY

 DtZ

Documentation

 DtX

 DtY

 DtZ

In file utmcoord.h:

class DtUtmCoord: public DtVector

class DtUtmCoord: Instances of DtUtmCoord are used to represent UTM coordinates

Inheritance:

DtUtmCoord - DtVector

Public Methods

	DtUtmCoord () <i>default constructor</i>
	DtUtmCoord (double east, double north, double up) <i>constructor</i>
	DtUtmCoord (const DtGeodeticCoord & geodetic) <i>constructor, construct UTM from geodetic</i>
 virtual	~DtUtmCoord () <i>destructor</i>
	DtUtmCoord (const DtUtmCoord & orig) <i>copy constructor</i>
 DtUtmCoord &	operator= (const DtUtmCoord & orig) <i>assignment operator</i>
 DtUtmCoord &	operator= (const DtGeodeticCoord & geodetic) <i>assignment operator, convert geodetic to UTM</i>
 virtual DtUtmCoord &	setGeodetic (const DtGeodeticCoord & geodetic) <i>Set from geodetic coordinate (convert geodetic to UTM)</i>
 virtual void	getGeodetic (DtGeodeticCoord & geodetic) const <i>Get geodetic coordinate equivalent (convert UTM to geodetic)</i>
 virtual DtGeodeticCoord	geodetic () const <i>Get geodetic coordinate equivalent (convert UTM to geodetic)</i>
 virtual DtUtmCoord &	setGeocentric (DtConstVector geocentric) <i>Set from geocentric coordinate (convert geocentric to UTM)</i>
 virtual void	getGeocentric (DtVectorRef geocentric) const <i>Get geocentric coordinate equivalent (convert UTM to geocentric)</i>
 virtual DtVector	geocentric () const <i>Get geocentric coordinate equivalent (convert UTM to geocentric)</i>
 virtual void	setEast (DtFloat64 east) <i>Set/Get east component</i>
 virtual DtFloat64	east () const
 virtual void	setNorth (DtFloat64 north) <i>Set/Get north component</i>

virtual DtFloat64	north () const
virtual void	setUp (DtFloat64 up) <i>Set/Get up component</i>
virtual DtFloat64	up () const
virtual double	gridDeclination () const <i>Get the heading that northing is off of north</i>
inline DtFloat64	east () const
inline DtFloat64	north () const
inline void	setEast (DtFloat64 east) <i>inline functions</i>
inline void	setNorth (DtFloat64 north)
inline void	setUp (DtFloat64 up)
inline DtFloat64	up () const

Public

Note: For coordinate conversions, *DtUtmInit* must be called first

Private Methods

- friend void **DtGeodToUtm** (const [DtGeodeticCoord](#)& gd, [DtUtmCoord](#)& utm)
 - friend void **DtGeocToUtm** (DtConstVector gd, [DtUtmCoord](#)& utm)
-

Inherited from [DtVector](#):

Public Methods

- operator double*** ()
- operator const double*** () const
- double& **operator[]**(int ind)
- double **operator[]**(int ind) const
- double& **operator[]**([DtVectorOffset](#) ind)
- double **operator[]**([DtVectorOffset](#) ind) const
- int **operator==(**const [DtVector](#)& vec) const
- const char* **string**() const
- static DtConstVector **zero**()

- static DtConstVector **i**()
- static DtConstVector **j**()
- static DtConstVector **k**()
- static DtConstVector **ones**()
- inline **operator const double*** () const
- inline **operator double*** ()
- inline int **operator==(**const [DtVector](#)& vec) const
- inline double& **operator[]**([DtVectorOffset](#) ind)
- inline double **operator[]**([DtVectorOffset](#) ind) const
- inline double& **operator[]**(int ind)
- inline double **operator[]**(int ind) const

Protected Fields

- double **myRep**[3]
-

Documentation

class DtUtmCoord:

Instances of DtUtmCoord are used to represent UTM coordinates. Universal Transverse Mercator projection coordinate with respect to reference set by DtUtmInit.

- friend void DtGeodToUtm**(const [DtGeodeticCoord](#)& gd, [DtUtmCoord](#)& utm)
- friend void DtGeocToUtm**(DtConstVector gd, [DtUtmCoord](#)& utm)

Note: For coordinate conversions, DtUtmInit must be called first

- DtUtmCoord()**
default constructor
- DtUtmCoord**(double east, double north, double up)
constructor
- DtUtmCoord**(const [DtGeodeticCoord](#)& geodetic)
constructor, construct UTM from geodetic
- virtual ~DtUtmCoord()**
destructor
- DtUtmCoord**(const [DtUtmCoord](#)& orig)
copy constructor

- `DtUtmCoord& operator=(const DtUtmCoord& orig)`
 assignment operator
 - `DtUtmCoord& operator=(const DtGeodeticCoord& geodetic)`
 assignment operator, convert geodetic to UTM
 - `virtual DtUtmCoord& setGeodetic(const DtGeodeticCoord& geodetic)`
 Set from geodetic coordinate (convert geodetic to UTM). Can also use assignment operator.
 - `virtual void getGeodetic(DtGeodeticCoord& geodetic) const`
 Get geodetic coordinate equivalent (convert UTM to geodetic)
 - `virtual DtGeodeticCoord geodetic() const`
 Get geodetic coordinate equivalent (convert UTM to geodetic). Note that this is not returning a reference.
 - `virtual DtUtmCoord& setGeocentric(DtConstVector geocentric)`
 Set from geocentric coordinate (convert geocentric to UTM). Can also use assignment operator.
 - `virtual void getGeocentric(DtVectorRef geocentric) const`
 Get geocentric coordinate equivalent (convert UTM to geocentric)
 - `virtual DtVector geocentric() const`
 Get geocentric coordinate equivalent (convert UTM to geocentric). Note that this is not returning a reference.
 - `virtual void setEast(DtFloat64 east)`
 Set/Get east component
 - `virtual DtFloat64 east() const`
 - `virtual void setNorth(DtFloat64 north)`
 Set/Get north component
 - `virtual DtFloat64 north() const`
 - `virtual void setUp(DtFloat64 up)`
 Set/Get up component
 - `virtual DtFloat64 up() const`
 - `virtual double gridDeclination() const`
 Get the heading that northing is off of north
 - `inline DtFloat64 east() const`
 - `inline DtFloat64 north() const`
 - `inline void setEast(DtFloat64 east)`
 inline functions
 - `inline void setNorth(DtFloat64 north)`
 - `inline void setUp(DtFloat64 up)`
 - `inline DtFloat64 up() const`
-

This class has no child classes.

[*Alphabetic index*](#) [*Hierarchy of classes*](#)

In file geodcoord.h:

class DtGeodeticCoord: public DtVector

class DtGeodeticCoord: Instances of DtGeodeticCoord are used to explicitly represent Geodetic coordinates

Inheritance:

DtGeodeticCoord - DtVector

Public Methods

	DtGeodeticCoord () <i>default constructor</i>
	DtGeodeticCoord (double lat, double lon, double alt) <i>constructor</i>
virtual	~DtGeodeticCoord () <i>destructor</i>
	DtGeodeticCoord (const DtGeodeticCoord & orig) <i>copy constructor</i>
DtGeodeticCoord &	operator= (const DtGeodeticCoord & orig) <i>assignment operator</i>
virtual DtGeodeticCoord &	setGeocentric (DtConstVector geocentric) <i>Set from geocentric coordinate (convert geocentric to geodetic)</i>
virtual void	getGeocentric (DtVectorRef geocentric) const <i>Get geocentric coordinate equivalent (convert geodetic to geocentric)</i>
virtual DtVector	geocentric () const <i>Get geocentric coordinate equivalent (convert geodetic to geocentric)</i>
virtual void	setLat (DtFloat64 lat) <i>Set/Get the latitude</i>
virtual DtFloat64	lat () const
virtual void	setLon (DtFloat64 lon) <i>Set/Get the longitude</i>
virtual DtFloat64	lon () const
virtual void	setAlt (DtFloat64 alt) <i>Set/Get the altitude</i>
virtual DtFloat64	alt () const
inline DtFloat64	alt () const
inline DtFloat64	lat () const <i>inline functions</i>
inline DtFloat64	lon () const

- inline void **setAlt** (DtFloat64 alt)
- inline void **setLat** (DtFloat64 lat)
- inline void **setLon** (DtFloat64 lon)

Private Methods

- friend void **DtGeocToGeod** (DtConstVector gc, [DtGeodeticCoord](#)& gd)
-

Inherited from [DtVector](#):

Public Methods

- **operator double*** ()
- **operator const double*** () const
- double& **operator[]**(int ind)
- double **operator[]**(int ind) const
- double& **operator[]**([DtVectorOffset](#) ind)
- double **operator[]**([DtVectorOffset](#) ind) const
- int **operator==(const [DtVector](#)& vec)** const
- const char* **string**() const
- static DtConstVector **zero**()
- static DtConstVector **i**()
- static DtConstVector **j**()
- static DtConstVector **k**()
- static DtConstVector **ones**()
- inline **operator const double*** () const
- inline **operator double*** ()
- inline int **operator==(const [DtVector](#)& vec)** const
- inline double& **operator[]**([DtVectorOffset](#) ind)
- inline double **operator[]**([DtVectorOffset](#) ind) const
- inline double& **operator[]**(int ind)
- inline double **operator[]**(int ind) const

Protected Fields

double myRep[3]

Documentation

class DtGeodeticCoord: Instances of DtGeodeticCoord are used to explicitly represent Geodetic coordinates

friend void DtGeocToGeod(DtConstVector gc, [DtGeodeticCoord&](#) gd)

DtGeodeticCoord()
default constructor

DtGeodeticCoord(double lat, double lon, double alt)
constructor

virtual ~DtGeodeticCoord()
destructor

DtGeodeticCoord(const [DtGeodeticCoord&](#) orig)
copy constructor

[DtGeodeticCoord&](#) operator=(const [DtGeodeticCoord&](#) orig)
assignment operator

virtual [DtGeodeticCoord&](#) setGeocentric(DtConstVector geocentric)
Set from geocentric coordinate (convert geocentric to geodetic). Can also use assignment operator.

virtual void getGeocentric(DtVectorRef geocentric) const
Get geocentric coordinate equivalent (convert geodetic to geocentric)

virtual [DtVector](#) geocentric() const
Get geocentric coordinate equivalent (convert geodetic to geocentric). Note that this is not returning a reference.

virtual void setLat(DtFloat64 lat)
Set/Get the latitude

virtual DtFloat64 lat() const

virtual void setLon(DtFloat64 lon)
Set/Get the longitude

virtual DtFloat64 lon() const

virtual void setAlt(DtFloat64 alt)
Set/Get the altitude

virtual DtFloat64 alt() const

inline DtFloat64 alt() const

inline DtFloat64 lat() const

inline functions

- `inline DtFloat64 lon() const`
- `inline void setAlt(DtFloat64 alt)`
- `inline void setLat(DtFloat64 lat)`
- `inline void setLon(DtFloat64 lon)`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vcpdu.h:

class DtTetherVCPdu: public DtViewControlPdu

Inheritance:

DtTetherVCPdu - DtViewControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtTetherVCPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtTetherVCPdu (const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 DtConstVector	topoUpOffset () const
 const DtTaitBryan &	orientation () const
 const DtGlobalObjectDesignator&	attachId () const
 void	setTopoUpOffset (DtConstVector offset)
 void	setOrientation (const DtTaitBryan &ori)
 void	setAttachId (const DtGlobalObjectDesignator& id)
 void	setFromString (const char *str)
 char*	string () const
 void	printData () const

Protected Fields

 DtVector	theVec
 DtTaitBryan	theAngles
 DtGlobalObjectDesignator	theId

Inherited from [DtViewControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- static void **removeCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- const DtGlobalObjectDesignator& **viewerId**() const
- const DtGlobalObjectDesignator& **senderId**() const
- [DtViewerCommand](#) **viewerCommand**() const
- unsigned long **ignoreFlags**() const
- void **setViewerId**(const DtGlobalObjectDesignator& id)
- void **setSenderId**(const DtGlobalObjectDesignator& id)
- void **setIgnoreFlags**(unsigned long flags)

Protected Fields

- DtGlobalObjectDesignator **retId**

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
 - void **setViewerCommand**([DtViewerCommand](#) com)
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const

- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)
- virtual void **insertBytes**(void *address, int numBytes)
- virtual void **deleteBytes**(int offset, int numBytes)
- virtual void **deleteBytes**(void *address, int numBytes)

- virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtTetherVCPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtTetherVCPdu(const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtConstVector topoUpOffset() const
 - const [DtTaitBryan](#)& orientation() const
 - const DtGlobalObjectDesignator& attachId() const
 - void setTopoUpOffset(DtConstVector offset)
 - void setOrientation(const [DtTaitBryan](#) &ori)
 - void setAttachId(const DtGlobalObjectDesignator& id)
 - void setFromString(const char *str)
 - char* string() const
 - void printData() const
 - [DtVector](#) theVec
 - [DtTaitBryan](#) theAngles
 - DtGlobalObjectDesignator theId
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file taitbryan.h:

class DtTaitBryan

Public Methods

	DtTaitBryan ()
	DtTaitBryan (double a, double b, double c)
	DtTaitBryan (const DtTaitBryan &tait)
DtTaitBryan &	operator= (const DtTaitBryan &tait)
int	operator== (const DtTaitBryan & other) const <i>Get whether another tait bryan equals or does not equal this one</i>
int	operator!= (const DtTaitBryan & other) const
const char*	string () const
DtFloat64	psi () const
DtFloat64	theta () const
DtFloat64	phi () const
void	setPsi (DtFloat64 psi)
void	setTheta (DtFloat64 theta)
void	setPhi (DtFloat64 phi)
inline	DtTaitBryan () <i>Inline functions</i>
inline	DtTaitBryan (double a, double b, double c)
inline	DtTaitBryan (const DtTaitBryan &tait)
inline int	operator!= (const DtTaitBryan & other) const
inline DtTaitBryan &	operator= (const DtTaitBryan &tait)
inline int	operator== (const DtTaitBryan & other) const
inline DtFloat64	phi () const
inline DtFloat64	psi () const
inline void	setPhi (DtFloat64 phi)
inline void	setPsi (DtFloat64 psi)
inline void	setTheta (DtFloat64 theta)
inline const char*	string () const
inline DtFloat64	theta () const

Protected Fields

- `DtFloat64 myPsi`
 - `DtFloat64 myTheta`
about reference z
 - `DtFloat64 myPhi`
about intermediate y
 - *about body x*
-

Documentation

- `DtTaitBryan()`
- `DtTaitBryan(double a, double b, double c)`
- `DtTaitBryan(const DtTaitBryan &tait)`
- `DtTaitBryan& operator=(const DtTaitBryan &tait)`
- `int operator==(const DtTaitBryan& other) const`
Get whether another tait bryan equals or does not equal this one
- `int operator!=(const DtTaitBryan& other) const`
- `const char* string() const`
- `DtFloat64 psi() const`
- `DtFloat64 theta() const`
- `DtFloat64 phi() const`
- `void setPsi(DtFloat64 psi)`
- `void setTheta(DtFloat64 theta)`
- `void setPhi(DtFloat64 phi)`
- `DtFloat64 myPsi`
- `DtFloat64 myTheta`
about reference z
- `DtFloat64 myPhi`
about intermediate y
- *about body x*
- `inline DtTaitBryan()`
Inline functions
- `inline DtTaitBryan(double a, double b, double c)`

- inline DtTaitBryan(const [DtTaitBryan](#) &tait)
- inline int operator!=(const [DtTaitBryan](#)& other) const
- inline [DtTaitBryan](#)& operator=(const [DtTaitBryan](#) &tait)
- inline int operator==(const [DtTaitBryan](#)& other) const
- inline DtFloat64 phi() const
- inline DtFloat64 psi() const
- inline void setPhi(DtFloat64 phi)
- inline void setPsi(DtFloat64 psi)
- inline void setTheta(DtFloat64 theta)
- inline const char* string() const
- inline DtFloat64 theta() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vcpdu.h:

class DtTetherTrackVCPdu: public DtViewControlPdu

Inheritance:

DtTetherTrackVCPdu - [DtViewControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtTetherTrackVCPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtTetherTrackVCPdu (const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtConstVector offset () const
	const DtGlobalObjectDesignator& attachId () const
	const DtGlobalObjectDesignator& trackId () const
	void setOffset (DtConstVector loc)
	void setAttachId (const DtGlobalObjectDesignator& id)
	void setTrackId (const DtGlobalObjectDesignator& id)
	void setFromString (const char *str)
	char* string () const
	void printData () const

Protected Fields

	DtVector theVec
	DtGlobalObjectDesignator theId

Inherited from [DtViewControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- static void **removeCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- const DtGlobalObjectDesignator& **viewerId**() const
- const DtGlobalObjectDesignator& **senderId**() const
- [DtViewerCommand](#) **viewerCommand**() const
- unsigned long **ignoreFlags**() const
- void **setViewerId**(const DtGlobalObjectDesignator& id)
- void **setSenderId**(const DtGlobalObjectDesignator& id)
- void **setIgnoreFlags**(unsigned long flags)

Protected Fields

- DtGlobalObjectDesignator **retId**

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
 - void **setViewerCommand**([DtViewerCommand](#) com)
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const

- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)
- virtual void **insertBytes**(void *address, int numBytes)
- virtual void **deleteBytes**(int offset, int numBytes)
- virtual void **deleteBytes**(void *address, int numBytes)

- virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtTetherTrackVCPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtTetherTrackVCPdu(const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtConstVector offset() const
 - const DtGlobalObjectDesignator& attachId() const
 - const DtGlobalObjectDesignator& trackId() const
 - void setOffset(DtConstVector loc)
 - void setAttachId(const DtGlobalObjectDesignator& id)
 - void setTrackId(const DtGlobalObjectDesignator& id)
 - void setFromString(const char *str)
 - char* string() const
 - void printData() const
 - [DtVector](#) theVec
 - DtGlobalObjectDesignator theId
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vcpdu.h:

class DtOrbitVCPdu: public DtViewControlPdu

Inheritance:

DtOrbitVCPdu - DtViewControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtOrbitVCPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtOrbitVCPdu (const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	radius () const
	azimuthRate () const
	heading () const
	pitch () const
	headingOffset () const
	pitchOffset () const
	roll () const
	const DtGlobalObjectDesignator& attachId () const
	setRadius (float r)
	setAzimuthRate (float a)
	setHeading (float h)
	setPitch (float p)
	setHeadingOffset (float o)
	setPitchOffset (float p)
	setRoll (float r)
	setAttachId (const DtGlobalObjectDesignator& id)
	setFromString (const char *str)
	string () const
	printData () const

Protected Fields

• DtGlobalObjectDesignator **theld**

Inherited from [DtViewControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- static void **removeCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- const DtGlobalObjectDesignator& **viewerId**() const
- const DtGlobalObjectDesignator& **senderId**() const
- [DtViewerCommand](#) **viewerCommand**() const
- unsigned long **ignoreFlags**() const
- void **setViewerId**(const DtGlobalObjectDesignator& id)
- void **setSenderId**(const DtGlobalObjectDesignator& id)
- void **setIgnoreFlags**(unsigned long flags)

Protected Fields

• DtGlobalObjectDesignator **retId**

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
 - void **setViewerCommand**([DtViewerCommand](#) com)
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtOrbitVCPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `DtOrbitVCPdu(const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `float radius() const`
 - `float azimuthRate() const`
 - `float heading() const`
 - `float pitch() const`
 - `float headingOffset() const`
 - `float pitchOffset() const`
 - `float roll() const`
 - `const DtGlobalObjectDesignator& attachId() const`
 - `void setRadius(float r)`
 - `void setAzimuthRate(float a)`
 - `void setHeading(float h)`
 - `void setPitch(float p)`
 - `void setHeadingOffset(float o)`
 - `void setPitchOffset(float p)`
 - `void setRoll(float r)`
 - `void setAttachId(const DtGlobalObjectDesignator& id)`
 - `void setFromString(const char *str)`
 - `char* string() const`
 - `void printData() const`
 - `DtGlobalObjectDesignator theId`
-

This class has no child classes.

In file vcpdu.h:

class DtModelessVCPdu: public DtViewControlPdu

Inheritance:

DtModelessVCPdu - DtViewControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtModelessVCPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtModelessVCPdu (const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	const DtGlobalObjectDesignator& primary () const
	const DtGlobalObjectDesignator& secondary () const
	int numEntitiesToRemove () const
	int numEntitiesToAdd () const
	const DtGlobalObjectDesignator& entityToRemove (int n) const
	const DtGlobalObjectDesignator& entityToAdd (int n) const
	void setPrimary (const DtGlobalObjectDesignator& id)
	void setSecondary (const DtGlobalObjectDesignator& id)
	void setNumEntitiesToRemove (int n)
	void setNumEntitiesToAdd (int n)
	void setEntityToRemove (int n, const DtGlobalObjectDesignator& id)
	void setEntityToAdd (int n, const DtGlobalObjectDesignator& id)
	void printData () const

Protected Fields

 DtGlobalObjectDesignator **theid**

Inherited from [DtViewControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- static void **removeCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- const DtGlobalObjectDesignator& **viewerId**() const
- const DtGlobalObjectDesignator& **senderId**() const
- [DtViewerCommand](#) **viewerCommand**() const
- unsigned long **ignoreFlags**() const
- void **setViewerId**(const DtGlobalObjectDesignator& id)
- void **setSenderId**(const DtGlobalObjectDesignator& id)
- void **setIgnoreFlags**(unsigned long flags)
- virtual void **setFromString**(const char *)
- virtual char* **string**() const

Protected Fields

- DtGlobalObjectDesignator **retId**

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
 - void **setViewerCommand**([DtViewerCommand](#) com)
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtModelessVCPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtModelessVCPdu(const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - const DtGlobalObjectDesignator& primary() const
 - const DtGlobalObjectDesignator& secondary() const
 - int numEntitiesToRemove() const
 - int numEntitiesToAdd() const
 - const DtGlobalObjectDesignator& entityToRemove(int n) const
 - const DtGlobalObjectDesignator& entityToAdd(int n) const
 - void setPrimary(const DtGlobalObjectDesignator& id)
 - void setSecondary(const DtGlobalObjectDesignator& id)
 - void setNumEntitiesToRemove(int n)
 - void setNumEntitiesToAdd(int n)
 - void setEntityToRemove(int n, const DtGlobalObjectDesignator& id)
 - void setEntityToAdd(int n, const DtGlobalObjectDesignator& id)
 - void printData() const
 - DtGlobalObjectDesignator theId
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vcpdu.h:

class DtMimicVCPdu: public DtViewControlPdu

Inheritance:

DtMimicVCPdu - DtViewControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtMimicVCPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtMimicVCPdu (const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 DtConstVector	offset () const
 const DtTaitBryan &	bodyOrientation () const
 const DtGlobalObjectDesignator&	attachId () const
 void	setOffset (DtConstVector offset)
 void	setBodyOrientation (const DtTaitBryan &ori)
 void	setAttachId (const DtGlobalObjectDesignator& id)
 void	setFromString (const char *str)
 char*	string () const
 void	printData () const

Protected Fields

 DtGlobalObjectDesignator	theId
 DtVector	theVec
 DtTaitBryan	theAngles

Inherited from [DtViewControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- static void **removeCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- const DtGlobalObjectDesignator& **viewerId**() const
- const DtGlobalObjectDesignator& **senderId**() const
- [DtViewerCommand](#) **viewerCommand**() const
- unsigned long **ignoreFlags**() const
- void **setViewerId**(const DtGlobalObjectDesignator& id)
- void **setSenderId**(const DtGlobalObjectDesignator& id)
- void **setIgnoreFlags**(unsigned long flags)

Protected Fields

- DtGlobalObjectDesignator **retId**

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
 - void **setViewerCommand**([DtViewerCommand](#) com)
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const

- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)
- virtual void **insertBytes**(void *address, int numBytes)
- virtual void **deleteBytes**(int offset, int numBytes)
- virtual void **deleteBytes**(void *address, int numBytes)

- virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtMimicVCPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtMimicVCPdu(const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtConstVector offset() const
 - const [DtTaitBryan](#)& bodyOrientation() const
 - const DtGlobalObjectDesignator& attachId() const
 - void setOffset(DtConstVector offset)
 - void setBodyOrientation(const [DtTaitBryan](#) &ori)
 - void setAttachId(const DtGlobalObjectDesignator& id)
 - void setFromString(const char *str)
 - char* string() const
 - void printData() const
 - DtGlobalObjectDesignator theId
 - [DtVector](#) theVec
 - [DtTaitBryan](#) theAngles
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vcpdu.h:

class DtMimicTrackVCPdu: public DtViewControlPdu

Inheritance:

DtMimicTrackVCPdu - [DtViewControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtMimicTrackVCPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtMimicTrackVCPdu (const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	offset () const
	const DtGlobalObjectDesignator& attachId () const
	const DtGlobalObjectDesignator& trackId () const
	void setOffset (DtConstVector loc)
	void setAttachId (const DtGlobalObjectDesignator& id)
	void setTrackId (const DtGlobalObjectDesignator& id)
	void setFromString (const char *str)
	char* string () const
	void printData () const

Protected Fields

	DtVector	theVec
	DtGlobalObjectDesignator	theId

Inherited from [DtViewControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- static void **removeCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- const DtGlobalObjectDesignator& **viewerId**() const
- const DtGlobalObjectDesignator& **senderId**() const
- [DtViewerCommand](#) **viewerCommand**() const
- unsigned long **ignoreFlags**() const
- void **setViewerId**(const DtGlobalObjectDesignator& id)
- void **setSenderId**(const DtGlobalObjectDesignator& id)
- void **setIgnoreFlags**(unsigned long flags)

Protected Fields

- DtGlobalObjectDesignator **retId**

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
 - void **setViewerCommand**([DtViewerCommand](#) com)
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const

- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)
- virtual void **insertBytes**(void *address, int numBytes)
- virtual void **deleteBytes**(int offset, int numBytes)
- virtual void **deleteBytes**(void *address, int numBytes)

- virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtMimicTrackVCPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtMimicTrackVCPdu(const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtConstVector offset() const
 - const DtGlobalObjectDesignator& attachId() const
 - const DtGlobalObjectDesignator& trackId() const
 - void setOffset(DtConstVector loc)
 - void setAttachId(const DtGlobalObjectDesignator& id)
 - void setTrackId(const DtGlobalObjectDesignator& id)
 - void setFromString(const char *str)
 - char* string() const
 - void printData() const
 - [DtVector](#) theVec
 - DtGlobalObjectDesignator theId
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vcpdu.h:

class DtGroupVCPdu: public DtViewControlPdu

Inheritance:

DtGroupVCPdu - DtViewControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtGroupVCPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtGroupVCPdu (const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	radialOffset () const
	heading () const
	pitch () const
	headingOffset () const
	pitchOffset () const
	groupCmd () const
	numEntitiesToRemove () const
	numEntitiesToAdd () const
	entityToRemove (int n) const
	entityToAdd (int n) const
	setRadialOffset (float o)
	setHeading (float h)
	setPitch (float p)
	setHeadingOffset (float o)
	setPitchOffset (float o)
	setGroupCmd (int c)
	setNumEntitiesToRemove (int n)
	setNumEntitiesToAdd (int n)
	setEntityToRemove (int n, const DtGlobalObjectDesignator& id)

void	setEntityToAdd (int n, const DtGlobalObjectDesignator& id)
void	setFromString (const char *str)
char*	string () const
void	printData () const

Protected Fields

DtGlobalObjectDesignator **theid**

Inherited from [DtViewControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- static void **removeCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- const DtGlobalObjectDesignator& **viewerId**() const
- const DtGlobalObjectDesignator& **senderId**() const
- [DtViewerCommand](#) **viewerCommand**() const
- unsigned long **ignoreFlags**() const
- void **setViewerId**(const DtGlobalObjectDesignator& id)
- void **setSenderId**(const DtGlobalObjectDesignator& id)
- void **setIgnoreFlags**(unsigned long flags)

Protected Fields

DtGlobalObjectDesignator **retId**

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setViewerCommand**([DtViewerCommand](#) com)

Inherited from [DtInteraction](#):

Public Methods

• [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)

- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtGroupVCPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtGroupVCPdu(const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- float radialOffset() const
- float heading() const
- float pitch() const
- float headingOffset() const
- float pitchOffset() const
- int groupCmd() const
- int numEntitiesToRemove() const
- int numEntitiesToAdd() const
- const DtGlobalObjectDesignator& entityToRemove(int n) const
- const DtGlobalObjectDesignator& entityToAdd(int n) const
- void setRadialOffset(float o)
- void setHeading(float h)
- void setPitch(float p)
- void setHeadingOffset(float o)
- void setPitchOffset(float o)
- void setGroupCmd(int c)

- `void setNumEntitiesToRemove(int n)`
- `void setNumEntitiesToAdd(int n)`
- `void setEntityToRemove(int n, const DtGlobalObjectDesignator& id)`
- `void setEntityToAdd(int n, const DtGlobalObjectDesignator& id)`
- `void setFromString(const char *str)`
- `char* string() const`
- `void printData() const`
- `DtGlobalObjectDesignator theId`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vcpdu.h:

class DtCompassVCPdu: public DtViewControlPdu

Inheritance:

DtCompassVCPdu - DtViewControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtCompassVCPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtCompassVCPdu (const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	radius () const
	bodyHeading () const
	pitch () const
	headingOffset () const
	pitchOffset () const
	roll () const
	const DtGlobalObjectDesignator& attachId () const
	setRadius (float r)
	setBodyHeading (float h)
	setPitch (float p)
	setHeadingOffset (float o)
	setPitchOffset (float p)
	setRoll (float r)
	setAttachId (const DtGlobalObjectDesignator& id)
	setFromString (const char *str)
	string () const
	printData () const

Protected Fields

 DtGlobalObjectDesignator **theid**

Inherited from [DtViewControlPdu](#):

Public Methods

-  static [DtPdu](#)* **create**(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  static void **addCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
-  static void **removeCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
-  const DtGlobalObjectDesignator& **viewerId**() const
-  const DtGlobalObjectDesignator& **senderId**() const
-  [DtViewerCommand](#) **viewerCommand**() const
-  unsigned long **ignoreFlags**() const
-  void **setViewerId**(const DtGlobalObjectDesignator& id)
-  void **setSenderId**(const DtGlobalObjectDesignator& id)
-  void **setIgnoreFlags**(unsigned long flags)

Protected Fields

 DtGlobalObjectDesignator **retId**

Protected Methods

-  [DtPduKind](#) **internalGetPduKind**() const
 -  void **setViewerCommand**([DtViewerCommand](#) com)
-

Inherited from [DtInteraction](#):

Public Methods

-  [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtCompassVCPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `DtCompassVCPdu(const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `float radius() const`
 - `float bodyHeading() const`
 - `float pitch() const`
 - `float headingOffset() const`
 - `float pitchOffset() const`
 - `float roll() const`
 - `const DtGlobalObjectDesignator& attachId() const`
 - `void setRadius(float r)`
 - `void setBodyHeading(float h)`
 - `void setPitch(float p)`
 - `void setHeadingOffset(float o)`
 - `void setPitchOffset(float p)`
 - `void setRoll(float r)`
 - `void setAttachId(const DtGlobalObjectDesignator& id)`
 - `void setFromString(const char *str)`
 - `char* string() const`
 - `void printData() const`
 - `DtGlobalObjectDesignator theId`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vcpdu.h:

class DtAbsoluteVCPdu: public DtViewControlPdu

Inheritance:

DtAbsoluteVCPdu - DtViewControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtAbsoluteVCPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtAbsoluteVCPdu** (const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtConstVector **worldLocation** () const
- const [DtTaitBryan](#)& **orientation** () const
- void **setWorldLocation** (DtConstVector loc)
- void **setOrientation** (const [DtTaitBryan](#) &ori)
- void **setFromString** (const char *str)
- char* **string** () const
- void **printData** () const

Protected Fields

- [DtVector](#) theVec
 - [DtTaitBryan](#) theAngles
-

Inherited from DtViewControlPdu:

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)

- static void **removeCallback**([DtExerciseConn](#) *conn, DtViewControlInteractionCb cb, void *usr)
- const DtGlobalObjectDesignator& **viewerId**() const
- const DtGlobalObjectDesignator& **senderId**() const
- [DtViewerCommand](#) **viewerCommand**() const
- unsigned long **ignoreFlags**() const
- void **setViewerId**(const DtGlobalObjectDesignator& id)
- void **setSenderId**(const DtGlobalObjectDesignator& id)
- void **setIgnoreFlags**(unsigned long flags)

Protected Fields

- DtGlobalObjectDesignator **retId**

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
 - void **setViewerCommand**([DtViewerCommand](#) com)
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const

- virtual [DtProtocolFamily](#) **protocolFamily()** const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp()** const
- virtual [DtTimeStampType](#) **timeStampType()** const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader()** const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)

- virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtAbsoluteVCPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtAbsoluteVCPdu(const DtNetViewControlPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

- `DtConstVector worldLocation() const`
- `const DtTaitBryan& orientation() const`
- `void setWorldLocation(DtConstVector loc)`
- `void setOrientation(const DtTaitBryan &ori)`
- `void setFromString(const char *str)`
- `char* string() const`
- `void printData() const`
- `DtVector theVec`
- `DtTaitBryan theAngles`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file signalpdu.h:

class DtSignalPdu: public DtInteraction

Inheritance:

DtSignalPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtSignalPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtSignalPdu (const DtNetSignalPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static void	addCallback (DtExerciseConn *conn, DtSignalInteractionCb cb, void *usr)
static void	removeCallback (DtExerciseConn *conn, DtSignalInteractionCb cb, void *usr)
const DtEntityIdentifier &	entityId () const
unsigned int	radioid () const
DtEncodingClass	encodingClass () const
unsigned int	encodingType () const
long	sampleRate () const
int	numDataBits () const
int	numDataBytes () const
int	numSamples () const
unsigned char	dataByte (int byteNum) const
void	getData (unsigned char *buff) const
int	tdlType () const
void	setEntityId (const DtEntityIdentifier &id)
void	setRadioid (unsigned int id)
void	setEncodingClass (DtEncodingClass encodingClass)
void	setEncodingType (unsigned int codeType)
void	setSampleRate (long rate)
void	setNumDataBits (int lengthInBits)

void	setNumDataBytes (int lengthInBytes)
void	setNumSamples (int numSamples)
void	setDataByte (int byteNum, unsigned char data)
void	setData (unsigned char *data, int numBytes)
void	setTdlType (int type)
void	printData () const

Protected Fields

[DtEntityIdentifier](#) theId

Protected Methods

[DtPduKind](#) internalGetPduKind () const

Inherited from [DtInteraction](#):

Public Methods

[DtInteraction](#)& operator=(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const

- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtSignalPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtSignalPdu(const DtNetSignalPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void addCallback([DtExerciseConn](#) *conn, DtSignalInteractionCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtSignalInteractionCb cb, void *usr)

- `const DtEntityIdentifier& entityId() const`
- `unsigned int radioId() const`
- `DtEncodingClass encodingClass() const`
- `unsigned int encodingType() const`
- `long sampleRate() const`
- `int numDataBits() const`
- `int numDataBytes() const`
- `int numSamples() const`
- `unsigned char dataByte(int byteNum) const`
- `void getData(unsigned char *buff) const`
- `int tdlType() const`
- `void setEntityId(const DtEntityIdentifier &id)`
- `void setRadioId(unsigned int id)`
- `void setEncodingClass(DtEncodingClass encodingClass)`
- `void setEncodingType(unsigned int codeType)`
- `void setSampleRate(long rate)`
- `void setNumDataBits(int lengthInBits)`
- `void setNumDataBytes(int lengthInBytes)`
- `void setNumSamples(int numSamples)`
- `void setDataByte(int byteNum, unsigned char data)`
- `void setData(unsigned char *data, int numBytes)`
- `void setTdlType(int type)`
- `void printData() const`
- `DtPduKind internalGetPduKind() const`
- `DtEntityIdentifier theId`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtEncodingClass

-  DtSignalClassEncodedVoice
 -  DtSignalClassRawBinary
 -  DtSignalClassApplicationSpecific
 -  DtSignalClassPreRecorded
-

Documentation

-  DtSignalClassEncodedVoice
-  DtSignalClassRawBinary
-  DtSignalClassApplicationSpecific
-  DtSignalClassPreRecorded

In file lcpdu.h:

class DtLgrControlPdu: public DtInteraction

class DtLgrControlPdu: Instances of DtLgrControlPdu are a family of messgae used to control a remote MaK-Logger over the network

Inheritance:

DtLgrControlPdu - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtLgrControlPdu () <i>default constructor</i>
 static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a PDU representation from a network packet</i>
 static DtLgrControlPdu *	create (DtLgrCommand command) <i>Create a PDU from DtLgrCommand;</i>
 static void	addCallback (DtExerciseConn * conn, DtLgrControlInteractionCb cb, void* usr) <i>DIS -- install/remove handler to call when receiving a LC packet</i>
 static void	removeCallback (DtExerciseConn * conn, DtLgrControlInteractionCb cb, void* usr)
 virtual	~DtLgrControlPdu () <i>destructor</i>
	DtLgrControlPdu (const DtLgrControlPdu & orig) <i>copy constructor</i>
 DtLgrControlPdu &	operator= (const DtLgrControlPdu & orig) <i>assignment operator</i>
 virtual void	setReceiverId (const DtGlobalObjectDesignator& id) <i>set/get Object Identifier of intended recipient (logger)</i>
 const DtGlobalObjectDesignator&	receiverId () const
 virtual void	setSenderId (const DtGlobalObjectDesignator& id) <i>set/get Object Identifier of sender, if applicable</i>
 virtual const DtGlobalObjectDesignator&	senderId () const
 DtLgrCommand	lgrCommand () const <i>get message-type for proper casting to a subclass</i>
 virtual void	printData () const <i>print contents</i>
 inline DtPduKind	internalGetPduKind () const

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
identity for use by toolkit
- void **setLgrCommand** ([DtLgrCommand](#) com)
set is done by way of constructing the proper subclass

Private Fields

- DtGlobalObjectDesignator **mySenderId**
parameter
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const

- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrControlPdu:

Instances of DtLgrControlPdu are a family of messgae used to control a remote MaK-Logger over the network. See lc*.h

- DtLgrControlPdu()**
default constructor
- static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)**
Create a PDU representation from a network packet
- static [DtLgrControlPdu](#)* create([DtLgrCommand](#) command)**
Create a PDU from DtLgrCommand;
- static void addCallback([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)**
DIS -- install/remove handler to call when receiving a LC packet
- static void removeCallback([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)**
- virtual ~DtLgrControlPdu()**

destructor

`DtLgrControlPdu(const DtLgrControlPdu& orig)`

copy constructor

`DtLgrControlPdu& operator=(const DtLgrControlPdu& orig)`

assignment operator

`virtual void setReceiverId(const DtGlobalObjectDesignator& id)`

set/get Object Identifier of intended recipient (logger)

`const DtGlobalObjectDesignator& receiverId() const`

`virtual void setSenderId(const DtGlobalObjectDesignator& id)`

set/get Object Identifier of sender, if applicable

`virtual const DtGlobalObjectDesignator& senderId() const`

`DtLgrCommand lgrCommand() const`

get message-type for proper casting to a subclass

`virtual void printData() const`

print contents

`DtPduKind internalGetPduKind() const`

identity for use by toolkit

`void setLgrCommand(DtLgrCommand com)`

set is done by way of constructing the proper subclass

`DtGlobalObjectDesignator mySenderId`

parameter

`DtGlobalObjectDesignator myReceiverId`

`inline DtPduKind internalGetPduKind() const`

Direct child classes:

[DtLgrTimeoutPdu](#)

[DtLgrTimeScalePdu](#)

[DtLgrStopPdu](#)

[DtLgrStartPdu](#)

[DtLgrSkipTimePdu](#)

[DtLgrSetTimePdu](#)

[DtLgrRecordPdu](#)

[DtLgrRecordActionPdu](#)

[DtLgrRecFilePdu](#)

[DtLgrQuitPdu](#)

[DtLgrPlayPdu](#)

[DtLgrPlayActionPdu](#)

[DtLgrPbFilePdu](#)

[DtLgrPausePdu](#)

[DtLgrHeartbeatPdu](#)

[DtLgrEofActionPdu](#)

[DtLgrEndPdu](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtLgrCommand

-  DtLgrPbFile
 -  DtLgrRecFile
 -  DtLgrQuit
 -  DtLgrStart
 -  DtLgrEnd
 -  DtLgrPlay
 -  DtLgrPause
 -  DtLgrStop
 -  DtLgrRecord
 -  DtLgrSetTime
 -  DtLgrSkipTime
 -  DtLgrTimeScale
 -  DtLgrPlayAction
 -  DtLgrRecordAction
 -  DtLgrEofAction
 -  DtLgrHeartbeat
 -  DtLgrTimeout
-

Documentation

-  DtLgrPbFile
-  DtLgrRecFile
-  DtLgrQuit
-  DtLgrStart
-  DtLgrEnd
-  DtLgrPlay

- DtLgrPause
- DtLgrStop
- DtLgrRecord
- DtLgrSetTime
- DtLgrSkipTime
- DtLgrTimeScale
- DtLgrPlayAction
- DtLgrRecordAction
- DtLgrEofAction
- DtLgrHeartbeat
- DtLgrTimeout

[Alphabetic index](#) [Hierarchy of classes](#)

In file lctimeout.h:

class DtLgrTimeoutPdu: public DtLgrControlPdu

class DtLgrTimeoutPdu: Instances of DtLgrTimeoutPdu are to tell remote logger how long to generate interim packets for entity before assuming no more packets in file

Inheritance:

DtLgrTimeoutPdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrTimeoutPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
- **DtLgrTimeoutPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
- **DtLgrTimeoutPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtReal tmout, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
- virtual **~DtLgrTimeoutPdu** ()
destructor
- **DtLgrTimeoutPdu** (const [DtLgrTimeoutPdu](#)& orig)
copy constructor
- [DtLgrTimeoutPdu](#)& **operator=** (const [DtLgrTimeoutPdu](#)& orig)
assignment operator
- virtual void **setTimeout** (DtReal val)
set/get timeout
- virtual DtReal **timeout** ()
- virtual DtLgrRealParams* **params** ()
return correctly casted parameters pointer; see lgrControlP

Private Fields

- DtReal **myTimeout**
parameter
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)

- virtual int **protocolVersion**() const
- virtual void **setExerciseId**(int id)
- virtual int **exerciseId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseId**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const

- virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrTimeoutPdu:

Instances of DtLgrTimeoutPdu are to tell remote logger how long to generate interim packets for entity before assuming no more packets in file. Useful during slow-motion playback.

- `DtLgrTimeoutPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
default constructor
- `DtLgrTimeoutPdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct from network representation
- `DtLgrTimeoutPdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtReal tmout, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct and prepare for send
- `virtual ~DtLgrTimeoutPdu()`
destructor
- `DtLgrTimeoutPdu(const DtLgrTimeoutPdu& orig)`
copy constructor
- `DtLgrTimeoutPdu& operator=(const DtLgrTimeoutPdu& orig)`
assignment operator
- `virtual void setTimeout( DtReal val)`
set/get timeout. see class description
- `virtual DtReal timeout()`
- `virtual DtLgrRealParams* params()`
return correctly casted parameters pointer; see lgrControlP
- `DtReal myTimeout`
parameter

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lctimescale.h:

class DtLgrTimeScalePdu: public DtLgrControlPdu

class DtLgrTimeScalePdu: Instances of DtLgrTimeScalePdu are to tell remote logger to slow down or speed up playback

Inheritance:

DtLgrTimeScalePdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrTimeScalePdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
- **DtLgrTimeScalePdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
- **DtLgrTimeScalePdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, double scale, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
- virtual **~DtLgrTimeScalePdu** ()
destructor
- **DtLgrTimeScalePdu** (const [DtLgrTimeScalePdu](#)& orig)
copy constructor
- [DtLgrTimeScalePdu](#)& **operator=** (const [DtLgrTimeScalePdu](#)& orig)
assignment operator
- virtual void **setTimeScale** (DtReal time)
set/get playback speed
- virtual double **timeScale** ()
- virtual DtLgrRealParams* **params** ()
return correctly casted parameters pointer; see lgrControlP

Private Fields

- DtReal **myScale**
parameter
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)

- virtual int **protocolVersion**() const
- virtual void **setExerciseId**(int id)
- virtual int **exerciseId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseId**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const

- virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrTimeScalePdu:

Instances of DtLgrTimeScalePdu are to tell remote logger to slow down or speed up playback. 1.0 is normal speed

- `DtLgrTimeScalePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
default constructor
- `DtLgrTimeScalePdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct from network representation
- `DtLgrTimeScalePdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, double scale, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct and prepare for send
- `virtual ~DtLgrTimeScalePdu()`
destructor
- `DtLgrTimeScalePdu(const DtLgrTimeScalePdu& orig)`
copy constructor
- `DtLgrTimeScalePdu& operator=(const DtLgrTimeScalePdu& orig)`
assignment operator
- `virtual void setTimeScale( DtReal time)`
set/get playback speed
- `virtual double timeScale()`
- `virtual DtLgrRealParams* params()`
return correctly casted parameters pointer; see lgrControlP
- `DtReal myScale`
parameter

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcstop.h:

class DtLgrStopPdu: public DtLgrControlPdu

class DtLgrStopPdu: Instances of DtLgrStopPdu are to tell remote logger to stop

Inheritance:

DtLgrStopPdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrStopPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
 - **DtLgrStopPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
 - **DtLgrStopPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
 - virtual **~DtLgrStopPdu** ()
destructor
 - **DtLgrStopPdu** (const [DtLgrStopPdu](#)& orig)
copy constructor
 - [DtLgrStopPdu](#)& **operator=** (const [DtLgrStopPdu](#)& orig)
assignment operator
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const

- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const

- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrStopPdu: Instances of DtLgrStopPdu are to tell remote logger to stop

- DtLgrStopPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor

- DtLgrStopPdu**(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation

- DtLgrStopPdu**(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

construct and prepare for send

- `virtual ~DtLgrStopPdu()`

destructor

- `DtLgrStopPdu(const DtLgrStopPdu& orig)`

copy constructor

- `DtLgrStopPdu& operator=(const DtLgrStopPdu& orig)`

assignment operator

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcstart.h:

class DtLgrStartPdu: public DtLgrControlPdu

class DtLgrStartPdu: Instances of DtLgrStartPdu are used to tell remote logger to start recording

Inheritance:

DtLgrStartPdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrStartPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
 - **DtLgrStartPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
 - **DtLgrStartPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
 - virtual **~DtLgrStartPdu** ()
destructor
 - **DtLgrStartPdu** (const [DtLgrStartPdu](#)& orig)
copy constructor
 - [DtLgrStartPdu](#)& **operator=** (const [DtLgrStartPdu](#)& orig)
assignment operator
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)

- `const DtGlobalObjectDesignator& receiverId() const`
- `virtual void setSenderId(const DtGlobalObjectDesignator& id)`
- `virtual const DtGlobalObjectDesignator& senderId() const`
- `DtLgrCommand lgrCommand() const`
- `virtual void printData() const`
- `inline DtPduKind internalGetPduKind() const`

Protected Methods

- `DtPduKind internalGetPduKind() const`
- `void setLgrCommand(DtLgrCommand com)`

Private Fields

- `DtGlobalObjectDesignator mySenderId`
 - `DtGlobalObjectDesignator myReceiverId`
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- `virtual void* packet() const`
- `virtual int status() const`
- `virtual void setVersion(int vers)`
- `virtual int protocolVersion() const`
- `virtual void setExercisId(int id)`
- `virtual int exercisId() const`
- `virtual DtPduKind kind() const`
- `virtual DtProtocolFamily protocolFamily() const`
- `virtual void setTimeStamp(DtTime time, DtTimeStampType type = DtTimeStampRelative)`
- `virtual DtTime timeStamp() const`
- `virtual DtTimeStampType timeStampType() const`

- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrStartPdu: Instances of DtLgrStartPdu are used to tell remote logger to start recording

- DtLgrStartPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)**
default constructor

- DtLgrStartPdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)**
construct from network representation

- DtLgrStartPdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)**

construct and prepare for send

- `virtual ~DtLgrStartPdu()`

destructor

- `DtLgrStartPdu(const DtLgrStartPdu& orig)`

copy constructor

- `DtLgrStartPdu& operator=(const DtLgrStartPdu& orig)`

assignment operator

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcskiptime.h:

class DtLgrSkipTimePdu: public DtLgrControlPdu

class DtLgrSkipTimePdu: Instances of DtLgrSetTimePdu are to tell remote logger to forward or backward
HH:MM:SS

Inheritance:

DtLgrSkipTimePdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrSkipTimePdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
- **DtLgrSkipTimePdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
- **DtLgrSkipTimePdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtTime tm, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
- virtual **~DtLgrSkipTimePdu** ()
destructor
- **DtLgrSkipTimePdu** (const [DtLgrSkipTimePdu](#)& orig)
copy constructor
- [DtLgrSkipTimePdu](#)& **operator=** (const [DtLgrSkipTimePdu](#)& orig)
assignment operator
- virtual void **setTime** (DtTime dt)
set/get amount of time to skip forward or back in playback file
- virtual DtTime **time** ()
- virtual DtLgrTimeParams* **params** ()
return correctly casted parameters pointer; see lgrControlIP

Private Fields

- DtTime **myTime**
parameter
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)

- virtual int **protocolVersion**() const
- virtual void **setExerciseId**(int id)
- virtual int **exerciseId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseId**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const

- virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrSkipTimePdu: Instances of DtLgrSetTimePdu are to tell remote logger to forward or backward HH:MM:SS

- `DtLgrSkipTimePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
default constructor
- `DtLgrSkipTimePdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct from network representation
- `DtLgrSkipTimePdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtTime tm, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct and prepare for send
- `virtual ~DtLgrSkipTimePdu()`
destructor
- `DtLgrSkipTimePdu(const DtLgrSkipTimePdu& orig)`
copy constructor
- `DtLgrSkipTimePdu& operator=(const DtLgrSkipTimePdu& orig)`
assignment operator
- `virtual void setTime(DtTime dt)`
set/get amount of time to skip forward or back in playback file
- `virtual DtTime time()`
- `virtual DtLgrTimeParams* params()`
return correctly casted parameters pointer; see lgrControlP
- `DtTime myTime`
parameter

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcsettime.h:

class DtLgrSetTimePdu: public DtLgrControlPdu

class DtLgrSetTimePdu: Instances of DtLgrSetTimePdu are to tell remote logger to go to set position to HH:MM:SS

Inheritance:

DtLgrSetTimePdu - DtLgrControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtLgrSetTimePdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
- **DtLgrSetTimePdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
- **DtLgrSetTimePdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtTime tm, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
- virtual **~DtLgrSetTimePdu** ()
destructor
- **DtLgrSetTimePdu** (const DtLgrSetTimePdu& orig)
copy constructor
- DtLgrSetTimePdu& **operator=** (const DtLgrSetTimePdu& orig)
assignment operator
- virtual void **setTime** (const DtTime time)
set/get time to jump to within logger playback file
- virtual DtTime **time** ()
- virtual DtLgrTimeParams* **params** ()
return correctly casted parameters pointer; see lgrControlIP

Private Fields

- DtTime **myTime**
parameter
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)

- virtual int **protocolVersion**() const
- virtual void **setExerciseId**(int id)
- virtual int **exerciseId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseId**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const

- virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrSetTimePdu: Instances of DtLgrSetTimePdu are to tell remote logger to go to set position to HH:MM:SS

- `DtLgrSetTimePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
default constructor
- `DtLgrSetTimePdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct from network representation
- `DtLgrSetTimePdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtTime tm, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct and prepare for send
- `virtual ~DtLgrSetTimePdu()`
destructor
- `DtLgrSetTimePdu(const DtLgrSetTimePdu& orig)`
copy constructor
- `DtLgrSetTimePdu& operator=(const DtLgrSetTimePdu& orig)`
assignment operator
- `virtual void setTime(const DtTime time)`
set/get time to jump to within logger playback file
- `virtual DtTime time()`
- `virtual DtLgrTimeParams* params()`
return correctly casted parameters pointer; see lgrControlP
- `DtTime myTime`
parameter

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcrecord.h:

class DtLgrRecordPdu: public DtLgrControlPdu

class DtLgrRecordPdu: Instances of DtLgrRecordPdu are to tell remote logger to begin recording

Inheritance:

DtLgrRecordPdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrRecordPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
 - **DtLgrRecordPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
 - **DtLgrRecordPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
 - virtual **~DtLgrRecordPdu** ()
destructor
 - **DtLgrRecordPdu** (const [DtLgrRecordPdu](#)& orig)
copy constructor
 - [DtLgrRecordPdu](#)& **operator=** (const [DtLgrRecordPdu](#)& orig)
assignment operator
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)

- `const DtGlobalObjectDesignator& receiverId() const`
- `virtual void setSenderId(const DtGlobalObjectDesignator& id)`
- `virtual const DtGlobalObjectDesignator& senderId() const`
- `DtLgrCommand lgrCommand() const`
- `virtual void printData() const`
- `inline DtPduKind internalGetPduKind() const`

Protected Methods

- `DtPduKind internalGetPduKind() const`
- `void setLgrCommand(DtLgrCommand com)`

Private Fields

- `DtGlobalObjectDesignator mySenderId`
 - `DtGlobalObjectDesignator myReceiverId`
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- `virtual void* packet() const`
- `virtual int status() const`
- `virtual void setVersion(int vers)`
- `virtual int protocolVersion() const`
- `virtual void setExercisId(int id)`
- `virtual int exercisId() const`
- `virtual DtPduKind kind() const`
- `virtual DtProtocolFamily protocolFamily() const`
- `virtual void setTimeStamp(DtTime time, DtTimeStampType type = DtTimeStampRelative)`
- `virtual DtTime timeStamp() const`
- `virtual DtTimeStampType timeStampType() const`

- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrRecordPdu: Instances of DtLgrRecordPdu are to tell remote logger to begin recording

- DtLgrRecordPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor

- DtLgrRecordPdu**(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation

- DtLgrRecordPdu**(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

construct and prepare for send

- `virtual ~DtLgrRecordPdu()`

destructor

- `DtLgrRecordPdu(const DtLgrRecordPdu& orig)`

copy constructor

- `DtLgrRecordPdu& operator=(const DtLgrRecordPdu& orig)`

assignment operator

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcreaction.h:

class DtLgrRecordActionPdu: public DtLgrControlPdu

class DtLgrRecordActionPdu: Instances of DtLgrRecordActionPdu are to tell remote logger to stop or begin recording when entering record-mode

Inheritance:

DtLgrRecordActionPdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrRecordActionPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
- **DtLgrRecordActionPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
- **DtLgrRecordActionPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, [DtLgrMode](#) mode, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
- virtual **~DtLgrRecordActionPdu** ()
destructor
- **DtLgrRecordActionPdu** (const [DtLgrRecordActionPdu](#)& orig)
copy constructor
- [DtLgrRecordActionPdu](#)& **operator=** (const [DtLgrRecordActionPdu](#)& orig)
assignment operator
- virtual void **setAction** ([DtLgrMode](#) cmd)
set/get action to take when switching to record-mode
- virtual [DtLgrMode](#) **action** ()
- virtual DtLgrIntParams* **params** ()
return correctly casted parameters pointer; see lgrControlIP

Private Fields

- [DtLgrMode](#) **myAction**
parameter
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)

- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void*

- virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrRecordActionPdu:

Instances of DtLgrRecordActionPdu are to tell remote logger to stop or begin recording when entering record-mode.

- `DtLgrRecordActionPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
default constructor
- `DtLgrRecordActionPdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct from network representation
- `DtLgrRecordActionPdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtLgrMode mode, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct and prepare for send
- `virtual ~DtLgrRecordActionPdu()`
destructor
- `DtLgrRecordActionPdu(const DtLgrRecordActionPdu& orig)`
copy constructor
- `DtLgrRecordActionPdu& operator=(const DtLgrRecordActionPdu& orig)`
assignment operator
- `virtual void setAction( DtLgrMode cmd )`
set/get action to take when switching to record-mode
- `virtual DtLgrMode action()`
- `virtual DtLgrIntParams* params()`
return correctly casted parameters pointer; see lgrControlP
- `DtLgrMode myAction`
parameter

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcpdu.h:

enum DtLgrMode

General operation modes

-  DtLGR_PLAY
 -  DtLGR_PAUSE
 -  DtLGR_RECORD
 -  DtLGR_STOP
 -  DtLGR_EMPTY
 -  DtLGR_LOOP
-

Documentation

General operation modes

-  DtLGR_PLAY
-  DtLGR_PAUSE
-  DtLGR_RECORD
-  DtLGR_STOP
-  DtLGR_EMPTY
-  DtLGR_LOOP

In file lcrecfile.h:

class DtLgrRecFilePdu: public DtLgrControlPdu

class DtLgrRecFilePdu: Instances of DtLgrRecFilePdu are used to switching remote logger to recording-mode and to set the file into which to store captured simulation traffic

Inheritance:

DtLgrRecFilePdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtLgrRecFilePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>default constructor</i>
	DtLgrRecFilePdu (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>construct from network representation</i>
	DtLgrRecFilePdu (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, char* filename, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>construct and prepare for send</i>
 virtual	~DtLgrRecFilePdu () <i>destructor</i>
	DtLgrRecFilePdu (const DtLgrRecFilePdu & orig) <i>copy constructor</i>
 DtLgrRecFilePdu &	operator= (const DtLgrRecFilePdu & orig) <i>assignment operator</i>
 virtual void	setFilename (const char* path) <i>set/get the name of file into which traffic will be recorded</i>
 virtual const char*	filename ()
 virtual DtLgrStringParams*	params () <i>return correctly casted parameters pointer; see lgrControlP</i>
 virtual void	printData () const <i>diagnostic output</i>

Private Fields

	char* myFilename <i>parameter</i>
---	---

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const

- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)

- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrRecFilePdu: Instances of DtLgrRecFilePdu are used to switching remote logger to recording-mode and to set the file into which to store captured simulation traffic

- `DtLgrRecFilePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
default constructor
- `DtLgrRecFilePdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct from network representation
- `DtLgrRecFilePdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, char* filename, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct and prepare for send
- `virtual ~DtLgrRecFilePdu()`
destructor
- `DtLgrRecFilePdu(const DtLgrRecFilePdu& orig)`
copy constructor
- `DtLgrRecFilePdu& operator=(const DtLgrRecFilePdu& orig)`
assignment operator
- `virtual void setFilename(const char* path)`
set/get the name of file into which traffic will be recorded
- `virtual const char* filename()`
- `virtual DtLgrStringParams* params()`
return correctly casted parameters pointer; see lgrControlP
- `virtual void printData() const`
diagnostic output
- `char* myFilename`
parameter

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcquit.h:

class DtLgrQuitPdu: public DtLgrControlPdu

class DtLgrQuitPdu: Instances of DtLgrQuitPdu are messages telling a remote logger to quit

Inheritance:

DtLgrQuitPdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrQuitPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
 - **DtLgrQuitPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
 - **DtLgrQuitPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
 - virtual **~DtLgrQuitPdu** ()
destructor
 - **DtLgrQuitPdu** (const [DtLgrQuitPdu](#)& orig)
copy constructor
 - [DtLgrQuitPdu](#)& **operator=** (const [DtLgrQuitPdu](#)& orig)
assignment operator
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)

- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const

- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrQuitPdu: Instances of DtLgrQuitPdu are messages telling a remote logger to quit

- DtLgrQuitPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)**
default constructor

- DtLgrQuitPdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)**
construct from network representation

- DtLgrQuitPdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)**

construct and prepare for send

- `virtual ~DtLgrQuitPdu()`

destructor

- `DtLgrQuitPdu(const DtLgrQuitPdu& orig)`

copy constructor

- `DtLgrQuitPdu& operator=(const DtLgrQuitPdu& orig)`

assignment operator

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcplay.h:

class DtLgrPlayPdu: public DtLgrControlPdu

class DtLgrPlayPdu: Instances of DtLgrPlayPdu are used to tell remote logger to begin recording

Inheritance:

DtLgrPlayPdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrPlayPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
 - **DtLgrPlayPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
 - **DtLgrPlayPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
 - virtual **~DtLgrPlayPdu** ()
destructor
 - **DtLgrPlayPdu** (const [DtLgrPlayPdu](#)& orig)
copy constructor
 - [DtLgrPlayPdu](#)& **operator=** (const [DtLgrPlayPdu](#)& orig)
assignment operator
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)

- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const

- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrPlayPdu: Instances of DtLgrPlayPdu are used to tell remote logger to begin recording

- DtLgrPlayPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor

- DtLgrPlayPdu**(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation

- DtLgrPlayPdu**(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

construct and prepare for send

- `virtual ~DtLgrPlayPdu()`

destructor

- `DtLgrPlayPdu(const DtLgrPlayPdu& orig)`

copy constructor

- `DtLgrPlayPdu& operator=(const DtLgrPlayPdu& orig)`

assignment operator

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcplayaction.h:

class DtLgrPlayActionPdu: public DtLgrControlPdu

class DtLgrPlayActionPdu: Instances of DtLgrPlayActionPdu are to tell remote logger what action to take initially when set to playback mode

Inheritance:

DtLgrPlayActionPdu - DtLgrControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtLgrPlayActionPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
- **DtLgrPlayActionPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
- **DtLgrPlayActionPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtLgrMode mode, DtBufferPtr buffer =DtUSE_INTERNAL_BUFFER)
construct and prepare for send
- virtual **~DtLgrPlayActionPdu** ()
destructor
- **DtLgrPlayActionPdu** (const DtLgrPlayActionPdu& orig)
copy constructor
- DtLgrPlayActionPdu& **operator=** (const DtLgrPlayActionPdu& orig)
assignment operator
- virtual void **setAction** (DtLgrMode cmd)
set/get action to take place when logger switched to playback mode
- virtual DtLgrMode **action** ()
- virtual DtLgrIntParams* **params** ()
return correctly casted parameters pointer; see lgrControlIP

Private Fields

- DtLgrMode **myAction**
parameter
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)

- virtual int **protocolVersion**() const
- virtual void **setExerciseId**(int id)
- virtual int **exerciseId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseId**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const

- virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrPlayActionPdu: Instances of DtLgrPlayActionPdu are to tell remote logger what action to take initially when set to playback mode

- `DtLgrPlayActionPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
default constructor
- `DtLgrPlayActionPdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct from network representation
- `DtLgrPlayActionPdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtLgrMode mode, DtBufferPtr buffer =DtUSE_INTERNAL_BUFFER)`
construct and prepare for send
- `virtual ~DtLgrPlayActionPdu()`
destructor
- `DtLgrPlayActionPdu(const DtLgrPlayActionPdu& orig)`
copy constructor
- `DtLgrPlayActionPdu& operator=(const DtLgrPlayActionPdu& orig)`
assignment operator
- `virtual void setAction( DtLgrMode cmd )`
set/get action to take place when logger switched to playback mode
- `virtual DtLgrMode action()`
- `virtual DtLgrIntParams* params()`
return correctly casted parameters pointer; see lgrControlP
- `DtLgrMode myAction`
parameter

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcplayfile.h:

class DtLgrPbFilePdu: public DtLgrControlPdu

class DtLgrPbFilePdu: Instances of DtLgrPbFilePdu are used for switching to play-mode and defining which file to read from when sending traffic into the simulation

Inheritance:

DtLgrPbFilePdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtLgrPbFilePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>default constructor</i>
	DtLgrPbFilePdu (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>construct from network representation</i>
	DtLgrPbFilePdu (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, char* filename, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>construct and prepare for send</i>
 virtual	~DtLgrPbFilePdu () <i>destructor</i>
	DtLgrPbFilePdu (const DtLgrPbFilePdu & orig) <i>copy constructor</i>
 DtLgrPbFilePdu &	operator= (const DtLgrPbFilePdu & orig) <i>assignment operator</i>
 virtual void	setFilename (const char* path) <i>set/get file to open for playback</i>
 virtual const char*	filename ()
 virtual DtLgrStringParams*	params () <i>return correctly casted parameters pointer; see lgrControlP</i>
 virtual void	printData () const <i>diagnostic output</i>

Private Fields

	char* myFilename <i>parameter</i>
---	---

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const

- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)

- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrPbFilePdu: Instances of DtLgrPbFilePdu are used for switching to play-mode and defining which file to read from when sending traffic into the simulation

- `DtLgrPbFilePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
default constructor
- `DtLgrPbFilePdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct from network representation
- `DtLgrPbFilePdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, char* filename, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct and prepare for send
- `virtual ~DtLgrPbFilePdu()`
destructor
- `DtLgrPbFilePdu(const DtLgrPbFilePdu& orig)`
copy constructor
- `DtLgrPbFilePdu& operator=(const DtLgrPbFilePdu& orig)`
assignment operator
- `virtual void setFilename(const char* path)`
set/get file to open for playback. (Path relative to logger)
- `virtual const char* filename()`
- `virtual DtLgrStringParams* params()`
return correctly casted parameters pointer; see lgrControlP
- `virtual void printData() const`
diagnostic output
- `char* myFilename`
parameter

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcpause.h:

class DtLgrPausePdu: public DtLgrControlPdu

class DtLgrPausePdu: Instances of DtLgrPausePdu are to tell remote logger to pause

Inheritance:

DtLgrPausePdu - DtLgrControlPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtLgrPausePdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
 - **DtLgrPausePdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
 - **DtLgrPausePdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
 - virtual **~DtLgrPausePdu** ()
destructor
 - **DtLgrPausePdu** (const [DtLgrPausePdu](#)& orig)
copy constructor
 - [DtLgrPausePdu](#)& **operator=** (const [DtLgrPausePdu](#)& orig)
assignment operator
-

Inherited from DtLgrControlPdu:

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)

- `const DtGlobalObjectDesignator& receiverId() const`
- `virtual void setSenderId(const DtGlobalObjectDesignator& id)`
- `virtual const DtGlobalObjectDesignator& senderId() const`
- `DtLgrCommand lgrCommand() const`
- `virtual void printData() const`
- `inline DtPduKind internalGetPduKind() const`

Protected Methods

- `DtPduKind internalGetPduKind() const`
- `void setLgrCommand(DtLgrCommand com)`

Private Fields

- `DtGlobalObjectDesignator mySenderId`
 - `DtGlobalObjectDesignator myReceiverId`
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- `virtual void* packet() const`
- `virtual int status() const`
- `virtual void setVersion(int vers)`
- `virtual int protocolVersion() const`
- `virtual void setExercisId(int id)`
- `virtual int exercisId() const`
- `virtual DtPduKind kind() const`
- `virtual DtProtocolFamily protocolFamily() const`
- `virtual void setTimeStamp(DtTime time, DtTimeStampType type = DtTimeStampRelative)`
- `virtual DtTime timeStamp() const`
- `virtual DtTimeStampType timeStampType() const`

- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrPausePdu: Instances of DtLgrPausePdu are to tell remote logger to pause

- DtLgrPausePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)**
default constructor

- DtLgrPausePdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)**
construct from network representation

- DtLgrPausePdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)**

construct and prepare for send

- `virtual ~DtLgrPausePdu()`

destructor

- `DtLgrPausePdu(const DtLgrPausePdu& orig)`

copy constructor

- `DtLgrPausePdu& operator=(const DtLgrPausePdu& orig)`

assignment operator

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lheartbeat.h:

class DtLgrHeartbeatPdu: public DtLgrControlPdu

class DtLgrHeartbeatPdu: Instances of DtLgrHeartbeatPdu are to tell remote logger to send packet for each playback entity at least every HEARTBEAT seconds

Inheritance:

DtLgrHeartbeatPdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrHeartbeatPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
- **DtLgrHeartbeatPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
- **DtLgrHeartbeatPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtReal hbeat, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
- virtual **~DtLgrHeartbeatPdu** ()
destructor
- **DtLgrHeartbeatPdu** (const [DtLgrHeartbeatPdu](#)& orig)
copy constructor
- [DtLgrHeartbeatPdu](#)& **operator=** (const [DtLgrHeartbeatPdu](#)& orig)
assignment operator
- virtual void **setHeartbeat** (DtReal val)
set/get the heartbeat
- virtual DtReal **heartbeat** ()
- virtual DtLgrRealParams* **params** ()
return correctly casted parameters pointer; see lgrControlP

Private Fields

- DtReal **myHeartbeat**
parameter
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)

- virtual int **protocolVersion**() const
- virtual void **setExerciseId**(int id)
- virtual int **exerciseId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseId**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const

- virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrHeartbeatPdu:

Instances of DtLgrHeartbeatPdu are to tell remote logger to send packet for each playback entity at least every HEARTBEAT seconds. Important when in slow-motion.

- `DtLgrHeartbeatPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
default constructor
- `DtLgrHeartbeatPdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct from network representation
- `DtLgrHeartbeatPdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtReal hbeat, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct and prepare for send
- `virtual ~DtLgrHeartbeatPdu()`
destructor
- `DtLgrHeartbeatPdu(const DtLgrHeartbeatPdu& orig)`
copy constructor
- `DtLgrHeartbeatPdu& operator=(const DtLgrHeartbeatPdu& orig)`
assignment operator
- `virtual void setHeartbeat( DtReal val)`
set/get the heartbeat. see class description.
- `virtual DtReal heartbeat()`
- `virtual DtLgrRealParams* params()`
return correctly casted parameters pointer; see lgrControlP
- `DtReal myHeartbeat`
parameter

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file Iceofaction.h:

class DtLgrEofActionPdu: public DtLgrControlPdu

class DtLgrEofActionPdu: Instances of DtLgrEofActionPdu are to tell remote logger that it should pause, wrap, or stop at the end of playback

Inheritance:

DtLgrEofActionPdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrEofActionPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
- **DtLgrEofActionPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
- **DtLgrEofActionPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, [DtLgrMode](#) mode, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
- virtual **~DtLgrEofActionPdu** ()
destructor
- **DtLgrEofActionPdu** (const [DtLgrEofActionPdu](#)& orig)
copy constructor
- [DtLgrEofActionPdu](#)& **operator=** (const [DtLgrEofActionPdu](#)& orig)
assignment operator
- virtual [DtLgrMode](#) **action** ()
Set/Get value for what to do when Logger reaches EOF
- virtual void **setAction** ([DtLgrMode](#) cmd)
mutator functions
- virtual DtLgrIntParams* **params** ()
return correctly casted parameters pointer; see lgrControlP

Private Fields

- [DtLgrMode](#) **myAction**
parameter
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const
- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)

- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const

- virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrEofActionPdu: Instances of DtLgrEofActionPdu are to tell remote logger that it should pause, wrap, or stop at the end of playback

- `DtLgrEofActionPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
default constructor
- `DtLgrEofActionPdu(const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct from network representation
- `DtLgrEofActionPdu(DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtLgrMode mode, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
construct and prepare for send
- `virtual ~DtLgrEofActionPdu()`
destructor
- `DtLgrEofActionPdu(const DtLgrEofActionPdu& orig)`
copy constructor
- `DtLgrEofActionPdu& operator=(const DtLgrEofActionPdu& orig)`
assignment operator
- `virtual DtLgrMode action()`
Set/Get value for what to do when Logger reaches EOF
- `virtual void setAction( DtLgrMode cmd )`
mutator functions
- `virtual DtLgrIntParams* params()`
return correctly casted parameters pointer; see lgrControlP
- `DtLgrMode myAction`
parameter

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcend.h:

class DtLgrEndPdu: public DtLgrControlPdu

class DtLgrEndPdu: Instances of DtLgrEndPdu are to tell remote logger to skip to end of file during playback

Inheritance:

DtLgrEndPdu - [DtLgrControlPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtLgrEndPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
 - **DtLgrEndPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
 - **DtLgrEndPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct and prepare for send
 - virtual **~DtLgrEndPdu** ()
destructor
 - **DtLgrEndPdu** (const [DtLgrEndPdu](#)& orig)
copy constructor
 - [DtLgrEndPdu](#)& **operator=** (const [DtLgrEndPdu](#)& orig)
assignment operator
-

Inherited from [DtLgrControlPdu](#):

Public Methods

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtLgrControlPdu](#)* **create**([DtLgrCommand](#) command)
- static void **addCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- static void **removeCallback**([DtExerciseConn](#)* conn, DtLgrControlInteractionCb cb, void* usr)
- virtual void **setReceiverId**(const DtGlobalObjectDesignator& id)
- const DtGlobalObjectDesignator& **receiverId**() const

- virtual void **setSenderId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **senderId**() const
- [DtLgrCommand](#) **lgrCommand**() const
- virtual void **printData**() const
- inline [DtPduKind](#) **internalGetPduKind**() const

Protected Methods

- [DtPduKind](#) **internalGetPduKind**() const
- void **setLgrCommand**([DtLgrCommand](#) com)

Private Fields

- DtGlobalObjectDesignator **mySenderId**
 - DtGlobalObjectDesignator **myReceiverId**
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const

- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtLgrEndPdu: Instances of DtLgrEndPdu are to tell remote logger to skip to end of file during playback

- DtLgrEndPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
default constructor
- DtLgrEndPdu** (const DtNetLgrControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
construct from network representation
- DtLgrEndPdu** (DtGlobalObjectDesignator& sender, DtGlobalObjectDesignator& receiver,

`DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

construct and prepare for send

• `virtual ~DtLgrEndPdu()`

destructor

• `DtLgrEndPdu(const DtLgrEndPdu& orig)`

copy constructor

• `DtLgrEndPdu& operator=(const DtLgrEndPdu& orig)`

assignment operator

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file firepdu.h:

class DtFirePdu: public DtInteraction

Inheritance:

DtFirePdu - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtFirePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtFirePdu (const DtNetFirePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static void	addCallback (DtExerciseConn *conn, DtFireInteractionCb cb, void *usr)
 static void	removeCallback (DtExerciseConn *conn, DtFireInteractionCb cb, void *usr)
	DtFirePdu (const DtEntityIdentifier &attacker, const DtEntityIdentifier &target, const DtEntityIdentifier &munition, const DtEventID &eventId, DtConstVector location, const DtBurstDescriptor &burst, DtConstVector linVelocity, const double range, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 const DtObjectID&	attackerId () const
 const DtObjectID&	targetId () const
 const DtObjectID&	munitionId () const
 const DtEventID&	eventId () const
 unsigned long	fireMissionIndex () const
 DtConstVector	location () const
 DtConstVector	velocity () const
 double	range () const
 DtDetonatorFuze	fuseType () const
 const DtEntityType &	munitionType () const
 int	quantity () const
 int	rate () const
 DtWarheadType	warheadType () const
 const DtBurstDescriptor &	burst () const
	<i>NOT USED IN HLA</i>

void	setAttackerId (const DtObjectId &id)
void	setTargetId (const DtObjectId &id)
void	setMunitionId (const DtObjectId &id)
void	setEventId (const DtEventID &id)
void	setFireMissionIndex (unsigned long i)
void	setLocation (DtConstVector location)
void	setVelocity (DtConstVector velocity)
void	setRange (double range)
void	setFuseType (DtDetonatorFuze f)
void	setMunitionType (const DtEntityType &t)
void	setQuantity (int q)
void	setRate (int r)
void	setWarheadType (DtWarheadType t)
void	setBurst (const DtBurstDescriptor &burst)
	<i>NOT USED IN HLA</i>
void	printData () const

Protected Fields

DtObjectId	att
DtObjectId	tar
DtObjectId	mun
DtEventID	event
DtVector	loc
DtVector	vel
DtBurstDescriptor	theBurst
DtEntityType	myMunType

Protected Methods

DtPduKind	internalGetPduKind () const
---------------------------	------------------------------------

Inherited from [DtInteraction](#):

Public Methods

• [DtInteraction](#)& operator=(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const

inline DtTime **timeStamp**() const

inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

int **myOKStatus**

Protected Methods

virtual void **setStatus**(int okStatus)

virtual DtNetPduHeader* **netHeader**() const

virtual void **initPdu**(int size, DtBufferPtr buffer)

virtual void **initPdu**(const void* initial, DtBufferPtr buffer)

virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const

virtual void **sizeChanged**(int size)

virtual void **clearData**()

virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

virtual const char* **netRep**() const

virtual int **length**() const

inline const char* **netRep**() const

Protected Fields

void* **myNetMessage**

void* **myInternalBuff**

int **myIntBuffSize**

int **myMsgSize**

Protected Methods

- `void initMsg(int size, DtBufferPtr bufferPtr)`
 - `void initMsg(const void *initial, int size, DtBufferPtr bufferPtr)`
 - `virtual void changeSize(int size)`
 - `virtual void insertBytes(int offset, int numBytes)`
 - `virtual void insertBytes(void *address, int numBytes)`
 - `virtual void deleteBytes(int offset, int numBytes)`
 - `virtual void deleteBytes(void *address, int numBytes)`
 - `virtual void clearAll()`
 - `virtual void* newInternalBuffer(int size)`
-

Documentation

- `DtFirePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtFirePdu(const DtNetFirePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `static void addCallback(DtExerciseConn *conn, DtFireInteractionCb cb, void *usr)`
- `static void removeCallback(DtExerciseConn *conn, DtFireInteractionCb cb, void *usr)`
- `DtFirePdu(const DtEntityIdentifier &attacker, const DtEntityIdentifier &target, const DtEntityIdentifier &muniton, const DtEventID &eventId, DtConstVector location, const DtBurstDescriptor &burst, DtConstVector linVelocity, const double range, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `const DtObjectId& attackerId() const`
- `const DtObjectId& targetId() const`
- `const DtObjectId& munitonId() const`
- `const DtEventID& eventId() const`
- `unsigned long fireMissionIndex() const`
- `DtConstVector location() const`
- `DtConstVector velocity() const`
- `double range() const`
- `DtDetonatorFuze fuseType() const`
- `const DtEntityType& munitonType() const`

```

int quantity() const
int rate() const
DtWarheadType warheadType() const
const DtBurstDescriptor& burst() const
    NOT USED IN HLA
void setAttackerId(const DtObjectId &id)
void setTargetId(const DtObjectId &id)
void setMunitionId(const DtObjectId &id)
void setEventId(const DtEventID &id)
void setFireMissionIndex(unsigned long i)
void setLocation(DtConstVector location)
void setVelocity(DtConstVector velocity)
void setRange(double range)
void setFuseType(DtDetonatorFuze f)
void setMunitionType(const DtEntityType &t)
void setQuantity(int q)
void setRate(int r)
void setWarheadType(DtWarheadType t)
void setBurst(const DtBurstDescriptor &burst)
    NOT USED IN HLA
void printData() const
DtPduKind internalGetPduKind() const
DtObjectId att
DtObjectId tar
DtObjectId mun
DtEventID event
DtVector loc
DtVector vel
DtBurstDescriptor theBurst
DtEntityType myMunType

```

This class has no child classes.

In file hoststructs.h:

class DtBurstDescriptor

Public Fields

- [DtEntityType](#) DtMunition
- [DtWarheadType](#) DtWarhead
- [DtDetonatorFuze](#) DtFuze
- int DtQuantity
- int DtRate

Public Methods

- [DtBurstDescriptor](#) ()
 - [DtBurstDescriptor](#) ([DtNetBurstDescriptor](#) *netBurst)
 - [DtBurstDescriptor](#) ([DtEntityType](#) *type, [DtWarheadType](#) warhead, [DtDetonatorFuze](#) fuze, DtU16 quantity, DtU16 rate)
 - operator [DtNetBurstDescriptor](#) () const
 - void init ([DtEntityType](#) *type, [DtWarheadType](#) warhead, [DtDetonatorFuze](#) fuze, int quantity, int rate)
-

Documentation

- [DtBurstDescriptor](#) ()
- [DtBurstDescriptor](#) ([DtNetBurstDescriptor](#) *netBurst)
- [DtBurstDescriptor](#) ([DtEntityType](#) *type, [DtWarheadType](#) warhead, [DtDetonatorFuze](#) fuze, DtU16 quantity, DtU16 rate)
- operator [DtNetBurstDescriptor](#) () const
- void init([DtEntityType](#) *type, [DtWarheadType](#) warhead, [DtDetonatorFuze](#) fuze, int quantity, int rate)
- [DtEntityType](#) DtMunition
- [DtWarheadType](#) DtWarhead
- [DtDetonatorFuze](#) DtFuze

int DtQuantity

int DtRate

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file hoststructs.h:

class DtEntityType

Inheritance:

DtEntityType

Public Fields

- int **DtEntityKind**
- int **DtDomain**
this conflicts with enum of same name, so must be before funcs that use it
- int **DtCountry**
- int **DtCategory**
- int **DtSubCategory**
- int **DtSpecific**
- int **DtExtra**

Public Methods

- DtEntityType** (const char* p)
- DtEntityType** (int kind, int domain, int country, int category, int subCategory, int specific, int extra)
- DtEntityType** ()
- DtEntityType** ([DtNetEntityType](#) netType)
- void **setFromNet** ([DtNetEntityType](#) netType)
- operator DtNetEntityType** () const
- void **init** (int kind, int domain, int country, int category, int subCategory, int specific, int extra)
- char* **string** () const
- static const [DtEntityType](#)& **nullType** ()
- int **kind** () const
int entityKind() const {return(DtEntityKind);}
- int **domain** () const
- int **country** () const

int	category () const
int	subCategory () const
int	specific () const
int	extra () const
void	setKind (int kind) <i>void setEntityKind(int kind){DtEntityKind = kind;}</i>
void	setDomain (int domain)
void	setCountry (int country)
void	setCategory (int category)
void	setSubCategory (int subCategory)
void	setSpecific (int specific)
void	setExtra (int extra)
void	setKind (long kind) <i>the funcs used to take longs, so for back-compat:</i>
void	setDomain (long domain)
void	setCountry (long country)
void	setCategory (long category)
void	setSubCategory (long subCategory)
void	setSpecific (long specific)
void	setExtra (long extra)
DtBoolean	matchPattern (const DtEntityType &pat) const
DtBoolean	DtTypeMatchPattern (const DtEntityType &pat) const
int	match (const DtEntityType &type) const <i>== returns 1 if they are equal, 0 if they are not equal</i>
int	operator== (const DtEntityType &type) const
int	operator!= (const DtEntityType &type) const <i>!= returns 0 if they are equal, nonzero if they are not equal</i>

Documentation

- `DtEntityType(const char* p)`
- `DtEntityType(int kind, int domain, int country, int category, int subCategory, int specific, int extra)`
- `DtEntityType()`
- `DtEntityType(DtNetEntityType netType)`
- `void setFromNet(DtNetEntityType netType)`
- `operator DtNetEntityType() const`

```

void init(int kind, int domain, int country, int category, int subCategory, int
specific, int extra)

char* string() const

static const DtEntityType& nullType()

int DtEntityKind

int DtDomain
    this conflicts with enum of same name, so must be before funcs that use it. Leave for awhile back-compat for
    awhile.

int DtCountry

int DtCategory

int DtSubCategory

int DtSpecific

int DtExtra

int kind() const
    int entityKind() const {return(DtEntityKind);}

int domain() const

int country() const

int category() const

int subCategory() const

int specific() const

int extra() const

void setKind(int kind)
    void setEntityKind(int kind){DtEntityKind = kind;}

void setDomain(int domain)

void setCountry(int country)

void setCategory(int category)

void setSubCategory(int subCategory)

void setSpecific(int specific)

void setExtra(int extra)

void setKind(long kind)
    the funcs used to take longs, so for back-compat:

void setDomain(long domain)

void setCountry(long country)

void setCategory(long category)

```

- `void setSubCategory(long subCategory)`
- `void setSpecific(long specific)`
- `void setExtra(long extra)`
- [DtBoolean](#) `matchPattern(const DtEntityType &pat) const`
- [DtBoolean](#) `DtTypeMatchPattern(const DtEntityType &pat) const`
- `int match(const DtEntityType &type) const`
== returns 1 if they are equal, 0 if they are not equal
- `int operator==(const DtEntityType &type) const`
- `int operator!=(const DtEntityType &type) const`
!= returns 0 if they are equal, nonzero if they are not equal

Direct child classes:

- [DtRadioEntityType](#)
- [DtEnvironmentTypeRecord](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file netstructs.h:

class DtNetEntityType

Public Fields

-  DtNetU8 **DtEntityKind**
-  DtNetU8 **DtDomain**
-  DtNetU16 **DtCountry**
-  DtNetU8 **DtCategory**
-  DtNetU8 **DtSubCategory**
-  DtNetU8 **DtSpecific**
-  DtNetU8 **DtExtra**

Public Methods

-  [DtEntityType](#) operator = ([DtEntityType](#))
 -  [DtNetEntityType](#) operator = ([DtNetEntityType](#))
-

Documentation

-  DtNetU8 DtEntityKind
 -  DtNetU8 DtDomain
 -  DtNetU16 DtCountry
 -  DtNetU8 DtCategory
 -  DtNetU8 DtSubCategory
 -  DtNetU8 DtSpecific
 -  DtNetU8 DtExtra
 -  [DtEntityType](#) operator = ([DtEntityType](#))
 -  [DtNetEntityType](#) operator = ([DtNetEntityType](#))
-

This class has no child classes.

In file hoststructs.h:

class DtRadioEntityType: public DtEntityType

Inheritance:

DtRadioEntityType - DtEntityType

Public Methods

	DtRadioEntityType ()
	DtRadioEntityType (const char* p)
	DtRadioEntityType (int kind, int domain, int country, int category, int nomenclatureVersion, int nomenclature)
	DtRadioEntityType (DtNetRadioEntityType netType)
 void	setFromNet (DtNetRadioEntityType netType)
 int	nomenclatureVersion () const
 int	nomenclature () const
 void	setNomenclatureVersion (int version)
 void	setNomenclature (int nom)
 char*	string () const
 static const DtRadioEntityType &	nullType ()
	operator DtNetRadioEntityType () const

Inherited from DtEntityType:

Public Fields

-  int **DtEntityKind**
-  int **DtDomain**
-  int **DtCountry**
-  int **DtCategory**
-  int **DtSubCategory**

int **DtSpecific**

int **DtExtra**

Public Methods

operator DtNetEntityType() const

void **init**(int kind, int domain, int country, int category, int subCategory, int specific, int extra)

int **kind**() const

int **domain**() const

int **country**() const

int **category**() const

int **subCategory**() const

int **specific**() const

int **extra**() const

void **setKind**(int kind)

void **setDomain**(int domain)

void **setCountry**(int country)

void **setCategory**(int category)

void **setSubCategory**(int subCategory)

void **setSpecific**(int specific)

void **setExtra**(int extra)

void **setKind**(long kind)

void **setDomain**(long domain)

void **setCountry**(long country)

void **setCategory**(long category)

void **setSubCategory**(long subCategory)

void **setSpecific**(long specific)

void **setExtra**(long extra)

[DtBoolean](#) **matchPattern**(const [DtEntityType](#) &pat) const

[DtBoolean](#) **DtTypeMatchPattern**(const [DtEntityType](#) &pat) const

int **match**(const [DtEntityType](#) &type) const

int **operator==**(const [DtEntityType](#) &type) const

int **operator!=**(const [DtEntityType](#) &type) const

Documentation

- `DtRadioEntityType()`
- `DtRadioEntityType(const char* p)`
- `DtRadioEntityType(int kind, int domain, int country, int category, int nomenclatureVersion, int nomenclature)`
- `DtRadioEntityType(DtNetRadioEntityType netType)`
- `void setFromNet(DtNetRadioEntityType netType)`
- `int nomenclatureVersion() const`
- `int nomenclature() const`
- `void setNomenclatureVersion(int version)`
- `void setNomenclature(int nom)`
- `char* string() const`
- `static const DtRadioEntityType& nullType()`
- `operator DtNetRadioEntityType() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file netstructs.h:

class DtNetRadioEntityType

Public Fields

-  DtNetU8 **kind**
-  DtNetU8 **domain**
-  DtNetU16 **country**
-  DtNetU8 **category**
-  DtNetU8 **nomenclatureVersion**
-  DtNetU16 **nomenclature**

Public Methods

-  [DtRadioEntityType](#) **operator =** ([DtRadioEntityType](#))
 -  [DtNetRadioEntityType](#) **operator =** ([DtNetRadioEntityType](#))
-

Documentation

-  DtNetU8 **kind**
 -  DtNetU8 **domain**
 -  DtNetU16 **country**
 -  DtNetU8 **category**
 -  DtNetU8 **nomenclatureVersion**
 -  DtNetU16 **nomenclature**
 -  [DtRadioEntityType](#) **operator =** ([DtRadioEntityType](#))
 -  [DtNetRadioEntityType](#) **operator =** ([DtNetRadioEntityType](#))
-

This class has no child classes.

In file envirotyperec.h:

class DtEnvironmentTypeRecord: public DtEntityType

class DtEnvironmentTypeRecord: Instances of DtEnvironmentTypeRecord are used by the DtGriddedDataPdu for specifying environmental type

Inheritance:

DtEnvironmentTypeRecord - DtEntityType

Public Methods

	DtEnvironmentTypeRecord () <i>default constructor</i>
virtual	~DtEnvironmentTypeRecord () <i>destructor</i>
	DtEnvironmentTypeRecord (const DtEnvironmentTypeRecord & orig) <i>copy constructor</i>
DtEnvironmentTypeRecord &	operator= (const DtEnvironmentTypeRecord & orig) <i>assignment operators</i>
DtEnvironmentTypeRecord &	operator= (const DtNetEntityType & orig)
virtual void	setClassOfEnv (int cls) <i>Set/Get the environmental class</i>
virtual int	classOfEnv () const

Protected Methods

virtual void	setCountry (int country) <i>Get/Get country</i>
virtual int	country () const

Inherited from DtEntityType:

Public Fields

- int **DtEntityKind**
- int **DtDomain**
- int **DtCountry**
- int **DtCategory**
- int **DtSubCategory**
- int **DtSpecific**
- int **DtExtra**

Public Methods

- void **setFromNet**([DtNetEntityType](#) netType)
- operator DtNetEntityType**() const
- void **init**(int kind, int domain, int country, int category, int subCategory, int specific, int extra)
- char* **string**() const
- static const [DtEntityType](#)& **nullType**()
- int **kind**() const
- int **domain**() const
- int **category**() const
- int **subCategory**() const
- int **specific**() const
- int **extra**() const
- void **setKind**(int kind)
- void **setDomain**(int domain)
- void **setCategory**(int category)
- void **setSubCategory**(int subCategory)
- void **setSpecific**(int specific)
- void **setExtra**(int extra)
- void **setKind**(long kind)
- void **setDomain**(long domain)
- void **setCategory**(long category)
- void **setSubCategory**(long subCategory)

- void **setSpecific**(long specific)
 - void **setExtra**(long extra)
 - [DtBoolean](#) **matchPattern**(const [DtEntityType](#) &pat) const
 - [DtBoolean](#) **DtTypeMatchPattern**(const [DtEntityType](#) &pat) const
 - int **match**(const [DtEntityType](#) &type) const
 - int **operator==**(const [DtEntityType](#) &type) const
 - int **operator!=**(const [DtEntityType](#) &type) const
-

Documentation

class DtEnvironmentTypeRecord: Instances of DtEnvironmentTypeRecord are used by the DtGriddedDataPdu for specifying environmental type

- DtEnvironmentTypeRecord()**
default constructor
 - virtual ~DtEnvironmentTypeRecord()**
destructor
 - DtEnvironmentTypeRecord(const [DtEnvironmentTypeRecord](#)& orig)**
copy constructor
 - [DtEnvironmentTypeRecord](#)& operator=(const [DtEnvironmentTypeRecord](#)& orig)**
assignment operators
 - [DtEnvironmentTypeRecord](#)& operator=(const [DtNetEntityType](#)& orig)**
 - virtual void setClassOfEnv(int cls)**
Set/Get the environmental class
 - virtual int classOfEnv() const**
 - virtual void setCountry(int country)**
Get/Get country. This field is used for the environmental class.
 - virtual int country() const**
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtWarheadType

-  DtWarheadOther
 -  DtWarheadHighExplosive
 -  DtWarheadHighExplosivePlastic
 -  DtWarheadHighExplosiveIncendiary
 -  DtWarheadHighExplosiveFragmentation
 -  DtWarheadHighExplosiveAntiTank
 -  DtWarheadHighExplosiveBomblets
 -  DtWarheadHighExplosiveShapedCharge
 -  DtWarheadSmoke
 -  DtWarheadIllumination
 -  DtWarheadPractice
 -  DtWarheadKinetic
 -  DtWarheadUnused
 -  DtWarheadNuclear
 -  DtWarheadChemicalGeneral
 -  DtWarheadChemicalBlisterAgent
 -  DtWarheadChemicalBloodAgent
 -  DtWarheadChemicalNerveAgent
 -  DtWarheadBiologicalGeneral
-

Documentation

-  DtWarheadOther
-  DtWarheadHighExplosive
-  DtWarheadHighExplosivePlastic
-  DtWarheadHighExplosiveIncendiary

- DtWarheadHighExplosiveFragmentation
- DtWarheadHighExplosiveAntiTank
- DtWarheadHighExplosiveBomblets
- DtWarheadHighExplosiveShapedCharge
- DtWarheadSmoke
- DtWarheadIllumination
- DtWarheadPractice
- DtWarheadKinetic
- DtWarheadUnused
- DtWarheadNuclear
- DtWarheadChemicalGeneral
- DtWarheadChemicalBlisterAgent
- DtWarheadChemicalBloodAgent
- DtWarheadChemicalNerveAgent
- DtWarheadBiologicalGeneral

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtDetonatorFuze

-
-  DtFuzeOther
 -  DtFuzeContact
 -  DtFuzeContactInstant
 -  DtFuzeContactDelayed
 -  DtFuzeTimed
 -  DtFuzeProximity
 -  DtFuzeCommand
 -  DtFuzeAltitude
 -  DtFuzeDepth
 -  DtFuzeAcoustic
-

Documentation

-  DtFuzeOther
-  DtFuzeContact
-  DtFuzeContactInstant
-  DtFuzeContactDelayed
-  DtFuzeTimed
-  DtFuzeProximity
-  DtFuzeCommand
-  DtFuzeAltitude
-  DtFuzeDepth
-  DtFuzeAcoustic

In file netstructs.h:

class DtNetBurstDescriptor

Public Fields

- [DtNetEntityType](#) DtMunition
- [DtNetU16](#) DtWarhead
- [DtNetU16](#) DtFuze
- [DtNetU16](#) DtQuantity
- [DtNetU16](#) DtRate

Public Methods

- [DtBurstDescriptor](#) operator = ([DtBurstDescriptor](#))
 - [DtNetBurstDescriptor](#) operator = ([DtNetBurstDescriptor](#))
-

Documentation

- [DtNetEntityType](#) DtMunition
 - [DtNetU16](#) DtWarhead
 - [DtNetU16](#) DtFuze
 - [DtNetU16](#) DtQuantity
 - [DtNetU16](#) DtRate
 - [DtBurstDescriptor](#) operator = ([DtBurstDescriptor](#))
 - [DtNetBurstDescriptor](#) operator = ([DtNetBurstDescriptor](#))
-

This class has no child classes.

In file detonatepdu.h:

class DtDetonationPdu: public DtInteraction

Inheritance:

DtDetonationPdu - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtDetonationPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtDetonationPdu (const DtNetDetonationPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtDetonationPdu (const DtFireInteraction &fire, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Fills in common fields with data from Fire Interaction</i>
 static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static void	addCallback (DtExerciseConn *conn, DtDetonationInteractionCb cb, void *usr)
 static void	removeCallback (DtExerciseConn *conn, DtDetonationInteractionCb cb, void *usr)
	DtDetonationPdu (const DtEntityIdentifier &attacker, const DtEntityIdentifier &target, const DtEntityIdentifier &munition, const DtEventID &eventId, DtConstVector linVelocity, DtConstVector location, const DtBurstDescriptor &burst, DtConstVector entLocation, DtDetonationResult result, const DtArtPartSpec *specList, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 const DtObjectID&	attackerId () const
 const DtObjectID&	targetId () const
 const DtObjectID&	munitionId () const
 const DtEventID&	eventId () const
 DtConstVector	velocity () const
 DtConstVector	worldLocation () const
 DtConstVector	entityLocation () const
 DtDetonationResult	result () const
 DtDetonatorFuze	fuseType () const
 const DtEntityType &	munitionType () const
 int	quantity () const

int	rate () const
DtWarheadType	warheadType () const
const DtBurstDescriptor &	burst () const <i>NOT USED IN HLA</i>
int	numArtParams () const
const DtArtPartSpec*	artPartSpecs () const
int	getArtParamVal (int paramIndex, DtFloat32 *value) const
int	getArtParamVal (int partType, int param, DtFloat32 *value) const
void	setAttackerId (const DtObjectId &id)
void	setTargetId (const DtObjectId &id)
void	setMunitionId (const DtObjectId &id)
void	setEventId (const DtEventID &id)
void	setVelocity (DtConstVector velocity)
void	setWorldLocation (DtConstVector location)
void	setEntityLocation (DtConstVector location)
void	setResult (DtDetonationResult result)
void	setFuseType (DtDetonatorFuze fuse)
void	setMunitionType (const DtEntityType &munType)
void	setQuantity (int quant)
void	setRate (int rate)
void	setWarheadType (DtWarheadType type)
void	setBurst (const DtBurstDescriptor &burst) <i>NOT USED IN HLA</i>
void	initArtParts (const DtArtPartSpec *specList)
int	setArtParamVal (int partType, int param, DtFloat32 value)
int	setArtParamVal (int paramIndex, DtFloat32 value)
void	printData () const

Protected Fields

DtObjectId	att
DtObjectId	tar
DtObjectId	mun
DtEventID	event
DtVector	vel
DtVector	world
DtVector	ent

 [DtBurstDescriptor](#) theBurst

 [DtEntityType](#) myMunType

Protected Methods

 [DtPduKind](#) internalGetPduKind () const

 void setNumArtParams (int numParams)

 DtNetArtPartRecord* artPartRecord (int partType, int param) const

 void setArtParamVal (DtNetArtPartRecord *part, DtFloat32 value)

 float getArtParamVal (DtNetArtPartRecord *part) const

Inherited from [DtInteraction](#):

Public Methods

 [DtInteraction](#)& operator=(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

 virtual void* packet() const

 virtual int status() const

 virtual void setVersion(int vers)

 virtual int protocolVersion() const

 virtual void setExerciseld(int id)

 virtual int exerciseld() const

 virtual [DtPduKind](#) kind() const

 virtual [DtProtocolFamily](#) protocolFamily() const

 virtual void setTimeStamp(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

 virtual DtTime timeStamp() const

 virtual [DtTimeStampType](#) timeStampType() const

 virtual DtTime guessTimeValid(DtTime referenceTime) const

 virtual void print() const

- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtDetonationPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtDetonationPdu(const DtNetDetonationPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtDetonationPdu(const DtFireInteraction &fire, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Fills in common fields with data from Fire Interaction
- `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `static void addCallback(DtExerciseConn *conn, DtDetonationInteractionCb cb, void *usr)`

```

static void removeCallback(DtExerciseConn *conn, DtDetonationInteractionCb cb, void
*usr)

DtDetonationPdu(const DtEntityIdentifier &attacker, const DtEntityIdentifier &target,
const DtEntityIdentifier &munition, const DtEventID &eventId, DtConstVector linVelocity,
DtConstVector location, const DtBurstDescriptor &burst, DtConstVector entLocation,
DtDetonationResult result, const DtArtPartSpec *specList, DtBufferPtr buffer =
DtUSE_INTERNAL_BUFFER)

const DtObjectId& attackerId() const

const DtObjectId& targetId() const

const DtObjectId& munitionId() const

const DtEventID& eventId() const

DtConstVector velocity() const

DtConstVector worldLocation() const

DtConstVector entityLocation() const

DtDetonationResult result() const

DtDetonatorFuze fuseType() const

const DtEntityType& munitionType() const

int quantity() const

int rate() const

DtWarheadType warheadType() const

const DtBurstDescriptor& burst() const

    NOT USED IN HLA

int numArtParams() const

const DtArtPartSpec* artPartSpecs() const

int getArtParamVal(int paramIndex, DtFloat32 *value) const

int getArtParamVal(int partType, int param, DtFloat32 *value) const

void setAttackerId(const DtObjectId &id)

void setTargetId(const DtObjectId &id)

void setMunitionId(const DtObjectId &id)

void setEventId(const DtEventID &id)

void setVelocity(DtConstVector velocity)

void setWorldLocation(DtConstVector location)

void setEntityLocation(DtConstVector location)

void setResult(DtDetonationResult result)

```

- `void setFuseType(DtDetonatorFuze fuse)`
- `void setMunitionType(const DtEntityType &munType)`
- `void setQuantity(int quant)`
- `void setRate(int rate)`
- `void setWarheadType(DtWarheadType type)`
- `void setBurst(const DtBurstDescriptor &burst)`
- NOT USED IN HLA
- `void initArtParts(const DtArtPartSpec *specList)`
- `int setArtParamVal(int partType, int param, DtFloat32 value)`
- `int setArtParamVal(int paramIndex, DtFloat32 value)`
- `void printData() const`
- `DtPduKind internalGetPduKind() const`
- `void setNumArtParams(int numParams)`
- `DtNetArtPartRecord* artPartRecord(int partType, int param) const`
- `void setArtParamVal(DtNetArtPartRecord *part, DtFloat32 value)`
- `float getArtParamVal(DtNetArtPartRecord *part) const`
- `DtObjectId att`
- `DtObjectId tar`
- `DtObjectId mun`
- `DtEventID event`
- `DtVector vel`
- `DtVector world`
- `DtVector ent`
- `DtBurstDescriptor theBurst`
- `DtEntityType myMunType`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtDetonationResult

- DtDetResOther
 - DtDetResEntityImpact
 - DtDetResEntityProximate
 - DtDetResGroundImpact
 - DtDetResGroundProximate
 - DtDetResDetonation
 - DtDetResNone
 - DtDetResHE_HitSmall
 - DtDetResHE_HitMedium
 - DtDetResHE_HitLarge
 - DtDetResArmorPiercingHit
 - DtDetResDirtBlastSmall
 - DtDetResDirtBlastMedium
 - DtDetResDirtBlastLarge
 - DtDetResWaterBlastSmall
 - DtDetResWaterBlastMedium
 - DtDetResWaterBlastLarge
 - DtDetResAirHit
 - DtDetResBuildingHitSmall
 - DtDetResBuildingHitMedium
 - DtDetResBuildingHitLarge
 - DtDetResMineClearingLineCharge
 - DtDetResEnvironmentObjectImpact
 - DtDetResEnvironmentObjectProximate
 - DtDetResWaterImpact
 - DtDetResAirBurst
-

Documentation

- DtDetResOther
- DtDetResEntityImpact
- DtDetResEntityProximate
- DtDetResGroundImpact
- DtDetResGroundProximate
- DtDetResDetonation
- DtDetResNone
- DtDetResHE_HitSmall
- DtDetResHE_HitMedium
- DtDetResHE_HitLarge
- DtDetResArmorPiercingHit
- DtDetResDirtBlastSmall
- DtDetResDirtBlastMedium
- DtDetResDirtBlastLarge
- DtDetResWaterBlastSmall
- DtDetResWaterBlastMedium
- DtDetResWaterBlastLarge
- DtDetResAirHit
- DtDetResBuildingHitSmall
- DtDetResBuildingHitMedium
- DtDetResBuildingHitLarge
- DtDetResMineClearingLineCharge
- DtDetResEnvironmentObjectImpact
- DtDetResEnvironmentObjectProximate
- DtDetResWaterImpact
- DtDetResAirBurst

[Alphabetic index](#) [Hierarchy of classes](#)

In file collidepdu.h:

class DtCollisionPdu: public DtInteraction

Inheritance:

DtCollisionPdu - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtCollisionPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>STANDARD CONSTRUCTORS *****</i>
	DtCollisionPdu (const DtNetCollisionPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static void	addCallback (DtExerciseConn *conn, DtCollisionInteractionCb cb, void *usr)
static void	removeCallback (DtExerciseConn *conn, DtCollisionInteractionCb cb, void *usr)
const DtEntityIdentifier &	issueld () const <i>INSPECTOR FUNCTIONS *****</i>
const DtEntityIdentifier &	collideld () const
const DtEventID&	eventld () const
DtCollisionType	collisionType () const
DtConstVector	velocity () const
DtFloat32	mass () const
DtConstVector	entityCoords () const
void	setIssueld (const DtEntityIdentifier &id) <i>MUTATOR FUNCTIONS *****</i>
void	setCollideld (const DtEntityIdentifier &id)
void	setEventld (const DtEventID &id)
void	setCollisionType (DtCollisionType t)
void	setVelocity (DtConstVector veloc)
void	setMass (DtFloat32 mass)
void	setEntityCoords (DtConstVector coords)
void	printData () const

Protected Fields

-  [DtEntityIdentifier](#) iss
-  [DtEntityIdentifier](#) col
-  DtEventID event
-  [DtVector](#) vel
-  [DtVector](#) ent

Protected Methods

-  [DtPduKind](#) internalGetPduKind () const
-

Inherited from [DtInteraction](#):

Public Methods

-  [DtInteraction](#)& operator=(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

-  virtual void* **packet**() const
-  virtual int **status**() const
-  virtual void **setVersion**(int vers)
-  virtual int **protocolVersion**() const
-  virtual void **setExerciseld**(int id)
-  virtual int **exerciseld**() const
-  virtual [DtPduKind](#) **kind**() const
-  virtual [DtProtocolFamily](#) **protocolFamily**() const
-  virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
-  virtual DtTime **timeStamp**() const
-  virtual [DtTimeStampType](#) **timeStampType**() const
-  virtual DtTime **guessTimeValid**(DtTime referenceTime) const

- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtCollisionPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
STANDARD CONSTRUCTORS *****
- DtCollisionPdu**(const DtNetCollisionPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* **create**(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback**([DtExerciseConn](#) *conn, DtCollisionInteractionCb cb, void *usr)

```

static void removeCallback(DtExerciseConn *conn, DtCollisionInteractionCb cb, void
*usr)

const DtEntityIdentifier& issueId() const
    INSPECTOR FUNCTIONS *****

const DtEntityIdentifier& collideId() const

const DtEventID& eventId() const

DtCollisionType collisionType() const

DtConstVector velocity() const

DtFloat32 mass() const

DtConstVector entityCoords() const

void setIssueId(const DtEntityIdentifier &id)
    MUTATOR FUNCTIONS *****

void setCollideId(const DtEntityIdentifier &id)

void setEventId(const DtEventID &id)

void setCollisionType(DtCollisionType t)

void setVelocity(DtConstVector veloc)

void setMass(DtFloat32 mass)

void setEntityCoords(DtConstVector coords)

void printData() const

DtPduKind internalGetPduKind() const

DtEntityIdentifier iss

DtEntityIdentifier col

DtEventID event

DtVector vel

DtVector ent

```

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtCollisionType

-
-  DtInelastic
 -  DtElastic
 -  DtCollisionTypeOther
-

Documentation

-  DtInelastic
-  DtElastic
-  DtCollisionTypeOther

In file stopfrzrpdu.h:

class DtStopFreezeRPdu: public DtSimanPdu

Inheritance:

DtStopFreezeRPdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtStopFreezeRPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtStopFreezeRPdu (const DtNetStopFreezeRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtStopFreezeRPdu (const DtEntityIdentifier & to, const DtEntityIdentifier & from, const DtTime realTime, DtStopReason reason, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtStopFreezeRPdu (const DtStopFreezeRPdu & orig)
virtual	~DtStopFreezeRPdu () <i>destructor</i>
DtStopFreezeRPdu &	operator= (const DtStopFreezeRPdu & orig) <i>Assignment operator</i>
virtual void	setRealWorldTime (const DtTime time) <i>Set/Get real world time</i>
virtual DtTime	realWorldTime () const
virtual void	setReason (DtStopReason reason) <i>Set/Get the stop reason</i>
virtual DtStopReason	reason () const
virtual void	setFrozenBehavior (int behavior) <i>Set/Get the frozen behavior</i>
virtual int	frozenBehavior () const
virtual void	setReqRelService (DtRequiredReliabilityService value) <i>Set/Get the required reliability service</i>
virtual DtRequiredReliabilityService	reqRelService () const
virtual void	setRequestId (int id) <i>Set/Get the request id</i>
virtual int	requestId () const

virtual void

static [DtPdu](#)*

static void

static void

inline [DtPduKind](#)

printData () const

Print the data contained by this PDU

create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtStopFreezeRPdu

addCallback ([DtExerciseConn](#)* conn, DtStopFreezeRPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

removeCallback ([DtExerciseConn](#)* conn, DtStopFreezeRPduCb cb, void* usr)

internalGetPduKind () const

Protected Fields

static int **theStaticSize**

Protected Methods

virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

Inherited from [DtSimanPdu](#):

Public Methods

virtual const [DtEntityIdentifier](#)& **senderId**() const

virtual const [DtEntityIdentifier](#)& **receiverId**() const

virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)

virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

[DtEntityIdentifier](#) cache

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtStopFreezeRPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtStopFreezeRPdu**(const DtNetStopFreezeRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtStopFreezeRPdu**(const [DtEntityIdentifier](#)& to, const [DtEntityIdentifier](#)& from, const DtTime realTime, [DtStopReason](#) reason, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtStopFreezeRPdu**(const [DtStopFreezeRPdu](#)& orig)
- virtual **~DtStopFreezeRPdu**()
destructor
- [DtStopFreezeRPdu](#)& operator=(const [DtStopFreezeRPdu](#)& orig)
Assignment operator
- virtual void **setRealWorldTime**(const DtTime time)
Set/Get real world time
- virtual DtTime **realWorldTime**() const
- virtual void **setReason**([DtStopReason](#) reason)
Set/Get the stop reason
- virtual [DtStopReason](#) **reason**() const
- virtual void **setFrozenBehavior**(int behavior)
Set/Get the frozen behavior
- virtual int **frozenBehavior**() const
- virtual void **setReqRelService**([DtRequiredReliabilityService](#) value)
Set/Get the required reliability service
- virtual [DtRequiredReliabilityService](#) **reqRelService**() const
- virtual void **setRequestId**(int id)
Set/Get the request id
- virtual int **requestId**() const

- `virtual void printData() const`
Print the data contained by this PDU
 - `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtStopFreezeRPdu. Caller is responsible for deletion.
 - `static void addCallback(DtExerciseConn* conn, DtStopFreezeRPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
 - `static void removeCallback(DtExerciseConn* conn, DtStopFreezeRPduCb cb, void* usr)`
 - `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
 - `static int theStaticSize`
 - `inline DtPduKind internalGetPduKind() const`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtStopReason

-
-  DtStopReasonOther
 -  DtStopReasonRecess
 -  DtStopReasonTermination
 -  DtStopReasonSystemFailure
 -  DtStopReasonSecurityViolation
 -  DtStopReasonEntityReconstitution
-

Documentation

-  DtStopReasonOther
-  DtStopReasonRecess
-  DtStopReasonTermination
-  DtStopReasonSystemFailure
-  DtStopReasonSecurityViolation
-  DtStopReasonEntityReconstitution

In file disenums.h:

enum DtRequiredReliabilityService

Reliable service PDU enums

-  **DtAcknowledged**
 -  **DtUnacknowledged**
-

Documentation

Reliable service PDU enums

-  **DtAcknowledged**
-  **DtUnacknowledged**

In file stoppdu.h:

class DtStopFreezePdu: public DtSimanPdu

Inheritance:

DtStopFreezePdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtStopFreezePdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtStopFreezePdu** (const DtNetStopFreezePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtStopFreezePdu** (const [DtEntityIdentifier](#) &to, const [DtEntityIdentifier](#) &from, const DtTime realTime, [DtStopReason](#) reason, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** ([DtExerciseConn](#) *conn, DtStopFreezePduCb cb, void *usr)
- static void **removeCallback** ([DtExerciseConn](#) *conn, DtStopFreezePduCb cb, void *usr)
- DtTime **realWorldTime** () const
- [DtStopReason](#) **reason** () const
- int **frozenBehavior** () const
- int **requestId** () const
- void **setRealWorldTime** (const DtTime time)
- void **setReason** ([DtStopReason](#) reason)
- void **setFrozenBehavior** (int behavior)
- void **setRequestId** (int id)
- void **printData** () const

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
-

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const

- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtStopFreezePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtStopFreezePdu(const DtNetStopFreezePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtStopFreezePdu(const [DtEntityIdentifier](#) &to, const [DtEntityIdentifier](#) &from, const DtTime realTime, [DtStopReason](#) reason, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

- static void addCallback([DtExerciseConn](#) *conn, DtStopFreezePduCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtStopFreezePduCb cb, void *usr)
- DtTime realWorldTime() const
- [DtStopReason](#) reason() const
- int frozenBehavior() const
- int requestId() const
- void setRealWorldTime(const DtTime time)
- void setReason([DtStopReason](#) reason)
- void setFrozenBehavior(int behavior)
- void setRequestId(int id)
- void printData() const
- [DtPduKind](#) internalGetPduKind() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file startresrpdu.h:

class DtStartResumeRPdu: public DtSimanPdu

Inheritance:

DtStartResumeRPdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtStartResumeRPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtStartResumeRPdu (const DtNetStartResumeRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtStartResumeRPdu (const DtEntityIdentifier & to, const DtEntityIdentifier & from, const DtTime realTime, const DtTime simTime, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtStartResumeRPdu (const DtStartResumeRPdu & orig)
virtual	~DtStartResumeRPdu () <i>destructor</i>
DtStartResumeRPdu &	operator= (const DtStartResumeRPdu & orig) <i>Assignment operator</i>
virtual void	setRealWorldTime (const DtTime time) <i>Set/Get real world time</i>
virtual DtTime	realWorldTime () const
virtual void	setSimulationTime (const DtTime time) <i>Set/Get simulation time</i>
virtual DtTime	simulationTime () const
virtual void	setReqRelService (DtRequiredReliabilityService value) <i>Set/Get the required reliability service</i>
virtual DtRequiredReliabilityService	reqRelService () const
virtual void	setRequestId (int id) <i>Set/Get the request id</i>
virtual int	requestId () const
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtStartResumeRPdu</i>

static void

static void

inline [DtPduKind](#)

addCallback ([DtExerciseConn](#)* conn, DtStartResumeRPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

removeCallback ([DtExerciseConn](#)* conn, DtStartResumeRPduCb cb, void* usr)

internalGetPduKind () const

Protected Fields

static int **theStaticSize**

Protected Methods

virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

Inherited from [DtSimanPdu](#):

Public Methods

virtual const [DtEntityIdentifier](#)& **senderId**() const

virtual const [DtEntityIdentifier](#)& **receiverId**() const

virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)

virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

[DtEntityIdentifier](#) cache

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

virtual void* **packet**() const

virtual int **status**() const

virtual void **setVersion**(int vers)

- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const

- virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtStartResumeRPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Constructors
- `DtStartResumeRPdu(const DtNetStartResumeRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtStartResumeRPdu(const DtEntityIdentifier& to, const DtEntityIdentifier& from, const DtTime realTime, const DtTime simTime, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtStartResumeRPdu(const DtStartResumeRPdu& orig)`
- `virtual ~DtStartResumeRPdu()`
destructor
- `DtStartResumeRPdu& operator=(const DtStartResumeRPdu& orig)`
Assignment operator
- `virtual void setRealWorldTime(const DtTime time)`
Set/Get real world time
- `virtual DtTime realWorldTime() const`
- `virtual void setSimulationTime(const DtTime time)`
Set/Get simulation time
- `virtual DtTime simulationTime() const`
- `virtual void setReqRelService(DtRequiredReliabilityService value)`
Set/Get the required reliability service
- `virtual DtRequiredReliabilityService reqRelService() const`
- `virtual void setRequestId(int id)`
Set/Get the request id
- `virtual int requestId() const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtStartResumeRPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtStartResumeRPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtStartResumeRPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind

```
static int theStaticSize  
inline DtPduKind internalGetPduKind() const
```

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file startpdu.h:

class DtStartResumePdu: public DtSimanPdu

Inheritance:

DtStartResumePdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtStartResumePdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtStartResumePdu** (const DtNetStartResumePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtStartResumePdu** (const [DtEntityIdentifier](#) &to, const [DtEntityIdentifier](#) &from, const DtTime realTime, const DtTime simTime, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** ([DtExerciseConn](#) *conn, DtStartResumePduCb cb, void *usr)
- static void **removeCallback** ([DtExerciseConn](#) *conn, DtStartResumePduCb cb, void *usr)
- DtTime **realWorldTime** () const
- DtTime **simulationTime** () const
- int **requestId** () const
- void **setRealWorldTime** (const DtTime time)
- void **setSimulationTime** (const DtTime time)
- void **setRequestId** (int id)
- void **printData** () const

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
-

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const

- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtStartResumePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtStartResumePdu(const DtNetStartResumePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtStartResumePdu(const [DtEntityIdentifier](#) &to, const [DtEntityIdentifier](#) &from, const DtTime realTime, const DtTime simTime, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

- static void addCallback([DtExerciseConn](#) *conn, DtStartResumePduCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtStartResumePduCb cb, void *usr)
- DtTime realWorldTime() const
- DtTime simulationTime() const
- int requestId() const
- void setRealWorldTime(const DtTime time)
- void setSimulationTime(const DtTime time)
- void setRequestId(int id)
- void printData() const
- [DtPduKind](#) internalGetPduKind() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file removeentrpdu.h:

class DtRemoveEntityRPdu: public DtSimanPdu

Inheritance:

DtRemoveEntityRPdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtRemoveEntityRPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtRemoveEntityRPdu (const DtNetRemoveEntityRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtRemoveEntityRPdu (const DtEntityIdentifier & to, const DtEntityIdentifier & from, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtRemoveEntityRPdu (const DtRemoveEntityRPdu & orig)
 virtual	~DtRemoveEntityRPdu () <i>destructor</i>
 DtRemoveEntityRPdu &	operator= (const DtRemoveEntityRPdu & orig) <i>Assignment operator</i>
 virtual void	setReqRelService (DtRequiredReliabilityService value) <i>Set/Get the required reliability service</i>
 virtual DtRequiredReliabilityService	reqRelService () const
 virtual void	setRequestId (int id) <i>Set/Get the request id</i>
 virtual int	requestId () const
 virtual void	printData () const <i>Print the data contained by this PDU</i>
 static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtRemoveEntityRPdu</i>
 static void	addCallback (DtExerciseConn * conn, DtRemoveEntityRPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
 static void	removeCallback (DtExerciseConn * conn, DtRemoveEntityRPduCb cb, void* usr)
 inline DtPduKind	internalGetPduKind () const

Protected Fields

static int **theStaticSize**

Protected Methods

virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

Inherited from [DtSimanPdu](#):

Public Methods

virtual const [DtEntityIdentifier](#)& **senderId**() const
virtual const [DtEntityIdentifier](#)& **receiverId**() const
virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

[DtEntityIdentifier](#) **cache**

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

virtual void* **packet**() const
virtual int **status**() const
virtual void **setVersion**(int vers)
virtual int **protocolVersion**() const
virtual void **setExercisId**(int id)
virtual int **exercisId**() const
virtual [DtPduKind](#) **kind**() const

- virtual [DtProtocolFamily](#) **protocolFamily()** const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp()** const
- virtual [DtTimeStampType](#) **timeStampType()** const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader()** const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)

- virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtRemoveEntityRPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Constructors

- DtRemoveEntityRPdu**(const DtNetRemoveEntityRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

- `DtRemoveEntityRPdu(const DtEntityIdentifier& to, const DtEntityIdentifier& from, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtRemoveEntityRPdu(const DtRemoveEntityRPdu& orig)`
- `virtual ~DtRemoveEntityRPdu()`
destructor
- `DtRemoveEntityRPdu& operator=(const DtRemoveEntityRPdu& orig)`
Assignment operator
- `virtual void setReqRelService(DtRequiredReliabilityService value)`
Set/Get the required reliability service
- `virtual DtRequiredReliabilityService reqRelService() const`
- `virtual void setRequestId(int id)`
Set/Get the request id
- `virtual int requestId() const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtRemoveEntityRPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtRemoveEntityRPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtRemoveEntityRPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `static int theStaticSize`
- `inline DtPduKind internalGetPduKind() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file removepdu.h:

class DtRemoveEntityPdu: public DtSimanPdu

Inheritance:

DtRemoveEntityPdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtRemoveEntityPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtRemoveEntityPdu** (const DtNetRemoveEntityPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtRemoveEntityPdu** (const [DtEntityIdentifier](#) &to, const [DtEntityIdentifier](#) &from, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** ([DtExerciseConn](#) *conn, DtRemoveEntityPduCb cb, void *usr)
- static void **removeCallback** ([DtExerciseConn](#) *conn, DtRemoveEntityPduCb cb, void *usr)
- int **requestId** () const
- void **setRequestId** (int id)
- void **printData** () const

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
-

Inherited from DtSimanPdu:

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const

- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const

- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtRemoveEntityPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtRemoveEntityPdu(const DtNetRemoveEntityPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtRemoveEntityPdu(const [DtEntityIdentifier](#) &to, const [DtEntityIdentifier](#) &from, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - static void addCallback([DtExerciseConn](#) *conn, DtRemoveEntityPduCb cb, void *usr)
 - static void removeCallback([DtExerciseConn](#) *conn, DtRemoveEntityPduCb cb, void *usr)
 - int requestId() const
 - void setRequestId(int id)
 - void printData() const
 - [DtPduKind](#) internalGetPduKind() const
-

This class has no child classes.

[*Alphabetic index*](#) [*Hierarchy of classes*](#)

In file messagepdu.h:

class DtMessagePdu: public DtSimanPdu

Inheritance:

DtMessagePdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtMessagePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtMessagePdu (const DtNetMessagePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtMessagePdu (const DtEntityIdentifier &sender, const DtEntityIdentifier &receiver, const char *message, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static void	addCallback (DtExerciseConn *conn, DtMessagePduCb cb, void *usr)
 static void	removeCallback (DtExerciseConn *conn, DtMessagePduCb cb, void *usr)
 int	numVarFields () const
 DtDatumParam	varDatumId (int index) const
 int	varDatumSizeBits (int index) const
 int	varDatumSizeBytes (int index) const
 int	paramIndex (DtDatumParam param, int *retcode = NULL) const
 DtFloat32	datumValFloat32 (int index, int *retcode = NULL) const
 DtFloat64	datumValFloat64 (int index, int *retcode = NULL) const
 unsigned long	datumValUnsigned32 (int index, int *retcode = NULL) const
 long	datumValInt32 (int index, int *retcode = NULL) const
 const char*	datumValByteArray (int index) const
 DtFloat32	datumValFloat32 (DtDatumParam param, int *retcode = NULL) const
 DtFloat64	datumValFloat64 (DtDatumParam param, int *retcode = NULL) const
 unsigned long	datumValUnsigned32 (DtDatumParam param, int *retcode = NULL) const
 long	datumValInt32 (DtDatumParam param, int *retcode = NULL) const
 const char*	datumValByteArray (DtDatumParam param, int *retcode = NULL) const
 void	initDataSection (const DtVarDatumSpec *specList)

void	setNumVarFields (int newNum)
void	setDatumParam (int index, DtDatumParam param, int *retcode = NULL)
void	setVarDataBytes (int index, int length, int *retcode = NULL)
void	setMessage (const char *message, int size)
void	setDatumValFloat64 (int index, DtFloat64 val, int *retcode = NULL)
void	setDatumValFloat32 (int index, DtFloat32 val, int *retcode = NULL)
void	setDatumValInt32 (int index, long val, int *retcode = NULL)
void	setDatumValUnsigned32 (int index, unsigned long val, int *retcode = NULL)
void	setDatumValByteArray (int index, const char *val, int *retcode = NULL)
void	setDatumValFloat64 (DtDatumParam param, DtFloat64 val, int *retcode = NULL)
void	setDatumValFloat32 (DtDatumParam param, DtFloat32 val, int *retcode = NULL)
void	setDatumValInt32 (DtDatumParam param, long val, int *retcode = NULL)
void	setDatumValUnsigned32 (DtDatumParam param, unsigned long val, int *retcode = NULL)
void	setDatumValByteArray (DtDatumParam param, const char *val, int *retcode = NULL)
void	printData () const

Protected Methods

DtPduKind	internalGetPduKind () const
DtNetVarDatum*	nthVarDataRecord (int index) const
void*	datumValPtr (int index) const

Inherited from [DtSimanPdu](#):

Public Methods

virtual const	DtEntityIdentifier & senderId () const
virtual const	DtEntityIdentifier & receiverId () const
virtual void	setSenderId (const DtEntityIdentifier &id)
virtual void	setReceiverId (const DtEntityIdentifier &id)

Protected Fields

DtEntityIdentifier	cache
------------------------------------	-------

Inherited from [DtInteraction](#):

Public Methods

• [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)

- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtMessagePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtMessagePdu(const DtNetMessagePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtMessagePdu(const [DtEntityIdentifier](#) &sender, const [DtEntityIdentifier](#) &receiver, const char *message, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void addCallback([DtExerciseConn](#) *conn, DtMessagePduCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtMessagePduCb cb, void *usr)
- int numVarFields() const
- [DtDatumParam](#) varDatumId(int index) const
- int varDatumSizeBits(int index) const
- int varDatumSizeBytes(int index) const
- int paramIndex([DtDatumParam](#) param, int *retcode = NULL) const
- DtFloat32 datumValFloat32(int index, int *retcode = NULL) const
- DtFloat64 datumValFloat64(int index, int *retcode = NULL) const
- unsigned long datumValUnsigned32(int index, int *retcode = NULL) const
- long datumValInt32(int index, int *retcode = NULL) const
- const char* datumValByteArray(int index) const

- `DtFloat32 datumValFloat32(DtDatumParam param, int *retcode = NULL) const`
- `DtFloat64 datumValFloat64(DtDatumParam param, int *retcode = NULL) const`
- `unsigned long datumValUnsigned32(DtDatumParam param, int *retcode = NULL) const`
- `long datumValInt32(DtDatumParam param, int *retcode = NULL) const`
- `const char* datumValByteArray(DtDatumParam param, int *retcode = NULL) const`
- `void initDataSection(const DtVarDatumSpec *specList)`
- `void setNumVarFields(int newNum)`
- `void setDatumParam(int index, DtDatumParam param, int *retcode = NULL)`
- `void setVarDataBytes(int index, int length, int *retcode = NULL)`
- `void setMessage(const char *message, int size)`
- `void setDatumValFloat64(int index, DtFloat64 val, int *retcode = NULL)`
- `void setDatumValFloat32(int index, DtFloat32 val, int *retcode = NULL)`
- `void setDatumValInt32(int index, long val, int *retcode = NULL)`
- `void setDatumValUnsigned32(int index, unsigned long val, int *retcode = NULL)`
- `void setDatumValByteArray(int index, const char *val, int *retcode = NULL)`
- `void setDatumValFloat64(DtDatumParam param, DtFloat64 val, int *retcode = NULL)`
- `void setDatumValFloat32(DtDatumParam param, DtFloat32 val, int *retcode = NULL)`
- `void setDatumValInt32(DtDatumParam param, long val, int *retcode = NULL)`
- `void setDatumValUnsigned32(DtDatumParam param, unsigned long val, int *retcode = NULL)`
- `void setDatumValByteArray(DtDatumParam param, const char *val, int *retcode = NULL)`
- `void printData() const`
- `DtPduKind internalGetPduKind() const`
- `DtNetVarDatum* nthVarDataRecord(int index) const`
- `void* datumValPtr(int index) const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file dataqueryrpdu.h:

class DtDataQueryRPdu: public DtSimanPdu

Inheritance:

DtDataQueryRPdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtDataQueryRPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtDataQueryRPdu (const DtNetDataQueryRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtDataQueryRPdu (const DtDataQueryRPdu & orig)
 virtual	~DtDataQueryRPdu () <i>destructor</i>
 DtDataQueryRPdu &	operator= (const DtDataQueryRPdu & orig) <i>Assignment operator</i>
 virtual void	setRequestId (unsigned long id) <i>Set/Get the request id</i>
 virtual unsigned long	requestId () const
 virtual void	setTimeInterval (DtTime seconds) <i>Set/Get the time interval</i>
 virtual DtTime	timeInterval () const
 virtual unsigned long	numFixedFields () const <i>Get the number of fixed fields</i>
 virtual unsigned long	numVarFields () const <i>Get the number of variable fields</i>
 virtual int	setFixedDatumId (int index, DtDatumParam id) <i>Set/Get the fixed datum id at the specified index</i>
 virtual DtDatumParam	fixedDatumId (int index) const
 virtual int	setVarDatumId (int index, DtDatumParam id) <i>Set/Get the variable datum id at the specified index</i>
 virtual DtDatumParam	varDatumId (int index) const
 virtual void	setReqRelService (DtRequiredReliabilityService value) <i>Set/Get the required reliability service</i>
 virtual DtRequiredReliabilityService	reqRelService () const
 virtual void	initDatumIds (unsigned long numFixedFields, unsigned long numVarFields) <i>Initialize the fixed and variable fields</i>

virtual void

static [DtPdu](#)*

static void

static void

inline [DtPduKind](#)

printData () const

Print the data contained by this PDU

create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtDataQueryRPdu

addCallback ([DtExerciseConn](#)* conn, DtDataQueryRPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

removeCallback ([DtExerciseConn](#)* conn, DtDataQueryRPduCb cb, void* usr)

internalGetPduKind () const

Protected Fields

static int **theStaticSize**

Protected Methods

virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

Inherited from [DtSimanPdu](#):

Public Methods

virtual const [DtEntityIdentifier](#)& **senderId**() const

virtual const [DtEntityIdentifier](#)& **receiverId**() const

virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)

virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

[DtEntityIdentifier](#) cache

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtDataQueryRPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtDataQueryRPdu**(const DtNetDataQueryRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtDataQueryRPdu**(const [DtDataQueryRPdu](#)& orig)
- virtual **~DtDataQueryRPdu**()
destructor
- [DtDataQueryRPdu](#)& operator=(const [DtDataQueryRPdu](#)& orig)
Assignment operator
- virtual void **setRequestId**(unsigned long id)
Set/Get the request id
- virtual unsigned long **requestId**() const
- virtual void **setTimeInterval**(DtTime seconds)
Set/Get the time interval
- virtual DtTime **timeInterval**() const
- virtual unsigned long **numFixedFields**() const
Get the number of fixed fields
- virtual unsigned long **numVarFields**() const
Get the number of variable fields
- virtual int **setFixedDatumId**(int index, [DtDatumParam](#) id)
Set/Get the fixed datum id at the specified index
- virtual [DtDatumParam](#) **fixedDatumId**(int index) const
- virtual int **setVarDatumId**(int index, [DtDatumParam](#) id)
Set/Get the variable datum id at the specified index
- virtual [DtDatumParam](#) **varDatumId**(int index) const
- virtual void **setReqRelService**([DtRequiredReliabilityService](#) value)

Set/Get the required reliability service

• virtual [DtRequiredReliabilityService](#) reqRelService() const

• virtual void initDatumIds(unsigned long numFixedFields, unsigned long numVarFields)

Initialize the fixed and variable fields

• virtual void printData() const

Print the data contained by this PDU

• static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtDataQueryRPdu. Caller is responsible for deletion.

• static void addCallback([DtExerciseConn](#)* conn, DtDataQueryRPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

• static void removeCallback([DtExerciseConn](#)* conn, DtDataQueryRPduCb cb, void* usr)

• virtual [DtPduKind](#) internalGetPduKind() const

Returns the internal PDU kind

• static int theStaticSize

• inline [DtPduKind](#) internalGetPduKind() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file querypdu.h:

class DtDataQueryPdu: public DtSimanPdu

Inheritance:

DtDataQueryPdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

-  **DtDataQueryPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  **DtDataQueryPdu** (const DtNetDataQueryPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  static DtPdu* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  static void **addCallback** (DtExerciseConn *conn, DtDataQueryPduCb cb, void *usr)
-  static void **removeCallback** (DtExerciseConn *conn, DtDataQueryPduCb cb, void *usr)
-  unsigned long **requestId** () const
-  DtTime **timeInterval** () const
-  unsigned long **numFixedFields** () const
-  unsigned long **numVarFields** () const
-  DtDatumParam **fixedDatumId** (int index) const
-  DtDatumParam **varDatumId** (int index) const
-  void **setRequestId** (unsigned long id)
-  void **setTimeInterval** (DtTime seconds)
-  int **setFixedDatumId** (int index, DtDatumParam id)
-  int **setVarDatumId** (int index, DtDatumParam id)
-  void **initDatumIds** (unsigned long numFixedFields, unsigned long numVarFields)
-  void **printData** () const

Protected Methods

-  DtPduKind **internalGetPduKind** () const
-

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator**=(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const

- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtDataQueryPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtDataQueryPdu(const DtNetDataQueryPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void addCallback([DtExerciseConn](#) *conn, DtDataQueryPduCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtDataQueryPduCb cb, void *usr)

- unsigned long requestId() const
- DtTime timeInterval() const
- unsigned long numFixedFields() const
- unsigned long numVarFields() const
- [DtDatumParam](#) fixedDatumId(int index) const
- [DtDatumParam](#) varDatumId(int index) const
- void setRequestId(unsigned long id)
- void setTimeInterval(DtTime seconds)
- int setFixedDatumId(int index, [DtDatumParam](#) id)
- int setVarDatumId(int index, [DtDatumParam](#) id)
- void initDatumIds(unsigned long numFixedFields, unsigned long numVarFields)
- void printData() const
- [DtPduKind](#) internalGetPduKind() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file createentrpdu.h:

class DtCreateEntityRPdu: public DtSimanPdu

Inheritance:

DtCreateEntityRPdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtCreateEntityRPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtCreateEntityRPdu (const DtNetCreateEntityRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtCreateEntityRPdu (const DtEntityIdentifier & to, const DtEntityIdentifier & from, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtCreateEntityRPdu (const DtCreateEntityRPdu & orig)
 virtual	~DtCreateEntityRPdu () <i>destructor</i>
 DtCreateEntityRPdu &	operator= (const DtCreateEntityRPdu & orig) <i>Assignment operator</i>
 virtual void	setRequestId (int id) <i>Set/Get the request id</i>
 virtual int	requestId () const
 virtual void	setReqRelService (DtRequiredReliabilityService value) <i>Set/Get the required reliability service</i>
 virtual DtRequiredReliabilityService	reqRelService () const
 virtual void	printData () const <i>Print the data contained by this PDU</i>
 static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtCreateEntityRPdu</i>
 static void	addCallback (DtExerciseConn * conn, DtCreateEntityRPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
 static void	removeCallback (DtExerciseConn * conn, DtCreateEntityRPduCb cb, void* usr)
 inline DtPduKind	internalGetPduKind () const

Protected Fields

static int **theStaticSize**

Protected Methods

virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

Inherited from [DtSimanPdu](#):

Public Methods

virtual const [DtEntityIdentifier](#)& **senderId**() const
virtual const [DtEntityIdentifier](#)& **receiverId**() const
virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

[DtEntityIdentifier](#) **cache**

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

virtual void* **packet**() const
virtual int **status**() const
virtual void **setVersion**(int vers)
virtual int **protocolVersion**() const
virtual void **setExercisId**(int id)
virtual int **exercisId**() const
virtual [DtPduKind](#) **kind**() const

- virtual [DtProtocolFamily](#) **protocolFamily()** const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp()** const
- virtual [DtTimeStampType](#) **timeStampType()** const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader()** const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)

- virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtCreateEntityRPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors

- DtCreateEntityRPdu**(const DtNetCreateEntityRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

- `DtCreateEntityRPdu(const DtEntityIdentifier& to, const DtEntityIdentifier& from, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtCreateEntityRPdu(const DtCreateEntityRPdu& orig)`
- `virtual ~DtCreateEntityRPdu()`
destructor
- `DtCreateEntityRPdu& operator=(const DtCreateEntityRPdu& orig)`
Assignment operator
- `virtual void setRequestId(int id)`
Set/Get the request id
- `virtual int requestId() const`
- `virtual void setReqRelService(DtRequiredReliabilityService value)`
Set/Get the required reliability service
- `virtual DtRequiredReliabilityService reqRelService() const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtCreateEntityRPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtCreateEntityRPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtCreateEntityRPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `static int theStaticSize`
- `inline DtPduKind internalGetPduKind() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file createpdu.h:

class DtCreateEntityPdu: public DtSimanPdu

Inheritance:

DtCreateEntityPdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtCreateEntityPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtCreateEntityPdu** (const DtNetCreateEntityPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtCreateEntityPdu** (const [DtEntityIdentifier](#) &to, const [DtEntityIdentifier](#) &from, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** ([DtExerciseConn](#) *conn, DtCreateEntityPduCb cb, void *usr)
- static void **removeCallback** ([DtExerciseConn](#) *conn, DtCreateEntityPduCb cb, void *usr)
- int **requestId** () const
- void **setRequestId** (int id)
- void **printData** () const

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
-

Inherited from DtSimanPdu:

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const

- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const

- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtCreateEntityPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtCreateEntityPdu(const DtNetCreateEntityPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtCreateEntityPdu(const [DtEntityIdentifier](#) &to, const [DtEntityIdentifier](#) &from, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - static void addCallback([DtExerciseConn](#) *conn, DtCreateEntityPduCb cb, void *usr)
 - static void removeCallback([DtExerciseConn](#) *conn, DtCreateEntityPduCb cb, void *usr)
 - int requestId() const
 - void setRequestId(int id)
 - void printData() const
 - [DtPduKind](#) internalGetPduKind() const
-

This class has no child classes.

[*Alphabetic index*](#) [*Hierarchy of classes*](#)

In file ackrpdu.h:

class DtAcknowledgeRPdu: public DtSimanPdu

class DtAcknowledgeRPdu: Instances of DtAcknowledgeRPdu are used to send reliable acknowledge PDUs

Inheritance:

DtAcknowledgeRPdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

•	DtAcknowledgeRPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
•	DtAcknowledgeRPdu (const DtNetAcknowledgeRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
•	DtAcknowledgeRPdu (const DtAcknowledgeRPdu & orig)
• virtual	~DtAcknowledgeRPdu () <i>destructor</i>
• DtAcknowledgeRPdu &	operator= (const DtAcknowledgeRPdu & orig) <i>Assignment operator</i>
• virtual void	setAcknowledgeFlag (DtAcknowledgeFlag flag) <i>Set/Get the acknowledge flag</i>
• virtual DtAcknowledgeFlag	acknowledgeFlag () const
• virtual void	setResponseFlag (DtResponseFlag flag) <i>Set/Get the response flag</i>
• virtual DtResponseFlag	responseFlag () const
• virtual void	setRequestId (int id) <i>Set/Get the request id</i>
• virtual int	requestId () const
• virtual void	printData () const <i>Print the data contained by this PDU</i>
• static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtAcknowledgeRPdu</i>
• static void	addCallback (DtExerciseConn * conn, DtAcknowledgeRPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
• static void	removeCallback (DtExerciseConn * conn, DtAcknowledgeRPduCb cb, void* usr)
• inline DtPduKind	internalGetPduKind () const

Protected Fields

static int **theStaticSize**
Size of PDU with zero data bytes

Protected Methods

virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

Inherited from [DtSimanPdu](#):

Public Methods

virtual const [DtEntityIdentifier](#)& **senderId**() const
virtual const [DtEntityIdentifier](#)& **receiverId**() const
virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

[DtEntityIdentifier](#) **cache**

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

virtual void* **packet**() const
virtual int **status**() const
virtual void **setVersion**(int vers)
virtual int **protocolVersion**() const
virtual void **setExercisId**(int id)
virtual int **exercisId**() const

- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const

- virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtAcknowledgeRPdu: Instances of DtAcknowledgeRPdu are used to send reliable acknowledge PDUs

- DtAcknowledgeRPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Constructors

- `DtAcknowledgeRPdu(const DtNetAcknowledgeRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtAcknowledgeRPdu(const DtAcknowledgeRPdu& orig)`
- `virtual ~DtAcknowledgeRPdu()`
destructor
- `DtAcknowledgeRPdu& operator=(const DtAcknowledgeRPdu& orig)`
Assignment operator
- `virtual void setAcknowledgeFlag(DtAcknowledgeFlag flag)`
Set/Get the acknowledge flag
- `virtual DtAcknowledgeFlag acknowledgeFlag() const`
- `virtual void setResponseFlag(DtResponseFlag flag)`
Set/Get the response flag
- `virtual DtResponseFlag responseFlag() const`
- `virtual void setRequestId(int id)`
Set/Get the request id
- `virtual int requestId() const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtAcknowledgeRPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtAcknowledgeRPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtAcknowledgeRPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `static int theStaticSize`
Size of PDU with zero data bytes
- `inline DtPduKind internalGetPduKind() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtResponseFlag

 DtNoWay

 DtSureThing

Documentation

 DtNoWay

 DtSureThing

In file ackpdu.h:

class DtAcknowledgePdu: public DtSimanPdu

Inheritance:

DtAcknowledgePdu - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtAcknowledgePdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtAcknowledgePdu** (const DtNetAcknowledgePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** ([DtExerciseConn](#) *conn, DtAcknowledgeInteractionCb cb, void *usr)
- static void **removeCallback** ([DtExerciseConn](#) *conn, DtAcknowledgeInteractionCb cb, void *usr)
- **DtAcknowledgePdu** (const [DtEntityIdentifier](#) &to, const [DtEntityIdentifier](#) &from, [DtAcknowledgeFlag](#) aFlag, [DtResponseFlag](#) rFlag, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- [DtAcknowledgeFlag](#) **acknowledgeFlag** () const
- [DtResponseFlag](#) **responseFlag** () const
- int **requestId** () const
- void **setAcknowledgeFlag** ([DtAcknowledgeFlag](#) flag)
- void **setResponseFlag** ([DtResponseFlag](#) flag)
- void **setRequestId** (int id)
- void **printData** () const

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
-

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Public Methods

- [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const

- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtAcknowledgePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtAcknowledgePdu(const DtNetAcknowledgePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `static void addCallback(DtExerciseConn *conn, DtAcknowledgeInteractionCb cb, void *usr)`
- `static void removeCallback(DtExerciseConn *conn, DtAcknowledgeInteractionCb cb, void`

*usr)

• DtAcknowledgePdu(const [DtEntityIdentifier](#) &to, const [DtEntityIdentifier](#) &from, [DtAcknowledgeFlag](#) aFlag, [DtResponseFlag](#) rFlag, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

• [DtAcknowledgeFlag](#) acknowledgeFlag() const

• [DtResponseFlag](#) responseFlag() const

• int requestId() const

• void setAcknowledgeFlag([DtAcknowledgeFlag](#) flag)

• void setResponseFlag([DtResponseFlag](#) flag)

• void setRequestId(int id)

• void printData() const

• [DtPduKind](#) internalGetPduKind() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file setdatarpdu.h:

class DtSetDataRPdu: public DtPduWithData

Inheritance:

DtSetDataRPdu - DtPduWithData - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtSetDataRPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtSetDataRPdu (const DtNetSetDataRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtSetDataRPdu (const DtSetDataRPdu & orig)
 virtual	~DtSetDataRPdu () <i>destructor</i>
 DtSetDataRPdu &	operator= (const DtSetDataRPdu & orig) <i>Assignment operator</i>
 virtual void	setRequestId (unsigned long id) <i>Set/Get the request id</i>
 virtual unsigned long	requestId () const
 virtual void	setReqRelService (DtRequiredReliabilityService value) <i>Set/Get the required reliability service</i>
 virtual DtRequiredReliabilityService	reqRelService () const
 virtual void	printData () const <i>Print the data contained by this PDU</i>
 static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtSetDataRPdu</i>
 static void	addCallback (DtExerciseConn * conn, DtSetDataRPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
 static void	removeCallback (DtExerciseConn * conn, DtSetDataRPduCb cb, void* usr)
 inline DtPduKind	internalGetPduKind () const

Protected Fields

static int **theStaticSize**

Protected Methods

virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

virtual DtNetDataPduTail* **dataSection** () const
Returns the data portion of the net structure

Inherited from [DtPduWithData](#):

Public Methods

virtual int **numFixedFields**() const

virtual int **numVarFields**() const

virtual [DtDatumParam](#) **varDatumId**(int index) const

virtual [DtDatumParam](#) **fixedDatumId**(int index) const

virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const

virtual int **datumSizeBits**(int fixedOrVar, int index) const

virtual int **datumSizeBytes**(int fixedOrVar, int index) const

virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const

virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const

virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual const char* **datumValByteArray**(int fixedOrVar, int index) const

virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const

virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const

virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)

- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)
- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)
- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printDataSection**() const
- virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public



Protected Fields

- [DtDatumRecordSet](#)* myRecSet

Protected Methods

- virtual const [DtDatumRecordSet](#)* **datumRecSet**() const
 - virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const
 - virtual DtNetVarDatum* **nthVarDataRecord**(int index) const
 - virtual void* **datumValPtr**(int fixedOrVar, int index) const
-

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const

- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtSetDataRPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtSetDataRPdu**(const DtNetSetDataRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtSetDataRPdu**(const [DtSetDataRPdu](#)& orig)
- virtual ~DtSetDataRPdu**()
destructor
- [DtSetDataRPdu](#)**& operator=(const [DtSetDataRPdu](#)& orig)
Assignment operator

- virtual void setRequestId(unsigned long id)
Set/Get the request id
- virtual unsigned long requestId() const
- virtual void setReqRelService([DtRequiredReliabilityService](#) value)
Set/Get the required reliability service
- virtual [DtRequiredReliabilityService](#) reqRelService() const
- virtual void printData() const
Print the data contained by this PDU
- static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Create a DtSetDataRPdu. Caller is responsible for deletion.
- static void addCallback([DtExerciseConn](#)* conn, DtSetDataRPduCb cb, void* usr)
Add (register)/Remove (deregister) function to be called when pdu is received
- static void removeCallback([DtExerciseConn](#)* conn, DtSetDataRPduCb cb, void* usr)
- virtual [DtPduKind](#) internalGetPduKind() const
Returns the internal PDU kind
- virtual DtNetDataPduTail* dataSection() const
Returns the data portion of the net structure
- static int theStaticSize
- inline [DtPduKind](#) internalGetPduKind() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file setdatapdu.h:

class DtSetDataPdu: public DtPduWithData

Inheritance:

DtSetDataPdu - DtPduWithData - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

-  **DtSetDataPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  **DtSetDataPdu** (const DtNetSetDataPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  static void **addCallback** ([DtExerciseConn](#) *conn, DtSetDataPduCb cb, void *usr)
-  static void **removeCallback** ([DtExerciseConn](#) *conn, DtSetDataPduCb cb, void *usr)
-  unsigned long **requestId** () const
-  void **setRequestId** (unsigned long id)
-  void **printData** () const

Protected Methods

-  [DtPduKind](#) **internalGetPduKind** () const
 -  DtNetDataPduTail* **dataSection** () const
-

Inherited from DtPduWithData:

Public Methods

-  virtual int **numFixedFields**() const
-  virtual int **numVarFields**() const
-  virtual [DtDatumParam](#) **varDatumId**(int index) const

- virtual [DtDatumParam](#) **fixedDatumId**(int index) const
- virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const
- virtual int **datumSizeBits**(int fixedOrVar, int index) const
- virtual int **datumSizeBytes**(int fixedOrVar, int index) const
- virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const
- virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual const char* **datumValByteArray**(int fixedOrVar, int index) const
- virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)
- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)
- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)
- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printDataSection**() const

virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public



Protected Fields

[DtDatumRecordSet](#)* **myRecSet**

Protected Methods

virtual const [DtDatumRecordSet](#)* **datumRecSet**() const

virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const

virtual DtNetVarDatum* **nthVarDataRecord**(int index) const

virtual void* **datumValPtr**(int fixedOrVar, int index) const

Inherited from [DtSimanPdu](#):

Public Methods

virtual const [DtEntityIdentifier](#)& **senderId**() const

virtual const [DtEntityIdentifier](#)& **receiverId**() const

virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)

virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

[DtEntityIdentifier](#) **cache**

Inherited from [DtInteraction](#):

Public Methods

[DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtSetDataPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `DtSetDataPdu(const DtNetSetDataPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static void addCallback(DtExerciseConn *conn, DtSetDataPduCb cb, void *usr)`
 - `static void removeCallback(DtExerciseConn *conn, DtSetDataPduCb cb, void *usr)`
 - `unsigned long requestId() const`
 - `void setRequestId(unsigned long id)`
 - `void printData() const`
 - `DtPduKind internalGetPduKind() const`
 - `DtNetDataPduTail* dataSection() const`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file eventreptpdu.h:

class DtEventReptRPdu: public DtPduWithData

Inheritance:

DtEventReptRPdu - DtPduWithData - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

-  **DtEventReptRPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
-  **DtEventReptRPdu** (const DtNetEventReportPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  **DtEventReptRPdu** (const [DtEventReptRPdu](#)& orig)
-  virtual **~DtEventReptRPdu** ()
destructor
-  [DtEventReptRPdu](#)& **operator=** (const [DtEventReptRPdu](#)& orig)
Assignment operator
-  virtual void **setEventType** ([DtEventType](#) type)
Set/Get event type
-  virtual [DtEventType](#) **eventType** () const
-  virtual void **printData** () const
Print the data contained by this PDU
-  static [DtPdu](#)* **create** (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Create a DtEventReptRPdu
-  static void **addCallback** ([DtExerciseConn](#)* conn, DtEventReptRPduCb cb, void* usr)
Add (register)/Remove (deregister) function to be called when pdu is received
-  static void **removeCallback** ([DtExerciseConn](#)* conn, DtEventReptRPduCb cb, void* usr)
-  inline [DtPduKind](#) **internalGetPduKind** () const

Protected Fields

-  static int **theStaticSize**

Protected Methods

- virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind
 - virtual DtNetDataPduTail* **dataSection** () const
Returns the data portion of the net structure
-

Inherited from [DtPduWithData](#):

Public Methods

- virtual int **numFixedFields**() const
- virtual int **numVarFields**() const
- virtual [DtDatumParam](#) **varDatumId**(int index) const
- virtual [DtDatumParam](#) **fixedDatumId**(int index) const
- virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const
- virtual int **datumSizeBits**(int fixedOrVar, int index) const
- virtual int **datumSizeBytes**(int fixedOrVar, int index) const
- virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const
- virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual const char* **datumValByteArray**(int fixedOrVar, int index) const
- virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)
- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)
- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)

- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printDataSection**() const
- virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public

Protected Fields

- [DtDatumRecordSet](#)* myRecSet

Protected Methods

- virtual const [DtDatumRecordSet](#)* **datumRecSet**() const
- virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const
- virtual DtNetVarDatum* **nthVarDataRecord**(int index) const
- virtual void* **datumValPtr**(int fixedOrVar, int index) const

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)

virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

[DtEntityIdentifier](#) cache

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

virtual void* **packet**() const

virtual int **status**() const

virtual void **setVersion**(int vers)

virtual int **protocolVersion**() const

virtual void **setExerciseld**(int id)

virtual int **exerciseld**() const

virtual [DtPduKind](#) **kind**() const

virtual [DtProtocolFamily](#) **protocolFamily**() const

virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

virtual DtTime **timeStamp**() const

virtual [DtTimeStampType](#) **timeStampType**() const

virtual DtTime **guessTimeValid**(DtTime referenceTime) const

virtual void **print**() const

virtual void **printHeader**() const

inline int **exerciseld**() const

inline [DtPduKind](#) **kind**() const

inline DtNetPduHeader* **netHeader**() const

inline void* **packet**() const

inline [DtProtocolFamily](#) **protocolFamily**() const

inline int **protocolVersion**() const

inline void **setExerciseld**(int id)

- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**

• `int myMsgSize`

Protected Methods

- `void initMsg(int size, DtBufferPtr bufferPtr)`
 - `void initMsg(const void *initial, int size, DtBufferPtr bufferPtr)`
 - `virtual void changeSize(int size)`
 - `virtual void insertBytes(int offset, int numBytes)`
 - `virtual void insertBytes(void *address, int numBytes)`
 - `virtual void deleteBytes(int offset, int numBytes)`
 - `virtual void deleteBytes(void *address, int numBytes)`
 - `virtual void clearAll()`
 - `virtual void* newInternalBuffer(int size)`
-

Documentation

• `DtEventReptRPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

Constructors

• `DtEventReptRPdu(const DtNetEventReportPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

• `DtEventReptRPdu(const DtEventReptRPdu& orig)`

• `virtual ~DtEventReptRPdu()`

destructor

• `DtEventReptRPdu& operator=(const DtEventReptRPdu& orig)`

Assignment operator

• `virtual void setEventType(DtEventType type)`

Set/Get event type

• `virtual DtEventType eventType() const`

• `virtual void printData() const`

Print the data contained by this PDU

• `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

Create a DtEventReptRPdu. Caller is responsible for deletion.

• `static void addCallback(DtExerciseConn* conn, DtEventReptRPduCb cb, void* usr)`

Add (register)/Remove (deregister) function to be called when pdu is received

- `static void removeCallback(DtExerciseConn* conn, DtEventReptRPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetDataPduTail* dataSection() const`
Returns the data portion of the net structure
- `static int theStaticSize`
- `inline DtPduKind internalGetPduKind() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtEventType

-
-  DtEventOther
 -  DtEventOutOfFuel
 -  DtEventOutOfAmmo
 -  DtEventKilledInAction
 -  DtEventDamage
-

Documentation

-  DtEventOther
-  DtEventOutOfFuel
-  DtEventOutOfAmmo
-  DtEventKilledInAction
-  DtEventDamage

In file eventreptpdu.h:

class DtEventReportPdu: public DtPduWithData

Inheritance:

DtEventReportPdu - DtPduWithData - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtEventReportPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtEventReportPdu** (const DtNetEventReportPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** ([DtExerciseConn](#) *conn, DtEventReportPduCb cb, void *usr)
- static void **removeCallback** ([DtExerciseConn](#) *conn, DtEventReportPduCb cb, void *usr)
- [DtEventType](#) **eventType** () const
- void **setEventType** ([DtEventType](#) type)
- void **printData** () const

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
 - DtNetDataPduTail* **dataSection** () const
-

Inherited from DtPduWithData:

Public Methods

- virtual int **numFixedFields**() const
- virtual int **numVarFields**() const
- virtual [DtDatumParam](#) **varDatumId**(int index) const

- virtual [DtDatumParam](#) **fixedDatumId**(int index) const
- virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const
- virtual int **datumSizeBits**(int fixedOrVar, int index) const
- virtual int **datumSizeBytes**(int fixedOrVar, int index) const
- virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const
- virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual const char* **datumValByteArray**(int fixedOrVar, int index) const
- virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)
- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)
- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)
- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printDataSection**() const

virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public



Protected Fields

[DtDatumRecordSet](#)* **myRecSet**

Protected Methods

virtual const [DtDatumRecordSet](#)* **datumRecSet**() const

virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const

virtual DtNetVarDatum* **nthVarDataRecord**(int index) const

virtual void* **datumValPtr**(int fixedOrVar, int index) const

Inherited from [DtSimanPdu](#):

Public Methods

virtual const [DtEntityIdentifier](#)& **senderId**() const

virtual const [DtEntityIdentifier](#)& **receiverId**() const

virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)

virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

[DtEntityIdentifier](#) **cache**

Inherited from [DtInteraction](#):

Public Methods

[DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtEventReportPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `DtEventReportPdu(const DtNetEventReportPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static void addCallback(DtExerciseConn *conn, DtEventReportPduCb cb, void *usr)`
 - `static void removeCallback(DtExerciseConn *conn, DtEventReportPduCb cb, void *usr)`
 - `DtEventType eventType() const`
 - `void setEventType(DtEventType type)`
 - `void printData() const`
 - `DtPduKind internalGetPduKind() const`
 - `DtNetDataPduTail* dataSection() const`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file datarpdu.h:

class DtDataRPdu: public DtPduWithData

Inheritance:

DtDataRPdu - DtPduWithData - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

	DtDataRPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtDataRPdu (const DtNetDataRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtDataRPdu (const <u>DtDataRPdu</u> & orig)
 virtual	~DtDataRPdu () <i>destructor</i>
 <u>DtDataRPdu</u> &	operator= (const <u>DtDataRPdu</u> & orig) <i>Assignment operator</i>
 virtual void	setRequestId (unsigned long id) <i>Set/Get the request id</i>
 virtual unsigned long	requestId () const
 virtual void	setReqRelService (<u>DtRequiredReliabilityService</u> value) <i>Set/Get the required reliability service</i>
 virtual <u>DtRequiredReliabilityService</u>	reqRelService () const
 virtual void	printData () const <i>Print the data contained by this PDU</i>
 static <u>DtPdu</u> *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtDataRPdu</i>
 static void	addCallback (<u>DtExerciseConn</u> * conn, DtDataRPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
 static void	removeCallback (<u>DtExerciseConn</u> * conn, DtDataRPduCb cb, void* usr)
 inline <u>DtPduKind</u>	internalGetPduKind () const

Protected Fields

static int **theStaticSize**

Protected Methods

virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

virtual DtNetDataPduTail* **dataSection** () const
Returns the data portion of the net structure

Inherited from [DtPduWithData](#):

Public Methods

virtual int **numFixedFields**() const

virtual int **numVarFields**() const

virtual [DtDatumParam](#) **varDatumId**(int index) const

virtual [DtDatumParam](#) **fixedDatumId**(int index) const

virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const

virtual int **datumSizeBits**(int fixedOrVar, int index) const

virtual int **datumSizeBytes**(int fixedOrVar, int index) const

virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const

virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const

virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual const char* **datumValByteArray**(int fixedOrVar, int index) const

virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const

virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const

virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)

- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)
- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)
- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printDataSection**() const
- virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public



Protected Fields

- [DtDatumRecordSet](#)* myRecSet

Protected Methods

- virtual const [DtDatumRecordSet](#)* **datumRecSet**() const
 - virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const
 - virtual DtNetVarDatum* **nthVarDataRecord**(int index) const
 - virtual void* **datumValPtr**(int fixedOrVar, int index) const
-

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const

- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtDataRPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtDataRPdu(const DtNetDataRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtDataRPdu(const [DtDataRPdu](#)& orig)
- virtual ~DtDataRPdu()
destructor
- [DtDataRPdu](#)& operator=(const [DtDataRPdu](#)& orig)
Assignment operator

- `virtual void setRequestId(unsigned long id)`
Set/Get the request id
- `virtual unsigned long requestId() const`
- `virtual void setReqRelService(DtRequiredReliabilityService value)`
Set/Get the required reliability service
- `virtual DtRequiredReliabilityService reqRelService() const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtDataRPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtDataRPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtDataRPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetDataPduTail* dataSection() const`
Returns the data portion of the net structure
- `static int theStaticSize`
- `inline DtPduKind internalGetPduKind() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file datapdu.h:

class DtDataPdu: public DtPduWithData

Inheritance:

DtDataPdu - DtPduWithData - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtDataPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtDataPdu** (const DtNetDataPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static DtPdu* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** (DtExerciseConn *conn, DtDataPduCb cb, void *usr)
- static void **removeCallback** (DtExerciseConn *conn, DtDataPduCb cb, void *usr)
- unsigned long **requestId** () const
- void **setRequestId** (unsigned long id)
- void **printData** () const

Protected Methods

- DtPduKind **internalGetPduKind** () const
 - DtNetDataPduTail* **dataSection** () const
-

Inherited from DtPduWithData:

Public Methods

- virtual int **numFixedFields**() const
- virtual int **numVarFields**() const
- virtual DtDatumParam **varDatumId**(int index) const

- virtual [DtDatumParam](#) **fixedDatumId**(int index) const
- virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const
- virtual int **datumSizeBits**(int fixedOrVar, int index) const
- virtual int **datumSizeBytes**(int fixedOrVar, int index) const
- virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const
- virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual const char* **datumValByteArray**(int fixedOrVar, int index) const
- virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)
- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)
- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)
- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printDataSection**() const

virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public



Protected Fields

[DtDatumRecordSet](#)* **myRecSet**

Protected Methods

virtual const [DtDatumRecordSet](#)* **datumRecSet**() const

virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const

virtual DtNetVarDatum* **nthVarDataRecord**(int index) const

virtual void* **datumValPtr**(int fixedOrVar, int index) const

Inherited from [DtSimanPdu](#):

Public Methods

virtual const [DtEntityIdentifier](#)& **senderId**() const

virtual const [DtEntityIdentifier](#)& **receiverId**() const

virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)

virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

[DtEntityIdentifier](#) **cache**

Inherited from [DtInteraction](#):

Public Methods

[DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtDataPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtDataPdu(const DtNetDataPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - static void addCallback([DtExerciseConn](#) *conn, DtDataPduCb cb, void *usr)
 - static void removeCallback([DtExerciseConn](#) *conn, DtDataPduCb cb, void *usr)
 - unsigned long requestId() const
 - void setRequestId(unsigned long id)
 - void printData() const
 - [DtPduKind](#) internalGetPduKind() const
 - DtNetDataPduTail* dataSection() const
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file commentrpdu.h:

class DtCommentRPdu: public DtPduWithData

Inheritance:

DtCommentRPdu - DtPduWithData - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtCommentRPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- **DtCommentRPdu** (const DtNetCommentRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtCommentRPdu** (const [DtEntityIdentifier](#)& sender, const [DtEntityIdentifier](#)& receiver, const char* message, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtCommentRPdu** (const [DtCommentRPdu](#)& orig)
- virtual **~DtCommentRPdu** ()
destructor
- [DtCommentRPdu](#)& **operator=** (const [DtCommentRPdu](#)& orig)
Assignment operator
- virtual void **setComment** (const char* message, int size)
Set the comment
- static [DtPdu](#)* **create** (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Create a DtCommentRPdu
- static void **addCallback** ([DtExerciseConn](#)* conn, DtCommentRPduCb cb, void* usr)
Add (register)/Remove (deregister) function to be called when pdu is received
- static void **removeCallback** ([DtExerciseConn](#)* conn, DtCommentRPduCb cb, void* usr)
- inline [DtPduKind](#) **internalGetPduKind** () const

Protected Fields

- static int **theStaticSize**

Protected Methods

- void **setCommentInternal** (const char* message, int size)
Set the comment
 - virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind
 - virtual DtNetDataPduTail* **dataSection** () const
Returns the data portion of the pdu
-

Inherited from [DtPduWithData](#):

Public Methods

- virtual int **numFixedFields**() const
- virtual int **numVarFields**() const
- virtual [DtDatumParam](#) **varDatumId**(int index) const
- virtual [DtDatumParam](#) **fixedDatumId**(int index) const
- virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const
- virtual int **datumSizeBits**(int fixedOrVar, int index) const
- virtual int **datumSizeBytes**(int fixedOrVar, int index) const
- virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const
- virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual const char* **datumValByteArray**(int fixedOrVar, int index) const
- virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)
- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)

- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)
- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printData**() const
- virtual void **printDataSection**() const
- virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public



Protected Fields

- [DtDatumRecordSet](#)* myRecSet

Protected Methods

- virtual const [DtDatumRecordSet](#)* **datumRecSet**() const
- virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const
- virtual DtNetVarDatum* **nthVarDataRecord**(int index) const
- virtual void* **datumValPtr**(int fixedOrVar, int index) const

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const

- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtCommentRPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtCommentRPdu(const DtNetCommentRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtCommentRPdu(const [DtEntityIdentifier](#)& sender, const [DtEntityIdentifier](#)& receiver, const char* message, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtCommentRPdu(const [DtCommentRPdu](#)& orig)
- virtual ~DtCommentRPdu()
destructor
- [DtCommentRPdu](#)& operator=(const [DtCommentRPdu](#)& orig)
Assignment operator
- virtual void **setComment**(const char* message, int size)
Set the comment
- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtCommentRPdu. Caller is responsible for deletion.

- `static void addCallback(DtExerciseConn* conn, DtCommentRPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
 - `static void removeCallback(DtExerciseConn* conn, DtCommentRPduCb cb, void* usr)`
 - `void setCommentInternal(const char* message, int size)`
Set the comment. Non-virtual since called from ctor.
 - `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
 - `virtual DtNetDataPduTail* dataSection() const`
Returns the data portion of the pdu
 - `static int theStaticSize`
 - `inline DtPduKind internalGetPduKind() const`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file commentpdu.h:

class DtCommentPdu: public DtPduWithData

Inheritance:

DtCommentPdu - DtPduWithData - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtCommentPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtCommentPdu** (const DtNetCommentPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtCommentPdu** (const [DtEntityIdentifier](#) &sender, const [DtEntityIdentifier](#) &receiver, const char *message, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** ([DtExerciseConn](#) *conn, DtCommentPduCb cb, void *usr)
- static void **removeCallback** ([DtExerciseConn](#) *conn, DtCommentPduCb cb, void *usr)
- void **setComment** (const char *message, int size)
none besides those inherited from DtPduWithData

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
 - DtNetDataPduTail* **dataSection** () const
-

Inherited from DtPduWithData:

Public Methods

- virtual int **numFixedFields**() const
- virtual int **numVarFields**() const
- virtual [DtDatumParam](#) **varDatumId**(int index) const
- virtual [DtDatumParam](#) **fixedDatumId**(int index) const

- virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const
- virtual int **datumSizeBits**(int fixedOrVar, int index) const
- virtual int **datumSizeBytes**(int fixedOrVar, int index) const
- virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const
- virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual const char* **datumValByteArray**(int fixedOrVar, int index) const
- virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)
- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)
- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)
- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printData**() const
- virtual void **printDataSection**() const

virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public



Protected Fields

[DtDatumRecordSet](#)* **myRecSet**

Protected Methods

virtual const [DtDatumRecordSet](#)* **datumRecSet**() const

virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const

virtual DtNetVarDatum* **nthVarDataRecord**(int index) const

virtual void* **datumValPtr**(int fixedOrVar, int index) const

Inherited from [DtSimanPdu](#):

Public Methods

virtual const [DtEntityIdentifier](#)& **senderId**() const

virtual const [DtEntityIdentifier](#)& **receiverId**() const

virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)

virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

[DtEntityIdentifier](#) **cache**

Inherited from [DtInteraction](#):

Public Methods

[DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtCommentPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `DtCommentPdu(const DtNetCommentPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `DtCommentPdu(const DtEntityIdentifier &sender, const DtEntityIdentifier &receiver, const char *message, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static void addCallback(DtExerciseConn *conn, DtCommentPduCb cb, void *usr)`
 - `static void removeCallback(DtExerciseConn *conn, DtCommentPduCb cb, void *usr)`
 - `void setComment(const char *message, int size)`
none besides those inherited from DtPduWithData
 - `DtPduKind internalGetPduKind() const`
 - `DtNetDataPduTail* dataSection() const`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file actresrpdu.h:

class [DtActionResponseRPdu](#): public [DtPduWithData](#)

Inheritance:

[DtActionResponseRPdu](#) - [DtPduWithData](#) - [DtSimanPdu](#) - [DtInteraction](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtActionResponseRPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtActionResponseRPdu (const DtNetActionResponseRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtActionResponseRPdu (const DtActionResponseRPdu & orig)
 virtual	~DtActionResponseRPdu () <i>destructor</i>
 DtActionResponseRPdu &	operator= (const DtActionResponseRPdu & orig) <i>Assignment operator</i>
 virtual void	setRequestId (unsigned long id) <i>Set/Get the request id</i>
 virtual unsigned long	requestId () const
 virtual void	setRequestStatus (DtRequestStatus rStatus) <i>Set/Get the request status</i>
 virtual DtRequestStatus	requestStatus () const
 virtual void	printData () const <i>Print the data contained by this PDU</i>
 static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtActionResponseRPdu</i>
 static void	addCallback (DtExerciseConn * conn, DtActionResponseRPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
 static void	removeCallback (DtExerciseConn * conn, DtActionResponseRPduCb cb, void* usr)
 inline DtPduKind	internalGetPduKind () const

Protected Fields

static int **theStaticSize**

Protected Methods

virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

virtual DtNetDataPduTail* **dataSection** () const
Returns the data portion of the net structure

Inherited from [DtPduWithData](#):

Public Methods

virtual int **numFixedFields**() const

virtual int **numVarFields**() const

virtual [DtDatumParam](#) **varDatumId**(int index) const

virtual [DtDatumParam](#) **fixedDatumId**(int index) const

virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const

virtual int **datumSizeBits**(int fixedOrVar, int index) const

virtual int **datumSizeBytes**(int fixedOrVar, int index) const

virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const

virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const

virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual const char* **datumValByteArray**(int fixedOrVar, int index) const

virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const

virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const

virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)

- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)
- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)
- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printDataSection**() const
- virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public



Protected Fields

- [DtDatumRecordSet](#)* myRecSet

Protected Methods

- virtual const [DtDatumRecordSet](#)* **datumRecSet**() const
 - virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const
 - virtual DtNetVarDatum* **nthVarDataRecord**(int index) const
 - virtual void* **datumValPtr**(int fixedOrVar, int index) const
-

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const

- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtActionResponseRPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtActionResponseRPdu**(const DtNetActionResponseRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtActionResponseRPdu**(const [DtActionResponseRPdu](#)& orig)
- virtual ~DtActionResponseRPdu**()
destructor
- [DtActionResponseRPdu](#)& operator=(const [DtActionResponseRPdu](#)& orig)
Assignment operator

- virtual void setRequestId(unsigned long id)
Set/Get the request id
- virtual unsigned long requestId() const
- virtual void setRequestStatus([DtRequestStatus](#) rStatus)
Set/Get the request status
- virtual [DtRequestStatus](#) requestStatus() const
- virtual void printData() const
Print the data contained by this PDU
- static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Create a DtActionResponseRPdu. Caller is responsible for deletion.
- static void addCallback([DtExerciseConn](#)* conn, DtActionResponseRPduCb cb, void* usr)
Add (register)/Remove (deregister) function to be called when pdu is received
- static void removeCallback([DtExerciseConn](#)* conn, DtActionResponseRPduCb cb, void* usr)
- virtual [DtPduKind](#) internalGetPduKind() const
Returns the internal PDU kind
- virtual DtNetDataPduTail* dataSection() const
Returns the data portion of the net structure
- static int theStaticSize
- inline [DtPduKind](#) internalGetPduKind() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtRequestStatus

- DtRequestStatusOther
 - DtRequestStatusPending
 - DtRequestStatusExecuting
 - DtRequestStatusPartial
 - DtRequestStatusComplete
 - DtRequestStatusRejected
 - DtRequestStatusRetransmitNow
 - DtRequestStatusRetransmitLater
 - DtRequestStatusInvalidTimeParameters
 - DtRequestStatusSimulationTimeExceeded
 - DtRequestStatusRequestDone
-

Documentation

- DtRequestStatusOther
- DtRequestStatusPending
- DtRequestStatusExecuting
- DtRequestStatusPartial
- DtRequestStatusComplete
- DtRequestStatusRejected
- DtRequestStatusRetransmitNow
- DtRequestStatusRetransmitLater
- DtRequestStatusInvalidTimeParameters
- DtRequestStatusSimulationTimeExceeded
- DtRequestStatusRequestDone

In file actionpdu.h:

class DtActionResponsePdu: public DtPduWithData

Inheritance:

DtActionResponsePdu - DtPduWithData - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtActionResponsePdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtActionResponsePdu** (const DtNetActionResponsePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static DtPdu* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** (DtExerciseConn *conn, DtActionResponseInteractionCb cb, void *usr)
- static void **removeCallback** (DtExerciseConn *conn, DtActionResponseInteractionCb cb, void *usr)
- unsigned long **requestId** () const
- DtRequestStatus **requestStatus** () const
- void **setRequestId** (unsigned long id)
- void **setRequestStatus** (DtRequestStatus rStatus)
- void **printData** () const

Protected Methods

- DtPduKind **internalGetPduKind** () const
 - DtNetDataPduTail* **dataSection** () const
-

Inherited from [DtPduWithData](#):

Public Methods

- virtual int **numFixedFields**() const
- virtual int **numVarFields**() const
- virtual [DtDatumParam](#) **varDatumId**(int index) const
- virtual [DtDatumParam](#) **fixedDatumId**(int index) const
- virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const
- virtual int **datumSizeBits**(int fixedOrVar, int index) const
- virtual int **datumSizeBytes**(int fixedOrVar, int index) const
- virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const
- virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual const char* **datumValByteArray**(int fixedOrVar, int index) const
- virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)
- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)
- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)
- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)

- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printDataSection**() const
- virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public

Protected Fields

- [DtDatumRecordSet](#)* myRecSet

Protected Methods

- virtual const [DtDatumRecordSet](#)* **datumRecSet**() const
 - virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const
 - virtual DtNetVarDatum* **nthVarDataRecord**(int index) const
 - virtual void* **datumValPtr**(int fixedOrVar, int index) const
-

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Public Methods

• [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)

- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtActionResponsePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtActionResponsePdu (const DtNetActionResponsePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - static [DtPdu](#)* create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - static void addCallback ([DtExerciseConn](#) *conn, DtActionResponseInteractionCb cb, void *usr)
 - static void removeCallback ([DtExerciseConn](#) *conn, DtActionResponseInteractionCb cb, void *usr)
 - unsigned long requestId() const
 - [DtRequestStatus](#) requestStatus() const
 - void setRequestId(unsigned long id)
 - void setRequestStatus ([DtRequestStatus](#) rStatus)
 - void printData() const
 - [DtPduKind](#) internalGetPduKind() const
 - DtNetDataPduTail* dataSection() const
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file actreqrpdu.h:

class DtActionRequestRPdu: public DtPduWithData

Inheritance:

**DtActionRequestRPdu - DtPduWithData - DtSimanPdu - DtInteraction
- DtPdu - DtBaseMsg**

Public Methods

	DtActionRequestRPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtActionRequestRPdu (const DtNetActionRequestRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtActionRequestRPdu (const DtActionRequestRPdu & orig)
 virtual	~DtActionRequestRPdu () <i>destructor</i>
 DtActionRequestRPdu &	operator= (const DtActionRequestRPdu & orig) <i>Assignment operator</i>
 virtual void	setRequestId (unsigned long id) <i>Set/Get the request id</i>
 virtual unsigned long	requestId () const
 virtual void	setActionId (unsigned long id) <i>Set/Get the action id</i>
 virtual unsigned long	actionId () const
 virtual void	setReqRelService (DtRequiredReliabilityService value) <i>Set/Get the required reliability service</i>
 virtual DtRequiredReliabilityService	reqRelService () const
 virtual void	printData () const <i>Print the data contained by this PDU</i>
 static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtActionRequestRPdu</i>
 static void	addCallback (DtExerciseConn * conn, DtActionRequestRPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
 static void	removeCallback (DtExerciseConn * conn, DtActionRequestRPduCb cb, void* usr)
 inline DtPduKind	internalGetPduKind () const

Protected Fields

static int **theStaticSize**

Protected Methods

virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

virtual DtNetDataPduTail* **dataSection** () const
Returns the data portion of the PDU

Inherited from [DtPduWithData](#):

Public Methods

virtual int **numFixedFields**() const

virtual int **numVarFields**() const

virtual [DtDatumParam](#) **varDatumId**(int index) const

virtual [DtDatumParam](#) **fixedDatumId**(int index) const

virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const

virtual int **datumSizeBits**(int fixedOrVar, int index) const

virtual int **datumSizeBytes**(int fixedOrVar, int index) const

virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const

virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const

virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const

virtual const char* **datumValByteArray**(int fixedOrVar, int index) const

virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const

virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const

virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const

virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)

- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)
- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)
- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printDataSection**() const
- virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public



Protected Fields

- [DtDatumRecordSet](#)* myRecSet

Protected Methods

- virtual const [DtDatumRecordSet](#)* **datumRecSet**() const
 - virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const
 - virtual DtNetVarDatum* **nthVarDataRecord**(int index) const
 - virtual void* **datumValPtr**(int fixedOrVar, int index) const
-

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) cache
-

Inherited from [DtInteraction](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExercisId**(int id)
- virtual int **exercisId**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const

- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtActionRequestRPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtActionRequestRPdu**(const DtNetActionRequestRPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtActionRequestRPdu**(const [DtActionRequestRPdu](#)& orig)
- virtual ~DtActionRequestRPdu**()
destructor
- [DtActionRequestRPdu](#)& **operator=**(const [DtActionRequestRPdu](#)& orig)
Assignment operator

- `virtual void setRequestId(unsigned long id)`
Set/Get the request id
- `virtual unsigned long requestId() const`
- `virtual void setActionId(unsigned long id)`
Set/Get the action id
- `virtual unsigned long actionId() const`
- `virtual void setReqRelService(DtRequiredReliabilityService value)`
Set/Get the required reliability service
- `virtual DtRequiredReliabilityService reqRelService() const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtActionRequestRPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtActionRequestRPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtActionRequestRPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetDataPduTail* dataSection() const`
Returns the data portion of the PDU
- `static int theStaticSize`
- `inline DtPduKind internalGetPduKind() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file actionpdu.h:

class DtActionRequestPdu: public DtPduWithData

Inheritance:

DtActionRequestPdu - DtPduWithData - DtSimanPdu - DtInteraction - DtPdu - DtBaseMsg

Public Methods

- **DtActionRequestPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtActionRequestPdu** (const DtNetActionRequestPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** ([DtExerciseConn](#) *conn, DtActionRequestInteractionCb cb, void *usr)
- static void **removeCallback** ([DtExerciseConn](#) *conn, DtActionRequestInteractionCb cb, void *usr)
- unsigned long **requestId** () const
- unsigned long **actionId** () const
- void **setRequestId** (unsigned long id)
- void **setActionId** (unsigned long id)
- void **printData** () const

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
 - DtNetDataPduTail* **dataSection** () const
-

Inherited from [DtPduWithData](#):

Public Methods

- virtual int **numFixedFields**() const
- virtual int **numVarFields**() const
- virtual [DtDatumParam](#) **varDatumId**(int index) const
- virtual [DtDatumParam](#) **fixedDatumId**(int index) const
- virtual [DtDatumParam](#) **datumId**(int fixedOrVar, int index) const
- virtual int **datumSizeBits**(int fixedOrVar, int index) const
- virtual int **datumSizeBytes**(int fixedOrVar, int index) const
- virtual void **getIndex**([DtDatumParam](#) param, int *fixedOrVar, int *index, int *retcode = NULL) const
- virtual DtFloat32 **datumValFloat32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual long **datumValInt32**(int fixedOrVar, int index, int *retcode = NULL) const
- virtual const char* **datumValByteArray**(int fixedOrVar, int index) const
- virtual DtFloat32 **datumValFloat32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual DtFloat64 **datumValFloat64**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual unsigned long **datumValUnsigned32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual long **datumValInt32**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual const char* **datumValByteArray**([DtDatumParam](#) param, int *retcode = NULL) const
- virtual void **initDataSection**(const [DtDatumParam](#) *fixedParams, const [DtVarDatumSpec](#) *specList)
- virtual void **setNumFixedFields**(int newNum)
- virtual void **setNumVarFields**(int newNum)
- virtual void **setDatumParam**(int fixedOrVar, int index, [DtDatumParam](#) param, int *retcode = NULL)
- virtual void **setVarDataBytes**(int index, int length, int *retcode = NULL)
- virtual void **setDatumValFloat64**(int fixedOrVar, int index, DtFloat64 val, int *retcode = NULL)
- virtual void **setDatumValFloat32**(int fixedOrVar, int index, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**(int fixedOrVar, int index, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**(int fixedOrVar, int index, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**(int fixedOrVar, int index, const char *val, int *retcode = NULL)
- virtual void **setDatumValFloat64**([DtDatumParam](#) param, DtFloat64 val, int *retcode = NULL)

- virtual void **setDatumValFloat32**([DtDatumParam](#) param, DtFloat32 val, int *retcode = NULL)
- virtual void **setDatumValInt32**([DtDatumParam](#) param, long val, int *retcode = NULL)
- virtual void **setDatumValUnsigned32**([DtDatumParam](#) param, unsigned long val, int *retcode = NULL)
- virtual void **setDatumValByteArray**([DtDatumParam](#) param, const char *val, int *retcode = NULL)
- virtual void **printDataSection**() const
- virtual [DtDatumRecordSet](#)* **datumRecSet**()

Public



Protected Fields

- [DtDatumRecordSet](#)* **myRecSet**

Protected Methods

- virtual const [DtDatumRecordSet](#)* **datumRecSet**() const
 - virtual DtNetFixDatum* **nthFixedDataRecord**(int index) const
 - virtual DtNetVarDatum* **nthVarDataRecord**(int index) const
 - virtual void* **datumValPtr**(int fixedOrVar, int index) const
-

Inherited from [DtSimanPdu](#):

Public Methods

- virtual const [DtEntityIdentifier](#)& **senderId**() const
- virtual const [DtEntityIdentifier](#)& **receiverId**() const
- virtual void **setSenderId**(const [DtEntityIdentifier](#) &id)
- virtual void **setReceiverId**(const [DtEntityIdentifier](#) &id)

Protected Fields

- [DtEntityIdentifier](#) **cache**
-

Inherited from [DtInteraction](#):

Public Methods

• [DtInteraction](#)& **operator=**(const [DtInteraction](#)& orig)

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)

- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtActionRequestPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - DtActionRequestPdu(const DtNetActionRequestPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 - static void addCallback([DtExerciseConn](#) *conn, DtActionRequestInteractionCb cb, void *usr)
 - static void removeCallback([DtExerciseConn](#) *conn, DtActionRequestInteractionCb cb, void *usr)
 - unsigned long requestId() const
 - unsigned long actionId() const
 - void setRequestId(unsigned long id)
 - void setActionId(unsigned long id)
 - void printData() const
 - [DtPduKind](#) internalGetPduKind() const
 - DtNetDataPduTail* dataSection() const
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file unknownpdu.h:

class DtUnknownPdu: public DtPdu

Inheritance:

DtUnknownPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- **DtUnknownPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtUnknownPdu** (const DtNetPduHeader *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** ([DtPduKind](#) kind, [DtExerciseConn](#) *conn, DtUnknownPduCb cb, void *usr)
- static void **removeCallback** ([DtPduKind](#) kind, [DtExerciseConn](#) *conn, DtUnknownPduCb cb, void *usr)
- void **setKind** ([DtPduKind](#) kind)

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
-

Inherited from [DtPdu](#):

Public Methods

- [DtPdu](#)& **operator=**(const [DtPdu](#) &pdu)
- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const

- virtual [DtPduKind](#) **kind()** const
- virtual [DtProtocolFamily](#) **protocolFamily()** const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp()** const
- virtual [DtTimeStampType](#) **timeStampType()** const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print()** const
- virtual void **printHeader()** const
- virtual void **printData()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader()** const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)

- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtUnknownPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `DtUnknownPdu(const DtNetPduHeader *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static void addCallback(DtPduKind kind, DtExerciseConn *conn, DtUnknownPduCb cb, void *usr)`
 - `static void removeCallback(DtPduKind kind, DtExerciseConn *conn, DtUnknownPduCb cb, void *usr)`
 - `void setKind(DtPduKind kind)`
 - `DtPduKind internalGetPduKind() const`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file undwaacpdu.h:

class DtUnderwaterAcousticPdu: public DtPdu

Inheritance:

DtUnderwaterAcousticPdu - DtPdu - DtBaseMsg

Public Methods

	DtUnderwaterAcousticPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtUnderwaterAcousticPdu (const DtNetUnderwaterAcousticPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtUnderwaterAcousticPdu (const DtUnderwaterAcousticPdu & orig)
 virtual	~DtUnderwaterAcousticPdu () <i>destructor</i>
 DtUnderwaterAcousticPdu &	operator= (const DtUnderwaterAcousticPdu & orig) <i>Assignment operator</i>
 virtual void	setEmittingEntityID (const DtEntityIdentifier & entity) <i>Set/Get emitting entity identifier</i>
 virtual const DtEntityIdentifier &	emittingEntityID () const
 virtual void	setEventID (const DtEventID& event) <i>Set/Get event identifier</i>
 virtual const DtEventID&	eventID () const
 virtual void	setStateChangeUpdate (DtUnderwaterAcousticUpdate upd) <i>Set/Get state/change update indicator</i>
 virtual DtUnderwaterAcousticUpdate	stateChangeUpdate () const
 virtual void	setPassiveParameterIndex (DtUnderwaterAcousticPassiveParmIndex ind) <i>Set/Get passive parameter index</i>
 virtual DtUnderwaterAcousticPassiveParmIndex	passiveParameterIndex () const
 virtual void	setPropulsionPlantConfig (DtUnderwaterAcousticPropulsionPlantConfig config) <i>Set/Get propulsion plant configuration</i>
 virtual DtUnderwaterAcousticPropulsionPlantConfig	propulsionPlantConfig () const

virtual [DtBoolean](#)

virtual DtU8

virtual [DtBoolean](#)

virtual DtU8

virtual [DtBoolean](#)

virtual DtU8

virtual [DtBoolean](#)

virtual DtU8

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual void

static [DtPdu](#)*

static void

static void

setNumShafts (DtU8 num)

Set/Get number of shafts May return DtFalse if PDU size would exceed the maximum PDU size

numShafts () const

setNumAdditionalPassiveAct (DtU8 num)

Set/Get number of additional passive activities May return DtFalse if PDU size would exceed the maximum PDU size

numAdditionalPassiveAct () const

setNumEmitterSystems (DtU8 num)

Set/Get number of emitter systems May return DtFalse if PDU size would exceed the maximum PDU size

numEmitterSystems () const

setNumBeams (int emitterSystemIndex, DtU8 num)

Set/Get number of beam data records for a particular emitter system

numBeams (int emitterSystemIndex) const

setShaftSpeed (int index, const

[DtUnderwaterAcousticShaftSpeed](#)& record)

Set/Get a shaft speed data record

getShaftSpeed (int index, [DtUnderwaterAcousticShaftSpeed](#)& record) const

setAddtnlPassiveAct (int index, const

[DtUnderwaterAcousticAddtnlPassiveAct](#)& record)

Set/Get an additional passive activity data record

getAddtnlPassiveAct (int index,

[DtUnderwaterAcousticAddtnlPassiveAct](#)& record) const

setEmitterSystem (int index, const

[DtUnderwaterAcousticEmitterSystem](#)& record)

Set/Get an underwater acoustic emitter system

getEmitterSystem (int index,

[DtUnderwaterAcousticEmitterSystem](#)& record) const

setBeam (int emitterSystemIndex, int beamIndex, const

[DtUnderwaterAcousticBeam](#)& record)

Set/Get beam fundamental data parameters for a beam

getBeam (int emitterSystemIndex, int beamIndex,

[DtUnderwaterAcousticBeam](#)& record) const

printData () const

Print the data contained by this PDU

create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtUnderwaterAcousticPdu

addCallback ([DtExerciseConn](#)* conn,

DtUnderwaterAcousticPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

removeCallback ([DtExerciseConn](#)* conn,

DtUnderwaterAcousticPduCb cb, void* usr)

Protected Fields

- static int **theStaticSize**
Size of PDU with zero parameter records
- [DtEntityIdentifier](#) **myEntityID**
Temporary objects for reference returns
- DtEventID **myEventID**

Protected Methods

- virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind
 - virtual DtNetUnderwaterAcousticPdu* **netPduStruct** () const
Casts the netPdu element to our network struct
 - virtual char* **var** () const
Returns a pointer to the first byte of variable length data
 - [DtUnderwaterAcousticPdu](#)* **self** () const
Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
 - virtual int **shaftSpeedRecordOffset** (int recordIndex) const
Returns a buffer offset to the given shaft speed record
 - virtual int **passiveActRecordOffset** (int recordIndex) const
Returns a buffer offset to the given additional passive activity record
 - virtual int **emitterSystemRecordOffset** (int recordIndex) const
Returns a buffer offset to the given acoustic emitter system
 - virtual int **beamDataRecordOffset** (int emitterSystemIndex, int beamIndex) const
Returns a buffer offset to the given beam fundamental data parameter of a particular emitter system
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const

- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtUnderwaterAcousticPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtUnderwaterAcousticPdu**(const DtNetUnderwaterAcousticPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtUnderwaterAcousticPdu**(const [DtUnderwaterAcousticPdu](#)& orig)
- virtual ~DtUnderwaterAcousticPdu**()
destructor

- [DtUnderwaterAcousticPdu](#)& operator=(const [DtUnderwaterAcousticPdu](#)& orig)
 Assignment operator
- virtual void setEmittingEntityID(const [DtEntityIdentifier](#)& entity)
 Set/Get emitting entity identifier
- virtual const [DtEntityIdentifier](#)& emittingEntityID() const
- virtual void setEventID(const DtEventID& event)
 Set/Get event identifier
- virtual const DtEventID& eventID() const
- virtual void setStateChangeUpdate([DtUnderwaterAcousticUpdate](#) upd)
 Set/Get state/change update indicator
- virtual [DtUnderwaterAcousticUpdate](#) stateChangeUpdate() const
- virtual void setPassiveParameterIndex([DtUnderwaterAcousticPassiveParmIndex](#) ind)
 Set/Get passive parameter index
- virtual [DtUnderwaterAcousticPassiveParmIndex](#) passiveParameterIndex() const
- virtual void setPropulsionPlantConfig([DtUnderwaterAcousticPropulsionPlantConfig](#) config)
 Set/Get propulsion plant configuration
- virtual [DtUnderwaterAcousticPropulsionPlantConfig](#) propulsionPlantConfig() const
- virtual [DtBoolean](#) setNumShafts(DtU8 num)
 Set/Get number of shafts May return DtFalse if PDU size would exceed the maximum PDU size
- virtual DtU8 numShafts() const
- virtual [DtBoolean](#) setNumAdditionalPassiveAct(DtU8 num)
 Set/Get number of additional passive activities May return DtFalse if PDU size would exceed the maximum PDU size
- virtual DtU8 numAdditionalPassiveAct() const
- virtual [DtBoolean](#) setNumEmitterSystems(DtU8 num)
 Set/Get number of emitter systems May return DtFalse if PDU size would exceed the maximum PDU size
- virtual DtU8 numEmitterSystems() const
- virtual [DtBoolean](#) setNumBeams(int emitterSystemIndex, DtU8 num)
 Set/Get number of beam data records for a particular emitter system. May return DtFalse if PDU size would exceed the maximum PDU size or if the emitter system number is invalid.
- virtual DtU8 numBeams(int emitterSystemIndex) const
- virtual [DtBoolean](#) setShaftSpeed(int index, const [DtUnderwaterAcousticShaftSpeed](#)& record)
 Set/Get a shaft speed data record. Returns false if the record does not exist.

virtual [DtBoolean](#) getShaftSpeed(int index, [DtUnderwaterAcousticShaftSpeed&](#) record) const

virtual [DtBoolean](#) setAddtnlPassiveAct(int index, const [DtUnderwaterAcousticAddtnlPassiveAct&](#) record)

Set/Get an additional passive activity data record. Returns false if the record does not exist.

virtual [DtBoolean](#) getAddtnlPassiveAct(int index, [DtUnderwaterAcousticAddtnlPassiveAct&](#) record) const

virtual [DtBoolean](#) setEmitterSystem(int index, const [DtUnderwaterAcousticEmitterSystem&](#) record)

Set/Get an underwater acoustic emitter system. Returns false if the record does not exist. Data length field is automatically set by the PDU class.

virtual [DtBoolean](#) getEmitterSystem(int index, [DtUnderwaterAcousticEmitterSystem&](#) record) const

virtual [DtBoolean](#) setBeam(int emitterSystemIndex, int beamIndex, const [DtUnderwaterAcousticBeam&](#) record)

Set/Get beam fundamental data parameters for a beam. Returns false if the record does not exist. Data length field is automatically set by the PDU class.

virtual [DtBoolean](#) getBeam(int emitterSystemIndex, int beamIndex, [DtUnderwaterAcousticBeam&](#) record) const

virtual void printData() const

Print the data contained by this PDU

static [DtPdu*](#) create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtUnderwaterAcousticPdu. Caller is responsible for deletion.

static void addCallback([DtExerciseConn*](#) conn, DtUnderwaterAcousticPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

static void removeCallback([DtExerciseConn*](#) conn, DtUnderwaterAcousticPduCb cb, void* usr)

virtual [DtPduKind](#) internalGetPduKind() const

Returns the internal PDU kind

virtual DtNetUnderwaterAcousticPdu* netPduStruct() const

Casts the netPdu element to our network struct

virtual char* var() const

Returns a pointer to the first byte of variable length data

[DtUnderwaterAcousticPdu*](#) self() const

Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type

virtual int shaftSpeedRecordOffset(int recordIndex) const

Returns a buffer offset to the given shaft speed record

- `virtual int passiveActRecordOffset(int recordIndex) const`
Returns a buffer offset to the given additional passive activity record
 - `virtual int emitterSystemRecordOffset(int recordIndex) const`
Returns a buffer offset to the given acoustic emitter system
 - `virtual int beamDataRecordOffset( int emitterSystemIndex, int beamIndex) const`
Returns a buffer offset to the given beam fundamental data parameter of a particular emitter system
 - `static int theStaticSize`
Size of PDU with zero parameter records
 - `DtEntityIdentifier myEntityID`
Temporary objects for reference returns
 - `DtEventID myEventID`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtUnderwaterAcousticUpdate

Underwater accoustic PDU enums

-  **DtUnderwaterAcousticStateUpdate**
 -  **DtUnderwaterAcousticChangedDataUpdate**
-

Documentation

Underwater accoustic PDU enums

-  **DtUnderwaterAcousticStateUpdate**
-  **DtUnderwaterAcousticChangedDataUpdate**

In file disenums.h:

enum DtUnderwaterAcousticPassiveParmIndex

 DtUnderwaterAcousticOtherPassiveIndex

Documentation

 DtUnderwaterAcousticOtherPassiveIndex

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtUnderwaterAcousticPropulsionPlantConfig

-  DtUnderwaterAcousticOtherConfig
 -  DtUnderwaterAcousticDiesel_electric
 -  DtUnderwaterAcousticDiesel
 -  DtUnderwaterAcousticBattery
 -  DtUnderwaterAcousticNuclear
 -  DtUnderwaterAcousticTurbineReduction
 -  DtUnderwaterAcousticSteam
 -  DtUnderwaterAcousticGasTurbine
-

Documentation

-  DtUnderwaterAcousticOtherConfig
-  DtUnderwaterAcousticDiesel_electric
-  DtUnderwaterAcousticDiesel
-  DtUnderwaterAcousticBattery
-  DtUnderwaterAcousticNuclear
-  DtUnderwaterAcousticTurbineReduction
-  DtUnderwaterAcousticSteam
-  DtUnderwaterAcousticGasTurbine

In file undwaacshaft.h:

class DtUnderwaterAcousticShaftSpeed

class DtUnderwaterAcousticShaftSpeed: DtUnderwaterAcousticShaftSpeed is a class which contains the shaft speed information for the underwater acoustics PDU

Public Methods

	DtUnderwaterAcousticShaftSpeed () <i>Constructors</i>
	DtUnderwaterAcousticShaftSpeed (const DtUnderwaterAcousticShaftSpeed & orig)
DtUnderwaterAcousticShaftSpeed &	operator= (const DtUnderwaterAcousticShaftSpeed & orig) <i>Assignment operator</i>
DtUnderwaterAcousticShaftSpeed &	operator= (const DtNetUnderwaterAcousticShaftSpeed& orig)
virtual	~DtUnderwaterAcousticShaftSpeed () <i>Destructor</i>
virtual void	setCurrentShaftRPM (DtU16 value) <i>Set/Get the current shaft RPM</i>
virtual DtU16	currentShaftRPM () const
virtual void	setOrderedShaftRPM (DtU16 value) <i>Set/Get the ordered shaft RPM</i>
virtual DtU16	orderedShaftRPM () const
virtual void	setShaftRPM_RateOfChange (DtU32 value) <i>Set/Get the shaft RPM rate of change</i>
virtual DtU32	shaftRPM_RateOfChange () const
virtual void	print () const <i>Print the information contained in the record</i>
virtual void	unloadNetRecord (const DtNetUnderwaterAcousticShaftSpeed& record) <i>Load/Unload data from a packet record</i>
virtual void	loadNetRecord (DtNetUnderwaterAcousticShaftSpeed& record) const

Protected Fields

DtU16	myCurrentShaftRPM <i>Data members:</i>
DtU16	myOrderedShaftRPM
DtU32	myShaftRPM_RateOfChange

Documentation

class DtUnderwaterAcousticShaftSpeed: DtUnderwaterAcousticShaftSpeed is a class which contains the shaft speed information for the underwater acoustics PDU

`DtUnderwaterAcousticShaftSpeed()`

Constructors

`DtUnderwaterAcousticShaftSpeed(const DtUnderwaterAcousticShaftSpeed& orig)`

`DtUnderwaterAcousticShaftSpeed& operator=( const DtUnderwaterAcousticShaftSpeed& orig)`

Assignment operator

`DtUnderwaterAcousticShaftSpeed& operator=( const DtNetUnderwaterAcousticShaftSpeed& orig)`

`virtual ~DtUnderwaterAcousticShaftSpeed()`

Destructor

`virtual void setCurrentShaftRPM(DtU16 value)`

Set/Get the current shaft RPM

`virtual DtU16 currentShaftRPM() const`

`virtual void setOrderedShaftRPM(DtU16 value)`

Set/Get the ordered shaft RPM

`virtual DtU16 orderedShaftRPM() const`

`virtual void setShaftRPM_RateOfChange(DtU32 value)`

Set/Get the shaft RPM rate of change

`virtual DtU32 shaftRPM_RateOfChange() const`

`virtual void print() const`

Print the information contained in the record

`virtual void unloadNetRecord( const DtNetUnderwaterAcousticShaftSpeed& record)`

Load/Unload data from a packet record

`virtual void loadNetRecord( DtNetUnderwaterAcousticShaftSpeed& record) const`

`DtU16 myCurrentShaftRPM`

Data members:

`DtU16 myOrderedShaftRPM`

`DtU32 myShaftRPM_RateOfChange`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file undwaacpasact.h:

class DtUnderwaterAcousticAddtnIPassiveAct

class DtUnderwaterAcousticAddtnIPassiveAct: DtUnderwaterAcousticAddtnIPassiveAct is a class which contains the additional passive activity information for the underwater acoustics PDU

Public Methods

	DtUnderwaterAcousticAddtnIPassiveAct () <i>Constructors</i>
	DtUnderwaterAcousticAddtnIPassiveAct (const DtUnderwaterAcousticAddtnIPassiveAct & orig)
 DtUnderwaterAcousticAddtnIPassiveAct &	operator= (const DtUnderwaterAcousticAddtnIPassiveAct & orig) <i>Assignment operator</i>
 DtUnderwaterAcousticAddtnIPassiveAct &	operator= (const DtNetUnderwaterAcousticAddtnIPassiveAct& orig)
 virtual	~DtUnderwaterAcousticAddtnIPassiveAct () <i>Destructor</i>
 virtual void	setParmIndex (DtUnderwaterAcousticAdditionalPassiveActivityParmIndex value) <i>Set/Get the parameter index</i>
 virtual DtUnderwaterAcousticAdditionalPassiveActivityParmIndex	parmIndex () const
 virtual void	setAPA (DtInt16 value) <i>Set/Get the APA value</i>
 virtual DtInt16	APA () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetUnderwaterAcousticAddtnIPassiveAct& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetUnderwaterAcousticAddtnIPassiveAct& record) const

Protected Fields

 DtUnderwaterAcousticPassiveParmIndex	myParmIndex <i>Data members:</i>
 DtU16	myAPA

Documentation

class DtUnderwaterAcousticAddtnlPassiveAct: DtUnderwaterAcousticAddtnlPassiveAct is a class which contains the additional passive activity information for the underwater acoustics PDU

`DtUnderwaterAcousticAddtnlPassiveAct()`

Constructors

`DtUnderwaterAcousticAddtnlPassiveAct( const DtUnderwaterAcousticAddtnlPassiveAct& orig)`

`DtUnderwaterAcousticAddtnlPassiveAct& operator=( const DtUnderwaterAcousticAddtnlPassiveAct& orig)`

Assignment operator

`DtUnderwaterAcousticAddtnlPassiveAct& operator=( const DtNetUnderwaterAcousticAddtnlPassiveAct& orig)`

`virtual ~DtUnderwaterAcousticAddtnlPassiveAct()`

Destructor

`virtual void setParmIndex( DtUnderwaterAcousticAdditionalPassiveActivityParmIndex value)`

Set/Get the parameter index

`virtual DtUnderwaterAcousticAdditionalPassiveActivityParmIndex parmIndex() const`

`virtual void setAPA(DtInt16 value)`

Set/Get the APA value

`virtual DtInt16 APA() const`

`virtual void print() const`

Print the information contained in the record

`virtual void unloadNetRecord( const DtNetUnderwaterAcousticAddtnlPassiveAct& record)`

Load/Unload data from a packet record

`virtual void loadNetRecord( DtNetUnderwaterAcousticAddtnlPassiveAct& record) const`

`DtUnderwaterAcousticPassiveParmIndex myParmIndex`

Data members:

`DtU16 myAPA`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum

DtUnderwaterAcousticAdditionalPassiveActivityParmIndex

-
-  DtUnderwaterAcousticOtherAdditionalIndex
 -  DtUnderwaterAcousticGrahamsMKV_CoffeeMaker
-

Documentation

-  DtUnderwaterAcousticOtherAdditionalIndex
-  DtUnderwaterAcousticGrahamsMKV_CoffeeMaker

In file undwaacemtsys.h:

class DtUnderwaterAcousticEmitterSystem

class DtUnderwaterAcousticEmitterSystem: DtUnderwaterAcousticEmitterSystem is a class which contains information describing the acoustic emitter systems in an underwater acoustics PDU

Public Methods

	DtUnderwaterAcousticEmitterSystem () <i>Constructors</i>
	DtUnderwaterAcousticEmitterSystem (const DtUnderwaterAcousticEmitterSystem & orig)
 DtUnderwaterAcousticEmitterSystem &	operator= (const DtUnderwaterAcousticEmitterSystem & orig) <i>Assignment operator</i>
 DtUnderwaterAcousticEmitterSystem &	operator= (const DtNetUnderwaterAcousticEmitterSystem& orig)
 virtual	~DtUnderwaterAcousticEmitterSystem () <i>Destructor</i>
 virtual void	setDataLength (DtU8 value) <i>Set/Get the data length</i>
 virtual DtU8	dataLength () const
 virtual void	setNumberOfBeams (DtU8 value) <i>Set/Get the number of beams</i>
 virtual DtU8	numberOfBeams () const
 virtual void	setAcousticName (DtUnderwaterAcousticSystemName name) <i>Set/Get the system name</i>
 virtual DtUnderwaterAcousticSystemName	acousticName () const
 virtual void	setFunction (DtUnderwaterAcousticFunction function) <i>Set/Get the function</i>
 virtual DtUnderwaterAcousticFunction	function () const
 virtual void	setAcousticID (DtU8 id) <i>Set/Get the acoustic ID number</i>
 virtual DtU8	acousticID () const
 virtual void	setLocation (DtConstVector coords) <i>Set/Get the location</i>
 virtual DtConstVector	location () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetUnderwaterAcousticEmitterSystem& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetUnderwaterAcousticEmitterSystem& record) const

Protected Fields

 DtU8	myDataLength <i>Data members:</i>
 DtU8	myNumberOfBeams
 DtUnderwaterAcousticSystemName	myAcousticName
 DtUnderwaterAcousticFunction	myFunction
 DtU8	myAcousticID
 DtVector	myLocation

Documentation

class DtUnderwaterAcousticEmitterSystem: DtUnderwaterAcousticEmitterSystem is a class which contains information describing the acoustic emitter systems in an underwater acoustics PDU

-  **DtUnderwaterAcousticEmitterSystem()**
Constructors
-  **DtUnderwaterAcousticEmitterSystem(const [DtUnderwaterAcousticEmitterSystem](#)& orig)**
-  **[DtUnderwaterAcousticEmitterSystem](#)& operator=(const [DtUnderwaterAcousticEmitterSystem](#)& orig)**
Assignment operator
-  **[DtUnderwaterAcousticEmitterSystem](#)& operator=(const DtNetUnderwaterAcousticEmitterSystem& orig)**
-  **virtual ~DtUnderwaterAcousticEmitterSystem()**
Destructor
-  **virtual void setDataLength(DtU8 value)**
Set/Get the data length
-  **virtual DtU8 dataLength() const**
-  **virtual void setNumberOfBeams(DtU8 value)**
Set/Get the number of beams
-  **virtual DtU8 numberOfBeams() const**
-  **virtual void setAcousticName([DtUnderwaterAcousticSystemName](#) name)**
Set/Get the system name
-  **virtual [DtUnderwaterAcousticSystemName](#) acousticName() const**
-  **virtual void setFunction([DtUnderwaterAcousticFunction](#) function)**
Set/Get the function
-  **virtual [DtUnderwaterAcousticFunction](#) function() const**

- `virtual void setAcousticID(DtU8 id)`
Set/Get the acoustic ID number
- `virtual DtU8 acousticID() const`
- `virtual void setLocation(DtConstVector coords)`
Set/Get the location
- `virtual DtConstVector location() const`
- `virtual void print() const`
Print the information contained in the record
- `virtual void unloadNetRecord( const DtNetUnderwaterAcousticEmitterSystem& record)`
Load/Unload data from a packet record
- `virtual void loadNetRecord( DtNetUnderwaterAcousticEmitterSystem& record) const`
- `DtU8 myDataLength`
Data members:
- `DtU8 myNumberOfBeams`
- `DtUnderwaterAcousticSystemName myAcousticName`
- `DtUnderwaterAcousticFunction myFunction`
- `DtU8 myAcousticID`
- `DtVector myLocation`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtUnderwaterAcousticSystemName

-
-  DtUnderwaterAcousticOtherSystem
 -  DtUnderwaterAcousticAN_BQQ_5
 -  DtUnderwaterAcousticAN_SSQ_62
 -  DtUnderwaterAcousticAN_SQS_23
 -  DtUnderwaterAcousticAN_SQS_26
 -  DtUnderwaterAcousticAN_SQS_53
 -  DtUnderwaterAcousticALFS
 -  DtUnderwaterAcousticLFA
 -  DtUnderwaterAcousticAN_AQS_901
 -  DtUnderwaterAcousticAN_AQS_902
-

Documentation

-  DtUnderwaterAcousticOtherSystem
-  DtUnderwaterAcousticAN_BQQ_5
-  DtUnderwaterAcousticAN_SSQ_62
-  DtUnderwaterAcousticAN_SQS_23
-  DtUnderwaterAcousticAN_SQS_26
-  DtUnderwaterAcousticAN_SQS_53
-  DtUnderwaterAcousticALFS
-  DtUnderwaterAcousticLFA
-  DtUnderwaterAcousticAN_AQS_901
-  DtUnderwaterAcousticAN_AQS_902

In file disenums.h:

enum DtUnderwaterAcousticFunction

-
-  DtUnderwaterAcousticOtherFunction
 -  DtUnderwaterAcousticPlatformSearch_detect_track
 -  DtUnderwaterAcousticNavigation
 -  DtUnderwaterAcousticMineHunting
 -  DtUnderwaterAcousticWeaponSearch_detect_track_detect
-

Documentation

-  DtUnderwaterAcousticOtherFunction
-  DtUnderwaterAcousticPlatformSearch_detect_track
-  DtUnderwaterAcousticNavigation
-  DtUnderwaterAcousticMineHunting
-  DtUnderwaterAcousticWeaponSearch_detect_track_detect

In file undwaacbeam.h:

class DtUnderwaterAcousticBeam

class DtUnderwaterAcousticBeam: DtUnderwaterAcousticBeam is a class which contains the beam fundamental data parameters for the underwater acoustics PDU

Public Methods

	DtUnderwaterAcousticBeam () <i>Constructors</i>
	DtUnderwaterAcousticBeam (const DtUnderwaterAcousticBeam& orig)
 DtUnderwaterAcousticBeam &	operator= (const DtUnderwaterAcousticBeam& orig) <i>Assignment operator</i>
 DtUnderwaterAcousticBeam &	operator= (const DtNetUnderwaterAcousticBeam& orig)
 virtual	~DtUnderwaterAcousticBeam () <i>Destructor</i>
 virtual void	setBeamDataLength (DtU8 length) <i>Set/Get the beam data length</i>
 virtual DtU8	beamDataLength () const
 virtual void	setBeamID_Number (DtU8 num) <i>Set/Get the beam ID number</i>
 virtual DtU8	beamID_Number () const
 virtual void	setActiveEmissionParmIndex (DtUnderwaterAcousticActiveEmissionParmIndex ind) <i>Set/Get the active emission parameter index</i>
 virtual DtUnderwaterAcousticActiveEmissionParmIndex	activeEmissionParmIndex () const
 virtual void	setScanPattern (DtUnderwaterAcousticScanPattern value) <i>Set/Get the acoustic scan pattern</i>
 virtual DtUnderwaterAcousticScanPattern	scanPattern () const
 virtual void	setBeamCenterAzimuth (DtFloat32 value) <i>Set/Get the beam center azimuth</i>
 virtual DtFloat32	beamCenterAzimuth () const
 virtual void	setAzimuthalBeamwidth (DtFloat32 value) <i>Set/Get the azimuthal beamwidth</i>
 virtual DtFloat32	azimuthalBeamwidth () const
 virtual void	setBeamCenterD_E (DtFloat32 value) <i>Set/Get the beam center D/E</i>
 virtual DtFloat32	beamCenterD_E () const
 virtual void	setD_E_Beamwidth (DtFloat32 value) <i>Set/Get the D/E beamwidth</i>
 virtual DtFloat32	D_E_Beamwidth () const

- virtual void
- virtual void
- virtual void

```

print () const
    Print the information contained in the record
unloadNetRecord ( const DtNetUnderwaterAcousticBeam&
record)
    Load/unload data from a packet record
loadNetRecord ( DtNetUnderwaterAcousticBeam& record)
const

```

Protected Fields

- DtU8 **myBeamDataLength**
Data members:
- DtU8 **myBeamID_Number**
- [DtUnderwaterAcousticActiveEmissionParmIndex](#) **myActiveEmissionParmIndex**
- [DtUnderwaterAcousticScanPattern](#) **myScanPattern**
- DtFloat32 **myBeamCenterAzimuth**
- DtFloat32 **myAzimuthalBeamwidth**
- DtFloat32 **myBeamCenterD_E**
- DtFloat32 **myD_E_Beamwidth**

Documentation

class DtUnderwaterAcousticBeam: DtUnderwaterAcousticBeam is a class which contains the beam fundamental data parameters for the underwater acoustics PDU

- DtUnderwaterAcousticBeam()**
Constructors
- DtUnderwaterAcousticBeam(const [DtUnderwaterAcousticBeam](#)& orig)**
- [DtUnderwaterAcousticBeam](#)& operator=(const [DtUnderwaterAcousticBeam](#)& orig)**
Assignment operator
- [DtUnderwaterAcousticBeam](#)& operator=(const DtNetUnderwaterAcousticBeam& orig)**
- virtual ~DtUnderwaterAcousticBeam()**
Destructor
- virtual void setBeamDataLength(DtU8 length)**
Set/Get the beam data length
- virtual DtU8 beamDataLength() const**
- virtual void setBeamID_Number(DtU8 num)**
Set/Get the beam ID number
- virtual DtU8 beamID_Number() const**

virtual void setActiveEmissionParmIndex([DtUnderwaterAcousticActiveEmissionParmIndex](#) ind)

Set/Get the active emission parameter index

virtual [DtUnderwaterAcousticActiveEmissionParmIndex](#) activeEmissionParmIndex() const

virtual void setScanPattern([DtUnderwaterAcousticScanPattern](#) value)

Set/Get the acoustic scan pattern

virtual [DtUnderwaterAcousticScanPattern](#) scanPattern() const

virtual void setBeamCenterAzimuth(DtFloat32 value)

Set/Get the beam center azimuth

virtual DtFloat32 beamCenterAzimuth() const

virtual void setAzimuthalBeamwidth(DtFloat32 value)

Set/Get the azimuthal beamwidth

virtual DtFloat32 azimuthalBeamwidth() const

virtual void setBeamCenterD_E(DtFloat32 value)

Set/Get the beam center D/E

virtual DtFloat32 beamCenterD_E() const

virtual void setD_E_Beamwidth(DtFloat32 value)

Set/Get the D/E beamwidth

virtual DtFloat32 D_E_Beamwidth() const

virtual void print() const

Print the information contained in the record

virtual void unloadNetRecord(const DtNetUnderwaterAcousticBeam& record)

Load/unload data from a packet record

virtual void loadNetRecord(DtNetUnderwaterAcousticBeam& record) const

DtU8 myBeamDataLength

Data members:

DtU8 myBeamID_Number

[DtUnderwaterAcousticActiveEmissionParmIndex](#) myActiveEmissionParmIndex

[DtUnderwaterAcousticScanPattern](#) myScanPattern

DtFloat32 myBeamCenterAzimuth

DtFloat32 myAzimuthalBeamwidth

DtFloat32 myBeamCenterD_E

DtFloat32 myD_E_Beamwidth

This class has no child classes.

In file disenums.h:

enum

DtUnderwaterAcousticActiveEmissionParmIndex

 **DtUnderwaterAcousticActiveOtherIndex**

Documentation

 **DtUnderwaterAcousticActiveOtherIndex**

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtUnderwaterAcousticScanPattern

-
-  DtUnderwaterAcousticScanPatternNotUsed
 -  DtUnderwaterAcousticConical
 -  DtUnderwaterAcousticHelical
 -  DtUnderwaterAcousticRaster
 -  DtUnderwaterAcousticSectorSearch
 -  DtUnderwaterAcousticContinuousSearch
-

Documentation

-  DtUnderwaterAcousticScanPatternNotUsed
-  DtUnderwaterAcousticConical
-  DtUnderwaterAcousticHelical
-  DtUnderwaterAcousticRaster
-  DtUnderwaterAcousticSectorSearch
-  DtUnderwaterAcousticContinuousSearch

In file transmitpdu.h:

class DtTransmitterPdu: public DtPdu

Inheritance:

DtTransmitterPdu - DtPdu - DtBaseMsg

Public Methods

	DtTransmitterPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtTransmitterPdu (const DtNetTransmitterPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	addCallback (DtExerciseConn *conn, DtTransmitterPduCb cb, void *usr)
 static void	removeCallback (DtExerciseConn *conn, DtTransmitterPduCb cb, void *usr)
 static void	
 const DtEntityIdentifier &	entityId () const
 unsigned int	radioid () const
 const DtRadioEntityType &	radioEntityType () const
 const DtEntityType &	radioEntityType () const
 DtRadioTransmitState	transmitState () const
 DtRadioInputSource	inputSource () const
 DtConstVector	worldAntennaLoc () const
 DtConstVector	relativeAntennaLoc () const
 DtAntennaPatternType	antennaPatternType () const
 int	antennaParamsLength () const
 unsigned long	frequency (int upperLower = DtFREQ_LOWER) const
 float	bandwidth () const
 float	power () const
 const DtModulationType &	modulationType () const
 DtRadioCryptoSystem	cryptoSystem () const
 int	cryptoKeyId () const

int	modulationParamsLength () const
void	getNetModulationParams (void *destination) const
void	getNetAntennaParams (void *destination) const
const DtBeamAntennaPatternRecord &	antennaParams () const
void	setEntityId (const DtEntityIdentifier &id)
void	setRadioid (unsigned int id)
void	setRadioEntityType (const DtRadioEntityType &type)
void	setRadioEntityType (const DtEntityType &type)
void	setTransmitState (DtRadioTransmitState state)
void	setInputSource (DtRadioInputSource source)
void	setWorldAntennaLoc (DtConstVector loc)
void	setRelativeAntennaLoc (DtConstVector loc)
void	setAntennaPatternType (DtAntennaPatternType type)
void	setFrequency (unsigned long upper, unsigned long lower)
void	setFrequency (unsigned long freq)
void	setBandwidth (float bw)
void	setPower (float power)
void	setModulationType (const DtModulationType &type)
void	setCryptoSystem (DtRadioCryptoSystem sys)
void	setCryptoKeyId (int key)
void	setModulationParams (void *networkRepresentation, int size)
void	setAntennaPatternParams (void *networkRepresentation, int size)
void	setAntennaPatternParams (const DtBeamAntennaPatternRecord &rec)
void	printData () const

Protected Fields

DtEntityIdentifier	theId
DtRadioEntityType	theType
DtEntityType	theType
DtVector	wLoc
DtVector	rLoc
DtModulationType	modType
DtBeamAntennaPatternRecord	pat

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
 - void **setModulationParamsLength** (int length)
 - void **setAntennaParamsLength** (int length)
 - void* **antennaParamsAddr** () const
 - void* **modulationParamsAddr** () const
 - virtual void **printModulationParams** () const
 - virtual void **printAntennaParams** () const
-

Inherited from [DtPdu](#):

Public Methods

- [DtPdu](#)& **operator=**(const [DtPdu](#) &pdu)
- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const

- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader()** const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtTransmitterPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtTransmitterPdu(const DtNetTransmitterPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void addCallback([DtExerciseConn](#) *conn, DtTransmitterPduCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtTransmitterPduCb cb, void *usr)
- const [DtEntityIdentifier](#)& entityId() const
- unsigned int radioId() const
- const [DtRadioEntityType](#)& radioEntityType() const
- const [DtEntityType](#)& radioEntityType() const
- [DtRadioTransmitState](#) transmitState() const
- [DtRadioInputSource](#) inputSource() const

`DtConstVector worldAntennaLoc() const`

`DtConstVector relativeAntennaLoc() const`

`DtAntennaPatternType antennaPatternType() const`

`int antennaParamsLength() const`

`unsigned long frequency(int upperLower = DtFREQ_LOWER) const`

`float bandwidth() const`

`float power() const`

`const DtModulationType& modulationType() const`

`DtRadioCryptoSystem cryptoSystem() const`

`int cryptoKeyId() const`

`int modulationParamsLength() const`

`void getNetModulationParams(void *destination) const`

`void getNetAntennaParams(void *destination) const`

`const DtBeamAntennaPatternRecord& antennaParams() const`

`void setEntityId(const DtEntityIdentifier &id)`

`void setRadioId(unsigned int id)`

`void setRadioEntityType(const DtRadioEntityType &type)`

`void setRadioEntityType(const DtEntityType &type)`

`void setTransmitState(DtRadioTransmitState state)`

`void setInputSource(DtRadioInputSource source)`

`void setWorldAntennaLoc(DtConstVector loc)`

`void setRelativeAntennaLoc(DtConstVector loc)`

`void setAntennaPatternType(DtAntennaPatternType type)`

`void setFrequency(unsigned long upper, unsigned long lower)`

`void setFrequency(unsigned long freq)`

`void setBandwidth(float bw)`

`void setPower(float power)`

`void setModulationType(const DtModulationType &type)`

`void setCryptoSystem(DtRadioCryptoSystem sys)`

`void setCryptoKeyId(int key)`

`void setModulationParams(void *networkRepresentation, int size)`

- `void setAntennaPatternParams(void *networkRepresentation, int size)`
- `void setAntennaPatternParams(const DtBeamAntennaPatternRecord &rec)`
- `void printData() const`
- `DtPduKind internalGetPduKind() const`
- `DtEntityIdentifier theId`
- `DtRadioEntityType theType`
- `DtEntityType theType`
- `DtVector wLoc`
- `DtVector rLoc`
- `DtModulationType modType`
- `DtBeamAntennaPatternRecord pat`
- `void setModulationParamsLength(int length)`
- `void setAntennaParamsLength(int length)`
- `void* antennaParamsAddr() const`
- `void* modulationParamsAddr() const`
- `virtual void printModulationParams() const`
- `virtual void printAntennaParams() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtRadioTransmitState

-
-  DtOff
 -  DtRadioOnNotTrasmitting
 -  DtOn
-

Documentation

-  DtOff
-  DtRadioOnNotTrasmitting
-  DtOn

In file disenums.h:

enum DtRadioInputSource

-  DtRadioInputOther
 -  DtPilot
 -  DtCoPilot
 -  DtFirstOfficer
 -  DtDriver
 -  DtLoader
 -  DtGunner
 -  DtCommander
 -  DtDigitalDataDevice
 -  DtIntercom
-

Documentation

-  DtRadioInputOther
-  DtPilot
-  DtCoPilot
-  DtFirstOfficer
-  DtDriver
-  DtLoader
-  DtGunner
-  DtCommander
-  DtDigitalDataDevice
-  DtIntercom

In file disenums.h:

enum DtAntennaPatternType

-
-  DtOmniDirectional
 -  DtBeam
 -  DtSphericalHarmonic
-

Documentation

-  DtOmniDirectional
-  DtBeam
-  DtSphericalHarmonic

In file transmitpdu.h:

class DtModulationType

Public Methods

	DtModulationType (int spectrum = 0, DtRadioMajorModulation major = DtRadioMajorModOther, int detail = 0, DtRadioSystem sys = DtRadioSystemOther)
unsigned int	spectrum () const
DtRadioMajorModulation	majorModType () const
int	detail () const
DtRadioSystem	systemType () const
void	setSpectrum (unsigned int s)
void	setMajor (DtRadioMajorModulation m)
void	setDetail (int d)
void	setSystemType (DtRadioSystem s)
void	printData () const

Private Fields

unsigned int	Spectrum
DtRadioMajorModulation	Major
int	Detail
DtRadioSystem	Sys

Documentation

```

DtModulationType(int spectrum = 0, DtRadioMajorModulation major =
DtRadioMajorModOther, int detail = 0, DtRadioSystem sys = DtRadioSystemOther)

unsigned int spectrum() const

DtRadioMajorModulation majorModType() const

int detail() const

DtRadioSystem systemType() const
    
```

- void setSpectrum(unsigned int s)
- void setMajor([DtRadioMajorModulation](#) m)
- void setDetail(int d)
- void setSystemType([DtRadioSystem](#) s)
- void printData() const
- unsigned int Spectrum
- [DtRadioMajorModulation](#) Major
- int Detail
- [DtRadioSystem](#) Sys

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtRadioMajorModulation

-
-  DtRadioMajorModOther
 -  DtAmplitude
 -  DtAmplitudeAndAngle
 -  DtAngle
 -  DtCombination
 -  DtPulse
 -  DtUnmodulated
-

Documentation

-  DtRadioMajorModOther
-  DtAmplitude
-  DtAmplitudeAndAngle
-  DtAngle
-  DtCombination
-  DtPulse
-  DtUnmodulated

In file disenums.h:

enum DtRadioSystem

 DtRadioSystemOther

 DtGeneric

 DtHq

 DtHqII

 DtHqIIA

 DtSincgards

Documentation

 DtRadioSystemOther

 DtGeneric

 DtHq

 DtHqII

 DtHqIIA

 DtSincgards

In file disenums.h:

enum DtRadioCryptoSystem

-
-  DtRadioCryptoSysOther
 -  DtRadioCryptoSysKY_28
 -  DtRadioCryptoSysKY_8
 -  DtRadioCryptoSysNarrowSpectrumSecureVoice_NSVE
 -  DtRadioCryptoSysWideSpectrumSecureVoice_WSVE
 -  DtRadioCryptoSysSINCGARS_ICOM
-

Documentation

-  DtRadioCryptoSysOther
-  DtRadioCryptoSysKY_28
-  DtRadioCryptoSysKY_8
-  DtRadioCryptoSysNarrowSpectrumSecureVoice_NSVE
-  DtRadioCryptoSysWideSpectrumSecureVoice_WSVE
-  DtRadioCryptoSysSINCGARS_ICOM

In file beamantrec.h:

class DtBeamAntennaPatternRecord: public DtAntennaPatternRecord

class DtBeamAntennaPatternRecord: Instances of DtBeamAntennaPatternRecord are beam antenna pattern records used in conjunction with DtRadioTransmitterStateRepository (TSR)

Inheritance:

DtBeamAntennaPatternRecord - DtAntennaPatternRecord

Public Methods

	DtBeamAntennaPatternRecord () <i>default constructor</i>
	DtBeamAntennaPatternRecord (const DtBeamAntennaPatternRecord & orig) <i>copy constructor</i>
 virtual	~DtBeamAntennaPatternRecord () <i>destructor</i>
 DtBeamAntennaPatternRecord &	operator= (const DtBeamAntennaPatternRecord & orig) <i>assignment operator</i>
	DtBeamAntennaPatternRecord (const DtAntennaPatternRecord & orig) <i>Initialize from DtAntennaPatternRecord</i>
 virtual void	setAngles (const DtTaitBryan &angles) <i>Set/Get the beam direction</i>
 virtual const DtTaitBryan &	angles () const
 virtual void	setBeamwidth (DtFloat32 azimuth, DtFloat32 elevation) <i>Set/Get the azimuth bandwidth and elevation</i>
 virtual DtFloat32	azimuthBeamwidth () const <i>Get the azimuth beam bandwidth</i>
 virtual DtFloat32	elevationBeamwidth () const <i>Get the elevation beam width</i>
 virtual void	setReferenceSystem (const DtReferenceSys sys) <i>Set/Get the beam reference system</i>
 virtual DtReferenceSys	referenceSystem () const
 virtual void	setElectricField (DtFloat32 z, DtFloat32 x) <i>Set electric field</i>
 virtual DtFloat32	electricFieldZ () const <i>Get electric field Z</i>

virtual DtFloat32	electricFieldX () const <i>Get electric field X</i>
virtual void	setPhase (DtFloat32 phase) <i>Set/Get the phase</i>
virtual DtFloat32	phase () const
virtual void	printData () const <i>print the class data</i>
virtual DtAntennaPatternRecord *	clone () const <i>Returns a clone of the instance</i>
virtual void	setFromNet (const void* net, int size) <i>Set from network representation</i>
virtual const void*	netRep (int* size = NULL) const <i>Returns network representation, fills size with size of netRep</i>
virtual int	netSize () const <i>Returns size of network representation</i>
virtual void	setFromNetBeam (const DtNetBeamAntennaPatternRecord& net) <i>Set from network representation</i>
virtual void	setFromNet (const DtNetBeamAntennaPatternRecord& net) <i>Also sets from network representation</i>
virtual const DtNetBeamAntennaPatternRecord&	netBeam () const <i>Returns network representation</i>
virtual const DtNetBeamAntennaPatternRecord&	networkRepresentation () const <i>Also returns network representation</i>
inline const DtNetBeamAntennaPatternRecord&	networkRepresentation () const
inline void	setFromNet (const DtNetBeamAntennaPatternRecord& net)

Protected Fields

DtTaitBryan	myBeamDirection
DtFloat32	myAzimuthBandwidth
DtFloat32	myElevationBeamwidth
DtReferenceSys	myReferenceSystem
DtFloat32	myEz
DtFloat32	myEx
DtFloat32	myPhase

Inherited from [DtAntennaPatternRecord](#):

Public Methods

- [DtAntennaPatternRecord](#)& **operator=** (const [DtAntennaPatternRecord](#)& orig)
- [DtBoolean](#) **operator==** (const [DtAntennaPatternRecord](#)& orig) const
- [DtBoolean](#) **operator!=** (const [DtAntennaPatternRecord](#)& orig) const

Protected Fields

- char* myNetRep
- int myNetSize

Documentation

class DtBeamAntennaPatternRecord:

Instances of DtBeamAntennaPatternRecord are beam antenna pattern records used in conjunction with DtRadioTransmitterStateRepository (TSR). A DtAntennaPatternRecord obtained using the TSR's antennaPattern() inspector can be used to initialize a DtBeamAntennaPatternRecord, whose inspectors you can then use to view the data (assuming the AntennaPattern Record really contained BeamAntennaPattern data. Similarly, a DtBeamAntennaPatternRecord can be filled out and passed to TSR's setAntennaPattern.

- DtBeamAntennaPatternRecord()
default constructor
- DtBeamAntennaPatternRecord(const [DtBeamAntennaPatternRecord](#)& orig)
copy constructor
- virtual ~DtBeamAntennaPatternRecord()
destructor
- [DtBeamAntennaPatternRecord](#)& operator=(const [DtBeamAntennaPatternRecord](#)& orig)
assignment operator
- DtBeamAntennaPatternRecord(const [DtAntennaPatternRecord](#)& orig)
Initialize from DtAntennaPatternRecord
- virtual void setAngles(const [DtTaitBryan](#) &angles)
Set/Get the beam direction
- virtual const [DtTaitBryan](#)& angles() const
- virtual void setBeamwidth(DtFloat32 azimuth, DtFloat32 elevation)
Set/Get the azimuth bandwidth and elevation
- virtual DtFloat32 azimuthBeamwidth() const
Get the azimuth beam bandwidth
- virtual DtFloat32 elevationBeamwidth() const
Get the elevation beam width
- virtual void setReferenceSystem(const [DtReferenceSys](#) sys)
Set/Get the beam reference system
- virtual [DtReferenceSys](#) referenceSystem() const

- virtual void setElectricField(DtFloat32 z, DtFloat32 x)
 Set electric field
- virtual DtFloat32 electricFieldZ() const
 Get electric field Z
- virtual DtFloat32 electricFieldX() const
 Get electric field X
- virtual void setPhase(DtFloat32 phase)
 Set/Get the phase
- virtual DtFloat32 phase() const
- virtual void printData() const
 print the class data
- virtual [DtAntennaPatternRecord](#)* clone() const
 Returns a clone of the instance
- virtual void setFromNet(const void* net, int size)
 Set from network representation
- virtual const void* netRep(int* size = NULL) const
 Returns network representation, fills size with size of netRep
- virtual int netSize() const
 Returns size of network representation
- virtual void setFromNetBeam(const DtNetBeamAntennaPatternRecord& net)
 Set from network representation
- virtual void setFromNet(const DtNetBeamAntennaPatternRecord& net)
 Also sets from network representation. This function is here for backwards compatibility with pre-3.3 releases.
- virtual const DtNetBeamAntennaPatternRecord& netBeam() const
 Returns network representation
- virtual const DtNetBeamAntennaPatternRecord& networkRepresentation() const
 Also returns network representation. This function is here for backwards compatibility with pre-3.3 releases.
- [DtTaitBryan](#) myBeamDirection
- DtFloat32 myAzimuthBandwidth
- DtFloat32 myElevationBeamwidth
- [DtReferenceSys](#) myReferenceSystem
- DtFloat32 myEz
- DtFloat32 myEx
- DtFloat32 myPhase
- inline const DtNetBeamAntennaPatternRecord& networkRepresentation() const

 `inline void setFromNet( const DtNetBeamAntennaPatternRecord& net)`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file antpatrec.h:

class DtAntennaPatternRecord

class DtAntennaPatternRecord: Instances of DtAntennaPatternRecord are

Inheritance:

DtAntennaPatternRecord

Public Methods

	DtAntennaPatternRecord () <i>default constructor</i>
	DtAntennaPatternRecord (const DtAntennaPatternRecord & orig) <i>copy constructor</i>
 virtual	~DtAntennaPatternRecord () <i>destructor</i>
 DtAntennaPatternRecord &	operator= (const DtAntennaPatternRecord & orig) <i>assignment operator</i>
 DtBoolean	operator== (const DtAntennaPatternRecord & orig) const <i>equality operator</i>
 DtBoolean	operator!= (const DtAntennaPatternRecord & orig) const <i>inequality operator</i>
 virtual void	setFromNet (const void* net, int size) <i>Set from network representation</i>
 virtual const void*	netRep (int* size = NULL) const <i>Returns network representation, fills size with size of netRep</i>
 virtual int	netSize () const <i>Returns size of network representation</i>
 virtual DtAntennaPatternRecord *	clone () const <i>Returns a new'ed clone of the instance</i>
 virtual void	printData () const <i>prints the data</i>

Protected Fields

	char* myNetRep
	int myNetSize

Documentation

class DtAntennaPatternRecord: Instances of DtAntennaPatternRecord are

- `DtAntennaPatternRecord()`
default constructor
- `DtAntennaPatternRecord(const DtAntennaPatternRecord& orig)`
copy constructor
- `virtual ~DtAntennaPatternRecord()`
destructor
- `DtAntennaPatternRecord& operator= (const DtAntennaPatternRecord& orig)`
assignment operator
- `DtBoolean operator== (const DtAntennaPatternRecord& orig) const`
equality operator
- `DtBoolean operator!= (const DtAntennaPatternRecord& orig) const`
inequality operator
- `virtual void setFromNet(const void* net, int size)`
Set from network representation
- `virtual const void* netRep(int* size = NULL) const`
Returns network representation, fills size with size of netRep. Return value is pointer to internal object with same lifetime as this. Do not delete.
- `virtual int netSize() const`
Returns size of network representation
- `virtual DtAntennaPatternRecord* clone() const`
Returns a new'ed clone of the instance. Caller deletes.
- `virtual void printData() const`
prints the data
- `char* myNetRep`
- `int myNetSize`

Direct child classes:

[DtBeamAntennaPatternRecord](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtReferenceSys

 DtReferenceSysWorldCoordinates

 DtReferenceSysEntityCoordinates

Documentation

 DtReferenceSysWorldCoordinates

 DtReferenceSysEntityCoordinates

In file supemisespdu.h:

class DtSuppEmisEntStatePdu: public DtPdu

Inheritance:

DtSuppEmisEntStatePdu - DtPdu - DtBaseMsg

Public Methods

	DtSuppEmisEntStatePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtSuppEmisEntStatePdu (const DtNetSuppEmisEntStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtSuppEmisEntStatePdu (const DtSuppEmisEntStatePdu & orig)
 virtual	~DtSuppEmisEntStatePdu () <i>destructor</i>
 DtSuppEmisEntStatePdu &	operator= (const DtSuppEmisEntStatePdu & orig) <i>Assignment operator</i>
 virtual void	setOriginatingEntityID (const DtEntityIdentifier & entity) <i>Set/Get originating entity identifier</i>
 virtual const DtEntityIdentifier &	originatingEntityID () const
 virtual void	setInfraRedSigRepIndex (DtU16 value) <i>Set/Get infra-red signature representation index</i>
 virtual DtU16	infraRedSigRepIndex () const
 virtual void	setAcousticSigRepIndex (DtU16 value) <i>Set/Get acoustic signature representation index</i>
 virtual DtU16	acousticSigRepIndex () const
 virtual void	setRadarSigRepIndex (DtU16 value) <i>Set/Get radar signature representation index</i>
 virtual DtU16	radarSigRepIndex () const
 virtual DtBoolean	setNumPropulsionSystems (DtU16 value) <i>Set/Get number of propulsion systems</i>
 virtual DtU16	numPropulsionSystems () const
 virtual DtBoolean	setNumNozzleSystems (DtU16 value) <i>Set/Get number of nozzle systems</i>
 virtual DtU16	numNozzleSystems () const
 virtual DtBoolean	setPropulsionSystem (int index, const DtPropulsionSystem & record) <i>Set/Get a propulsion system data record</i>

virtual DtBoolean	getPropulsionSystem (int index, DtPropulsionSystem & record) const
virtual DtBoolean	setNozzleSystem (int index, const DtVectoringNozzleSystem & record) <i>Set/Get a vectoring nozzle system data record</i>
virtual DtBoolean	getNozzleSystem (int index, DtVectoringNozzleSystem & record) const
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtSuppEmisEntStatePdu</i>
static void	addCallback (DtExerciseConn * conn, DtSuppEmisEntStatePduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtSuppEmisEntStatePduCb cb, void* usr)

Protected Fields

static int	theStaticSize <i>Size of PDU with zero parameter records</i>
DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
virtual DtNetSuppEmisEntStatePdu*	netPduStruct () const <i>Casts the netPdu element to our network struct</i>
DtSuppEmisEntStatePdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type</i>
virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>
virtual int	propulsionSystemRecordOffset (int recordIndex) const <i>Returns a buffer offset to the propulsion system data record</i>
virtual int	nozzleSystemRecordOffset (int recordIndex) const <i>Returns a buffer offset to the given nozzle system data record</i>

Inherited from [DtPdu](#):

Public Methods

virtual void* packet () const
virtual int status () const
virtual void setVersion (int vers)
virtual int protocolVersion () const
virtual void setExerciseld (int id)

- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)

- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

`DtSuppEmisEntStatePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

Constructors

`DtSuppEmisEntStatePdu(const DtNetSuppEmisEntStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

`DtSuppEmisEntStatePdu(const DtSuppEmisEntStatePdu& orig)`

`virtual ~DtSuppEmisEntStatePdu()`

destructor

`DtSuppEmisEntStatePdu& operator=(const DtSuppEmisEntStatePdu& orig)`

Assignment operator

`virtual void setOriginatingEntityID(const DtEntityIdentifier& entity)`

Set/Get originating entity identifier

`virtual const DtEntityIdentifier& originatingEntityID() const`

`virtual void setInfraRedSigRepIndex(DtU16 value)`

Set/Get infra-red signature representation index

`virtual DtU16 infraRedSigRepIndex() const`

`virtual void setAcousticSigRepIndex(DtU16 value)`

Set/Get acoustic signature representation index

`virtual DtU16 acousticSigRepIndex() const`

`virtual void setRadarSigRepIndex(DtU16 value)`

Set/Get radar signature representation index

`virtual DtU16 radarSigRepIndex() const`

`virtual DtBoolean setNumPropulsionSystems(DtU16 value)`

Set/Get number of propulsion systems

`virtual DtU16 numPropulsionSystems() const`

`virtual DtBoolean setNumNozzleSystems(DtU16 value)`

Set/Get number of nozzle systems

`virtual DtU16 numNozzleSystems() const`

`virtual DtBoolean setPropulsionSystem(int index, const DtPropulsionSystem& record)`

Set/Get a propulsion system data record. Returns false if the record does not exist.

`virtual DtBoolean getPropulsionSystem(int index, DtPropulsionSystem& record) const`

`virtual DtBoolean setNozzleSystem(int index, const DtVectoringNozzleSystem& record)`

Set/Get a vectoring nozzle system data record. Returns false if the record does not exist.

- `virtual DtBoolean getNozzleSystem(int index, DtVectoringNozzleSystem& record) const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtSuppEmisEntStatePdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtSuppEmisEntStatePduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtSuppEmisEntStatePduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetSuppEmisEntStatePdu* netPduStruct() const`
Casts the netPdu element to our network struct
- `DtSuppEmisEntStatePdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- `virtual char* var() const`
Returns a pointer to the first byte of variable length data
- `virtual int propulsionSystemRecordOffset(int recordIndex) const`
Returns a buffer offset to the propulsion system data record
- `virtual int nozzleSystemRecordOffset(int recordIndex) const`
Returns a buffer offset to the given nozzle system data record
- `static int theStaticSize`
Size of PDU with zero parameter records
- `DtEntityIdentifier myEntityID`
Temporary objects for reference returns

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file supemisesprop.h:

class DtPropulsionSystem

class DtPropulsionSystem: DtPropulsionSystem is a class which contains the propulsion system data for the supplemental emissions/entity state PDU

Public Methods

	DtPropulsionSystem () <i>Constructors</i>
	DtPropulsionSystem (const DtPropulsionSystem & orig)
 DtPropulsionSystem &	operator= (const DtPropulsionSystem & orig) <i>Assignment operator</i>
 DtPropulsionSystem &	operator= (const DtNetPropulsionSystem& orig)
 virtual	~DtPropulsionSystem () <i>Destructor</i>
 virtual void	setPowerSetting (DtFloat32 value) <i>Set/Get the power setting</i>
 virtual DtFloat32	powerSetting () const
 virtual void	setEngineRPM (DtFloat32 value) <i>Set/Get the engine RPM</i>
 virtual DtFloat32	engineRPM () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetPropulsionSystem& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetPropulsionSystem& record) const

Protected Fields

 DtFloat32	myPowerSetting <i>Data members:</i>
 DtFloat32	myEngineRPM

Documentation

class DtPropulsionSystem: DtPropulsionSystem is a class which contains the propulsion system data for the supplemental emissions/entity state PDU

	DtPropulsionSystem () Constructors
---	--

- `DtPropulsionSystem(const DtPropulsionSystem& orig)`
 - `DtPropulsionSystem& operator=( const DtPropulsionSystem& orig)`
Assignment operator
 - `DtPropulsionSystem& operator=( const DtNetPropulsionSystem& orig)`
 - `virtual ~DtPropulsionSystem()`
Destructor
 - `virtual void setPowerSetting(DtFloat32 value)`
Set/Get the power setting
 - `virtual DtFloat32 powerSetting() const`
 - `virtual void setEngineRPM(DtFloat32 value)`
Set/Get the engine RPM
 - `virtual DtFloat32 engineRPM() const`
 - `virtual void print() const`
Print the information contained in the record
 - `virtual void unloadNetRecord( const DtNetPropulsionSystem& record)`
Load/Unload data from a packet record
 - `virtual void loadNetRecord( DtNetPropulsionSystem& record) const`
 - `DtFloat32 myPowerSetting`
Data members:
 - `DtFloat32 myEngineRPM`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file supemisesnozz.h:

class DtVectoringNozzleSystem

class DtVectoringNozzleSystem: DtVectoringNozzleSystem is a class which contains the vectoring nozzle system data for the supplemental emissions/entity state PDU

Public Methods

	DtVectoringNozzleSystem () <i>Constructors</i>
	DtVectoringNozzleSystem (const DtVectoringNozzleSystem & orig)
 DtVectoringNozzleSystem &	operator= (const DtVectoringNozzleSystem & orig) <i>Assignment operator</i>
 DtVectoringNozzleSystem &	operator= (const DtNetVectoringNozzleSystem& orig)
 virtual	~DtVectoringNozzleSystem () <i>Destructor</i>
 virtual void	setHorDeflection (DtFloat32 value) <i>Set/Get the horizontal deflection</i>
 virtual DtFloat32	horDeflection () const
 virtual void	setVertDeflection (DtFloat32 value) <i>Set/Get the vertical deflection</i>
 virtual DtFloat32	vertDeflection () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetVectoringNozzleSystem& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetVectoringNozzleSystem& record) const

Protected Fields

 DtFloat32	myHorizontalDeflection <i>Data members:</i>
 DtFloat32	myVerticalDeflection

Documentation

class DtVectoringNozzleSystem: DtVectoringNozzleSystem is a class which contains the vectoring nozzle system data for the supplemental emissions/entity state PDU

	DtVectoringNozzleSystem () Constructors
---	---

- `DtVectoringNozzleSystem(const DtVectoringNozzleSystem& orig)`
 - `DtVectoringNozzleSystem& operator=( const DtVectoringNozzleSystem& orig)`
Assignment operator
 - `DtVectoringNozzleSystem& operator=( const DtNetVectoringNozzleSystem& orig)`
 - `virtual ~DtVectoringNozzleSystem()`
Destructor
 - `virtual void setHorDeflection(DtFloat32 value)`
Set/Get the horizontal deflection
 - `virtual DtFloat32 horDeflection() const`
 - `virtual void setVertDeflection(DtFloat32 value)`
Set/Get the vertical deflection
 - `virtual DtFloat32 vertDeflection() const`
 - `virtual void print() const`
Print the information contained in the record
 - `virtual void unloadNetRecord( const DtNetVectoringNozzleSystem& record)`
Load/Unload data from a packet record
 - `virtual void loadNetRecord( DtNetVectoringNozzleSystem& record) const`
 - `DtFloat32 myHorizontalDeflection`
Data members:
 - `DtFloat32 myVerticalDeflection`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file servreqpdu.h:

class DtServiceRequestPdu: public DtPdu

Inheritance:

DtServiceRequestPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtServiceRequestPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtServiceRequestPdu (const DtNetServiceRequestPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static void	addCallback (DtExerciseConn *conn, DtServiceRequestPduCb cb, void *usr)
static void	removeCallback (DtExerciseConn *conn, DtServiceRequestPduCb cb, void *usr)
	DtServiceRequestPdu (const DtEntityIdentifier &requester, const DtEntityIdentifier &servicer, <code>#ifdef _DOS DtServiceType serviceType = (DtServiceType) 2,</code> <code>else DtServiceType serviceType = DtServiceTypeRepair,</code> <code>#endif DtBufferPtr another</code> <code>= DtUSE_INTERNAL_BUFFER</code>)
	DtServiceRequestPdu (const DtEntityIdentifier &requester, const DtEntityIdentifier &servicer, const DtEntityType &supply, float quantity, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
const DtEntityIdentifier &	requestingId () const
const DtEntityIdentifier &	servicingId () const
DtServiceType	serviceType () const
int	numSupplyTypes () const
const DtEntityType &	nthSupply (int supplyIndex, float *quantity = NULL, int *retCode = NULL) const
float	supplyQuantity (const DtEntityType &supply) const
void	setRequestingId (const DtEntityIdentifier &requester)
void	setServicingId (const DtEntityIdentifier &servicer)
void	setServiceType (DtServiceType service)
void	setNumSupplyTypes (int numTypes)
void	setNthSupply (int supplyIndex, const DtEntityType &supply, float quantity, int *retCode = NULL)

 void **printData** () const

Protected Fields

 [DtEntityIdentifier](#) **reqId**

 [DtEntityIdentifier](#) **servId**

 [DtEntityType](#) **supType**

Protected Methods

 [DtPduKind](#) **internalGetPduKind** () const

 int **indexInBounds** (int ind) const

Inherited from [DtPdu](#):

Public Methods

 [DtPdu](#)& **operator**=(const [DtPdu](#) &pdu)

 virtual void* **packet**() const

 virtual int **status**() const

 virtual void **setVersion**(int vers)

 virtual int **protocolVersion**() const

 virtual void **setExerciseld**(int id)

 virtual int **exerciseld**() const

 virtual [DtPduKind](#) **kind**() const

 virtual [DtProtocolFamily](#) **protocolFamily**() const

 virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

 virtual DtTime **timeStamp**() const

 virtual [DtTimeStampType](#) **timeStampType**() const

 virtual DtTime **guessTimeValid**(DtTime referenceTime) const

 virtual void **print**() const

 virtual void **printHeader**() const

 inline int **exerciseld**() const

 inline [DtPduKind](#) **kind**() const

- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtServiceRequestPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtServiceRequestPdu(const DtNetServiceRequestPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void addCallback([DtExerciseConn](#) *conn, DtServiceRequestPduCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtServiceRequestPduCb cb, void *usr)
- DtServiceRequestPdu(const [DtEntityIdentifier](#) &requester, const [DtEntityIdentifier](#) &servicer, `ifdef _DOS DtServiceType serviceType = (DtServiceType) 2, else DtServiceType serviceType = DtServiceTypeRepair, endif DtBufferPtr another = DtUSE_INTERNAL_BUFFER)`
- DtServiceRequestPdu(const [DtEntityIdentifier](#) &requester, const [DtEntityIdentifier](#) &servicer, const [DtEntityType](#) &supply, float quantity, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- const [DtEntityIdentifier](#)& requestingId() const
- const [DtEntityIdentifier](#)& servicingId() const

- `DtServiceType serviceType() const`
- `int numSupplyTypes() const`
- `const DtEntityType& nthSupply(int supplyIndex, float *quantity = NULL, int *retCode = NULL) const`
- `float supplyQuantity(const DtEntityType &supply) const`
- `void setRequestingId(const DtEntityIdentifier &requester)`
- `void setServicingId(const DtEntityIdentifier &servicer)`
- `void setServiceType(DtServiceType service)`
- `void setNumSupplyTypes(int numTypes)`
- `void setNthSupply(int supplyIndex, const DtEntityType &supply, float quantity, int *retCode = NULL)`
- `void printData() const`
- `DtPduKind internalGetPduKind() const`
- `int indexInBounds(int ind) const`
- `DtEntityIdentifier reqId`
- `DtEntityIdentifier servId`
- `DtEntityType supType`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtServiceType

-
-  DtServiceTypeOther
 -  DtServiceTypeResupply
 -  DtServiceTypeRepair
-

Documentation

-  DtServiceTypeOther
-  DtServiceTypeResupply
-  DtServiceTypeRepair

In file resupplypdu.h:

class DtResupplyReceivedPdu: public DtPdu

Inheritance:

DtResupplyReceivedPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtResupplyReceivedPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtResupplyReceivedPdu (const DtNetResupplyReceivedPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtResupplyReceivedPdu (const DtEntityIdentifier &receiver, const DtEntityIdentifier &supplier, const DtEntityType &supply, float quantity, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static void	addCallback (DtExerciseConn *conn, DtResupplyReceivedPduCb cb, void *usr)
static void	removeCallback (DtExerciseConn *conn, DtResupplyReceivedPduCb cb, void *usr)
const DtEntityIdentifier &	receivingId () const
const DtEntityIdentifier &	supplyingId () const
int	numSupplyTypes () const
const DtEntityType &	nthSupply (int supplyIndex, float *quantity = NULL, int *retCode = NULL) const
float	supplyQuantity (const DtEntityType &supply) const
void	setReceivingId (const DtEntityIdentifier &requester)
void	setSupplyingId (const DtEntityIdentifier &servicer)
void	setNumSupplyTypes (int numTypes)
void	setNthSupply (int supplyIndex, const DtEntityType &supply, float quantity, int *retCode = NULL)
void	printData () const

Protected Fields

 [DtEntityIdentifier](#) **theId**

 [DtEntityType](#) **supType**

Protected Methods

 [DtPduKind](#) **internalGetPduKind** () const

 **int** **indexInBounds** (int ind) const

Inherited from [DtPdu](#):

Public Methods

 [DtPdu](#)& **operator=**(const [DtPdu](#) &pdu)

 virtual void* **packet**() const

 virtual int **status**() const

 virtual void **setVersion**(int vers)

 virtual int **protocolVersion**() const

 virtual void **setExerciseld**(int id)

 virtual int **exerciseld**() const

 virtual [DtPduKind](#) **kind**() const

 virtual [DtProtocolFamily](#) **protocolFamily**() const

 virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

 virtual DtTime **timeStamp**() const

 virtual [DtTimeStampType](#) **timeStampType**() const

 virtual DtTime **guessTimeValid**(DtTime referenceTime) const

 virtual void **print**() const

 virtual void **printHeader**() const

 inline int **exerciseld**() const

 inline [DtPduKind](#) **kind**() const

 inline DtNetPduHeader* **netHeader**() const

 inline void* **packet**() const

- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader()** const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtResupplyReceivedPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtResupplyReceivedPdu(const DtNetResupplyReceivedPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtResupplyReceivedPdu(const [DtEntityIdentifier](#) &receiver, const [DtEntityIdentifier](#) &supplier, const [DtEntityType](#) &supply, float quantity, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void addCallback([DtExerciseConn](#) *conn, DtResupplyReceivedPduCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtResupplyReceivedPduCb cb, void *usr)
- const [DtEntityIdentifier](#)& receivingId() const
- const [DtEntityIdentifier](#)& supplyingId() const
- int numSupplyTypes() const
- const [DtEntityType](#)& nthSupply(int supplyIndex, float *quantity = NULL, int *retCode =

NULL) const

- float supplyQuantity(const [DtEntityType](#) &supply) const
- void setReceivingId(const [DtEntityIdentifier](#) &requester)
- void setSupplyingId(const [DtEntityIdentifier](#) &servicer)
- void setNumSupplyTypes(int numTypes)
- void setNthSupply(int supplyIndex, const [DtEntityType](#) &supply, float quantity, int *retCode = NULL)
- void printData() const
- [DtPduKind](#) internalGetPduKind() const
- int indexInBounds(int ind) const
- [DtEntityIdentifier](#) theId
- [DtEntityType](#) supType

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file resupplypdu.h:

class DtResupplyOfferPdu: public DtPdu

Inheritance:

DtResupplyOfferPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtResupplyOfferPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtResupplyOfferPdu (const DtNetResupplyOfferPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtResupplyOfferPdu (const DtEntityIdentifier &receiver, const DtEntityIdentifier &supplier, const DtEntityType &supply, float quantity, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static void	addCallback (DtExerciseConn *conn, DtResupplyOfferPduCb cb, void *usr)
 static void	removeCallback (DtExerciseConn *conn, DtResupplyOfferPduCb cb, void *usr)
	const DtEntityIdentifier & receivingId () const
	const DtEntityIdentifier & supplyingId () const
 int	numSupplyTypes () const
 const DtEntityType &	nthSupply (int supplyIndex, float *quantity = NULL, int *retCode = NULL) const
 float	supplyQuantity (const DtEntityType &supply) const
 void	setReceivingId (const DtEntityIdentifier &requester)
 void	setSupplyingId (const DtEntityIdentifier &servicer)
 void	setNumSupplyTypes (int numTypes)
 void	setNthSupply (int supplyIndex, const DtEntityType &supply, float quantity, int *retCode = NULL)
 void	printData () const

Protected Fields

 [DtEntityIdentifier](#) **theId**

 [DtEntityType](#) **supType**

Protected Methods

 [DtPduKind](#) **internalGetPduKind** () const

 **int** **indexInBounds** (int ind) const

Inherited from [DtPdu](#):

Public Methods

 [DtPdu](#)& **operator=**(const [DtPdu](#) &pdu)

 virtual void* **packet**() const

 virtual int **status**() const

 virtual void **setVersion**(int vers)

 virtual int **protocolVersion**() const

 virtual void **setExerciseld**(int id)

 virtual int **exerciseld**() const

 virtual [DtPduKind](#) **kind**() const

 virtual [DtProtocolFamily](#) **protocolFamily**() const

 virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

 virtual DtTime **timeStamp**() const

 virtual [DtTimeStampType](#) **timeStampType**() const

 virtual DtTime **guessTimeValid**(DtTime referenceTime) const

 virtual void **print**() const

 virtual void **printHeader**() const

 inline int **exerciseld**() const

 inline [DtPduKind](#) **kind**() const

 inline DtNetPduHeader* **netHeader**() const

 inline void* **packet**() const

- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader()** const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtResupplyOfferPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtResupplyOfferPdu(const DtNetResupplyOfferPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtResupplyOfferPdu(const [DtEntityIdentifier](#) &receiver, const [DtEntityIdentifier](#) &supplier, const [DtEntityType](#) &supply, float quantity, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void addCallback([DtExerciseConn](#) *conn, DtResupplyOfferPduCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtResupplyOfferPduCb cb, void *usr)
- const [DtEntityIdentifier](#)& receivingId() const
- const [DtEntityIdentifier](#)& supplyingId() const
- int numSupplyTypes() const
- const [DtEntityType](#)& nthSupply(int supplyIndex, float *quantity = NULL, int *retCode =

NULL) const

- float supplyQuantity(const [DtEntityType](#) &supply) const
- void setReceivingId(const [DtEntityIdentifier](#) &requester)
- void setSupplyingId(const [DtEntityIdentifier](#) &servicer)
- void setNumSupplyTypes(int numTypes)
- void setNthSupply(int supplyIndex, const [DtEntityType](#) &supply, float quantity, int *retCode = NULL)
- void printData() const
- [DtPduKind](#) internalGetPduKind() const
- int indexInBounds(int ind) const
- [DtEntityIdentifier](#) theId
- [DtEntityType](#) supType

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file resupplypdu.h:

class DtResupplyCancelPdu: public DtPdu

Inheritance:

DtResupplyCancelPdu - DtPdu - DtBaseMsg

Public Methods

-  **DtResupplyCancelPdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  **DtResupplyCancelPdu** (const DtNetResupplyCancelPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  **DtResupplyCancelPdu** (const [DtEntityIdentifier](#) &receiveld, const [DtEntityIdentifier](#) &supplyld, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
-  static void **addCallback** ([DtExerciseConn](#) *conn, DtResupplyCancelPduCb cb, void *usr)
-  static void **removeCallback** ([DtExerciseConn](#) *conn, DtResupplyCancelPduCb cb, void *usr)
-  const [DtEntityIdentifier](#)& **receiveld** () const
-  const [DtEntityIdentifier](#)& **supplyingld** () const
-  void **setReceiveld** (const [DtEntityIdentifier](#) &id)
-  void **setSupplyingld** (const [DtEntityIdentifier](#) &id)
-  void **printData** () const

Protected Fields

-  [DtEntityIdentifier](#) theId

Protected Methods

-  [DtPduKind](#) internalGetPduKind () const
-

Inherited from [DtPdu](#):

Public Methods

- [DtPdu](#)& **operator**=(const [DtPdu](#) &pdu)
- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtResupplyCancelPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `DtResupplyCancelPdu(const DtNetResupplyCancelPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `DtResupplyCancelPdu(const DtEntityIdentifier &receiveId, const DtEntityIdentifier &supplyId, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static void addCallback(DtExerciseConn *conn, DtResupplyCancelPduCb cb, void *usr)`
 - `static void removeCallback(DtExerciseConn *conn, DtResupplyCancelPduCb cb, void *usr)`
 - `const DtEntityIdentifier& receivingId() const`
 - `const DtEntityIdentifier& supplyingId() const`
 - `void setReceivingId(const DtEntityIdentifier &id)`
 - `void setSupplyingId(const DtEntityIdentifier &id)`
 - `void printData() const`
 - `DtPduKind internalGetPduKind() const`
 - `DtEntityIdentifier theId`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file repairpdu.h:

class DtRepairResponsePdu: public DtPdu

Inheritance:

DtRepairResponsePdu - DtPdu - DtBaseMsg

Public Methods

- **DtRepairResponsePdu** (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- **DtRepairResponsePdu** (const DtNetRepairResponsePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* **create** (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void **addCallback** ([DtExerciseConn](#) *conn, DtRepairResponsePduCb cb, void *usr)
- static void **removeCallback** ([DtExerciseConn](#) *conn, DtRepairResponsePduCb cb, void *usr)
- const [DtEntityIdentifier](#)& **receivingId** () const
- const [DtEntityIdentifier](#)& **repairingId** () const
- [DtRepairResult](#) **repairResult** () const
- void **setReceivingId** (const [DtEntityIdentifier](#) &id)
- void **setRepairingId** (const [DtEntityIdentifier](#) &id)
- void **setRepairResult** ([DtRepairResult](#) result)
- void **printData** () const

Protected Fields

- [DtEntityIdentifier](#) **theId**

Protected Methods

- [DtPduKind](#) **internalGetPduKind** () const
-

Inherited from [DtPdu](#):

Public Methods

- [DtPdu](#)& **operator**=(const [DtPdu](#) &pdu)
- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtRepairResponsePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `DtRepairResponsePdu(const DtNetRepairResponsePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static void addCallback(DtExerciseConn *conn, DtRepairResponsePduCb cb, void *usr)`
 - `static void removeCallback(DtExerciseConn *conn, DtRepairResponsePduCb cb, void *usr)`
 - `const DtEntityIdentifier& receivingId() const`
 - `const DtEntityIdentifier& repairingId() const`
 - `DtRepairResult repairResult() const`
 - `void setReceivingId(const DtEntityIdentifier &id)`
 - `void setRepairingId(const DtEntityIdentifier &id)`
 - `void setRepairResult(DtRepairResult result)`
 - `void printData() const`
 - `DtPduKind internalGetPduKind() const`
 - `DtEntityIdentifier theId`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtRepairResult

-
-  DtRepairResultOther
 -  DtRepairResultEnded
 -  DtRepairResultInvalid
 -  DtRepairResultInterrupted
 -  DtRepairResultCancelled
-

Documentation

-  DtRepairResultOther
-  DtRepairResultEnded
-  DtRepairResultInvalid
-  DtRepairResultInterrupted
-  DtRepairResultCancelled

In file repairpdu.h:

class DtRepairCompletePdu: public DtPdu

Inheritance:

DtRepairCompletePdu - DtPdu - DtBaseMsg

Public Methods

	DtRepairCompletePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtRepairCompletePdu (const DtNetRepairCompletePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static void	addCallback (DtExerciseConn *conn, DtRepairCompletePduCb cb, void *usr)
 static void	removeCallback (DtExerciseConn *conn, DtRepairCompletePduCb cb, void *usr)
 const DtEntityIdentifier &	receivingId () const
 const DtEntityIdentifier &	repairingId () const
 DtRepairType	repairType () const
 void	setReceivingId (const DtEntityIdentifier &id)
 void	setRepairingId (const DtEntityIdentifier &id)
 void	setRepairType (DtRepairType type)
 void	printData () const

Protected Fields

 [DtEntityIdentifier](#) theId

Protected Methods

 [DtPduKind](#) internalGetPduKind () const

Inherited from [DtPdu](#):

Public Methods

- [DtPdu](#)& **operator**=(const [DtPdu](#) &pdu)
- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtRepairCompletePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `DtRepairCompletePdu(const DtNetRepairCompletePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 - `static void addCallback(DtExerciseConn *conn, DtRepairCompletePduCb cb, void *usr)`
 - `static void removeCallback(DtExerciseConn *conn, DtRepairCompletePduCb cb, void *usr)`
 - `const DtEntityIdentifier& receivingId() const`
 - `const DtEntityIdentifier& repairingId() const`
 - `DtRepairType repairType() const`
 - `void setReceivingId(const DtEntityIdentifier &id)`
 - `void setRepairingId(const DtEntityIdentifier &id)`
 - `void setRepairType(DtRepairType type)`
 - `void printData() const`
 - `DtPduKind internalGetPduKind() const`
 - `DtEntityIdentifier theId`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtRepairType

Section 2

- DtNoRepairsPerformed
- DtRepairEverything
- DtDriveTrainMotor
- DtDriveTrainStarter
- DtDriveTrainAlternator
- DtDriveTrainGenerator
- DtDriveTrainBattery
- DtDriveTrainEngineCoolantLeak
- DtDriveTrainFuelFilter
- DtDriveTrainTranmissionOilLeak
- DtDriveTrainEngineOilLeak
- DtDriveTrainPumps
- DtDriveTrainFilters
- DtBodyHull
- DtBodyAirframe
- DtInterfacePropeller
- DtDriveTrainTransmission
- DtDriveTrainBrakes
- DtInterfaceFilters
- DtWeaponGunElevationDrive
- DtWeaponGunStabilizationSys
- DtWeaponGunnersPrimarySight
- DtWeaponCommandersGPSExtension
- DtWeaponLoadingMecahnism
- DtFuelTransferPump
- DtFuelLines

- DtFuelGauges
 - DtElectronicWarfareSys
 - DtElectronicDetectionSys
 - DtElectronicRangeFinder
 - DtElectronicSys
 - DtElectronicRadios
 - DtElectronicIntercoms
 - DtElectronicLasers
 - DtElectronicCoders
 - DtElectronicDecoders
 - DtElectronicComputers
 - DtLifeSupportAirSupply
 - DtLifeSupportFilters
 - DtLifeSupportWaterSupply
 - DtLifeSupportRefrigerationSys
 - DtLifeSupportCBRProtection
 - DtLifeSupportWaterWashSys
 - DtLifeSupportDecontaminationSys
 - DtActuatorWaterSupply
 - DtActuatorCoolingSys
 - DtActuatorWinches
 - DtActuatorCatapults
 - DtActuatorCranes
 - DtActuatorLaunchers
 - DtAuxCraftLifeBoat
 - DtAuxCraftLandingCraft
 - DtAuxCraftEjectionSeat
-

Documentation

Section 2

- DtNoRepairsPerformed

- DtRepairEverything
- DtDriveTrainMotor
- DtDriveTrainStarter
- DtDriveTrainAlternator
- DtDriveTrainGenerator
- DtDriveTrainBattery
- DtDriveTrainEngineCoolantLeak
- DtDriveTrainFuelFilter
- DtDriveTrainTranmissionOilLeak
- DtDriveTrainEngineOilLeak
- DtDriveTrainPumps
- DtDriveTrainFilters
- DtBodyHull
- DtBodyAirframe
- DtInterfacePropeller
- DtDriveTrainTransmission
- DtDriveTrainBrakes
- DtInterfaceFilters
- DtWeaponGunElevationDrive
- DtWeaponGunStabilizationSys
- DtWeaponGunnersPrimarySight
- DtWeaponCommandersGPSExtension
- DtWeaponLoadingMecahnism
- DtFuelTransferPump
- DtFuelLines
- DtFuelGauges
- DtElectronicWarfareSys
- DtElectronicDetectionSys
- DtElectronicRangeFinder
- DtElectronicSys
- DtElectronicRadios
- DtElectronicIntercoms

- DtElectronicLasers
- DtElectronicCoders
- DtElectronicDecoders
- DtElectronicComputers
- DtLifeSupportAirSupply
- DtLifeSupportFilters
- DtLifeSupportWaterSupply
- DtLifeSupportRefrigerationSys
- DtLifeSupportCBRProtection
- DtLifeSupportWaterWashSys
- DtLifeSupportDecontaminationSys
- DtActuatorWaterSupply
- DtActuatorCoolingSys
- DtActuatorWinches
- DtActuatorCatapults
- DtActuatorCranes
- DtActuatorLaunchers
- DtAuxCraftLifeBoat
- DtAuxCraftLandingCraft
- DtAuxCraftEjectionSeat

[Alphabetic index](#) [Hierarchy of classes](#)

In file recordsetpdu.h:

class DtRecordSetPdu: public DtPdu

class DtRecordSetPdu: DtRecordSetPdu is an abstract base class used by PDUs which require the record set functionality

Inheritance:

DtRecordSetPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

- virtual ~DtRecordSetPdu ()
destructor
- [DtRecordSetPdu](#)& operator= (const [DtRecordSetPdu](#)& orig)
Assignment operator
- virtual [DtBoolean](#) setNumRecordSets (DtU32 numSets)
Set/Get the number of record sets
- virtual DtU32 numRecordSets () const
- virtual [DtBoolean](#) setRecordSet (int index, const [DtRecordSet](#)& recSet)
Set/Get a record set
- virtual [DtBoolean](#) getRecordSet (int index, [DtRecordSet](#)& recSet) const
- virtual void printData () const
Print the data contained by this PDU

Protected Fields

- int mySaveRecCount
These are used by recordSetOffset() to save the value and index of the last offset which was computed in case it can be used again
- int mySaveBuffCount

Protected Methods

- [DtRecordSetPdu](#) ()
Constructors (protected because this is an abstract base class)
- [DtRecordSetPdu](#) (const [DtRecordSetPdu](#)& orig)
- [DtBoolean](#) validateRecordSets (DtNetRecordSetPdu& netStruct)
This function checks the data records for consistency between the actual size of the PDU and the size suggested by the PDU

virtual DtNetRecordSetPdu*	netRecordSetPduStruct () const <i>Returns a pointer to the beginning of the record set data struct</i>
virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>
virtual int	netRecordSetLength () const <i>Returns the length of data used by the record sets beginning at the location specified by netRecordSetPduStruct()</i>
virtual int	recordSetOffset (int recordSetIndex) const <i>Returns a buffer offset to the given record set</i>
virtual DtRecordSetPdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type</i>

Inherited from [DtPdu](#):

Public Methods

virtual void*	packet () const
virtual int	status () const
virtual void	setVersion (int vers)
virtual int	protocolVersion () const
virtual void	setExerciseld (int id)
virtual int	exerciseld () const
virtual DtPduKind	kind () const
virtual DtProtocolFamily	protocolFamily () const
virtual void	setTimeStamp (DtTime time, DtTimeStampType type = DtTimeStampRelative)
virtual DtTime	timeStamp () const
virtual DtTimeStampType	timeStampType () const
virtual DtTime	guessTimeValid (DtTime referenceTime) const
virtual void	print () const
virtual void	printHeader () const
inline int	exerciseld () const
inline DtPduKind	kind () const
inline DtNetPduHeader*	netHeader () const
inline void*	packet () const
inline DtProtocolFamily	protocolFamily () const
inline int	protocolVersion () const

- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtPduKind](#) **internalGetPduKind**() const
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

class DtRecordSetPdu: DtRecordSetPdu is an abstract base class used by PDUs which require the record set functionality

- virtual ~DtRecordSetPdu()
destructor
- [DtRecordSetPdu](#)& operator=(const [DtRecordSetPdu](#)& orig)
Assignment operator
- virtual [DtBoolean](#) setNumRecordSets(DtU32 numSets)
Set/Get the number of record sets
- virtual DtU32 numRecordSets() const
- virtual [DtBoolean](#) setRecordSet(int index, const [DtRecordSet](#)& recSet)
Set/Get a record set. If the index is out of bounds, then DtFalse is returned. If the space would exceed the maximum space permitted for a PDU (for a set) then DtFalse may be returned.
- virtual [DtBoolean](#) getRecordSet(int index, [DtRecordSet](#)& recSet) const
- virtual void printData() const
Print the data contained by this PDU

DtRecordSetPdu()

Constructors (protected because this is an abstract base class)

DtRecordSetPdu(const [DtRecordSetPdu](#)& orig)

[DtBoolean](#) validateRecordSets(DtNetRecordSetPdu& netStruct)

This function checks the data records for consistency between the actual size of the PDU and the size suggested by the PDU. It also tries to fix any problems that are detected. This function is intended to be called in the PDU constructor that accepts network data immediately following initpdu. Non-virtual since meant to be called from ctor.

virtual DtNetRecordSetPdu* netRecordSetPduStruct() const

Returns a pointer to the beginning of the record set data struct. This must be implemented by the derived class because the location of the record sets within the PDU can vary.

virtual char* var() const

Returns a pointer to the first byte of variable length data

virtual int netRecordSetLength() const

Returns the length of data used by the record sets beginning at the location specified by netRecordSetPduStruct(). This can be used by the derived class if any data follows the record set data.

virtual int recordSetOffset(int recordSetIndex) const

Returns a buffer offset to the given record set

[DtRecordSetPdu](#)* self() const

Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type

int mySaveRecCount

These are used by recordSetOffset() to save the value and index of the last offset which was computed in case it can be used again

int mySaveBuffCount

Direct child classes:

[DtTransferControlRequestPdu](#)

[DtSetRecordPdu](#)

[DtRecordPdu](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file recordset.h:

class DtRecordSet

class DtRecordSet: Instances of DtRecordSet are used to contain the record sets used by the PDU classes which inherit from the record set PDU base class

Public Methods

	DtRecordSet () <i>Constructors</i>
	DtRecordSet (const DtRecordSet & orig)
virtual	~DtRecordSet () <i>Destructor</i>
DtRecordSet &	operator= (const DtRecordSet & orig) <i>Assignment operators</i>
DtRecordSet &	operator= (const DtNetRecordSet& orig)
virtual DtBoolean	setDatumID (DtDatumParam value) <i>Set/Get the datum ID</i>
virtual DtDatumParam	datumID () const
virtual void	setSerialNumber (DtU32 value) <i>Set/Get the serial number</i>
virtual DtU32	serialNumber () const
virtual DtBoolean	setRecordBitLength (DtU16 bitLength) <i>Set/Get the record length in bits</i>
virtual DtU16	recordBitLength () const
virtual DtBoolean	setRecordByteLength (int byteLength) <i>Set/Get the record length in bytes</i>
virtual int	recordByteLength () const
virtual DtBoolean	setRecordCount (DtU16 bytes) <i>Set/Get the number of of records in this set When the number of records is changed, old records are not preserved</i>
virtual DtU16	recordCount () const
virtual DtBoolean	setDataBytes (int recordIndex, const void* data) <i>Set/Get multiple data bytes</i>
virtual DtBoolean	getDataBytes (int recordIndex, void* buff) const
virtual DtBoolean	setDataByte (int recordIndex, DtU8 value, int byteIndex = 0) <i>Set/Get a data byte</i>
virtual DtBoolean	getDataByte (int recordIndex, DtU8& value, int byteIndex = 0) const
virtual DtBoolean	setDataInt16 (int recordIndex, DtInt16& value) <i>Set/Get data in signed 16-bit integer format</i>
virtual DtBoolean	getDataInt16 (int recordIndex, DtInt16& value) const

virtual DtBoolean	setDataU16 (int recordIndex, DtU16& value) <i>Set/Get data in unsigned 16-bit integer format</i>
virtual DtBoolean	getDataU16 (int recordIndex, DtU16& value) const
virtual DtBoolean	setDataInt32 (int recordIndex, DtInt32& value) <i>Set/Get data in signed 32-bit integer format</i>
virtual DtBoolean	getDataInt32 (int recordIndex, DtInt32& value) const
virtual DtBoolean	setDataU32 (int recordIndex, DtU32& value) <i>Set/Get data in unsigned 32-bit integer format</i>
virtual DtBoolean	getDataU32 (int recordIndex, DtU32& value) const
virtual DtBoolean	setDataFloat32 (int recordIndex, DtFloat32& value) <i>Set/Get data in 32-bit floating point format</i>
virtual DtBoolean	getDataFloat32 (int recordIndex, DtFloat32& value) const
virtual DtBoolean	setDataFloat64 (int recordIndex, DtFloat64& value) <i>Set/Get data in 64-bit floating point format</i>
virtual DtBoolean	getDataFloat64 (int recordIndex, DtFloat64& value) const
virtual int	recordSetLength () const <i>Get the total number of bytes required for the network representation of the record set</i>
virtual void	print () const <i>Print this record</i>
virtual void	unloadNetRecordSet (const DtNetRecordSet& record) <i>Load/unload data from a packet record</i>
virtual void	loadNetRecordSet (DtNetRecordSet& record) const

Protected Fields

DtDatumParam	myDatumID
DtU32	mySerialNumber
DtU16	myRecordLength
DtU16	myRecordCount
char*	myData

Documentation

class DtRecordSet: Instances of DtRecordSet are used to contain the record sets used by the PDU classes which inherit from the record set PDU base class

DtRecordSet ()	Constructors
DtRecordSet (const DtRecordSet & orig)	
virtual ~DtRecordSet ()	Destructor

• `DtRecordSet& operator=(const DtRecordSet& orig)`

Assignment operators

• `DtRecordSet& operator=(const DtNetRecordSet& orig)`

• `virtual DtBoolean setDatumID(DtDatumParam value)`

Set/Get the datum ID

• `virtual DtDatumParam datumID() const`

• `virtual void setSerialNumber(DtU32 value)`

Set/Get the serial number

• `virtual DtU32 serialNumber() const`

• `virtual DtBoolean setRecordBitLength(DtU16 bitLength)`

Set/Get the record length in bits. `setRecordBitLength()` also resets the number of records to zero, and may return `DtFalse` (with no action) if the length is invalid.

• `virtual DtU16 recordBitLength() const`

• `virtual DtBoolean setRecordByteLength(int byteLength)`

Set/Get the record length in bytes. `setRecordLength()` also resets the number of records to zero, and may return `DtFalse` (with no action) if the length is invalid.

• `virtual int recordByteLength() const`

• `virtual DtBoolean setRecordCount(DtU16 bytes)`

Set/Get the number of of records in this set When the number of records is changed, old records are not preserved

• `virtual DtU16 recordCount() const`

• `virtual DtBoolean setDataBytes(int recordIndex, const void* data)`

Set/Get multiple data bytes. Provide a pointer and data is copied to and from that memory. Will copy `recordByteLength()` bytes on set/get. These functions return `DtFalse` if the record index is out of range.

• `virtual DtBoolean getDataBytes(int recordIndex, void* buff) const`

• `virtual DtBoolean setDataByte( int recordIndex, DtU8 value, int byteIndex = 0)`

Set/Get a data byte. These functions return `DtFalse` if the record index or byte index is out of range.

• `virtual DtBoolean getDataByte( int recordIndex, DtU8& value, int byteIndex = 0) const`

• `virtual DtBoolean setDataInt16(int recordIndex, DtInt16& value)`

Set/Get data in signed 16-bit integer format. These functions return `DtFalse` if the record length is not 16-bits, or if the record index is out of range.

• `virtual DtBoolean getDataInt16(int recordIndex, DtInt16& value) const`

• `virtual DtBoolean setDataU16(int recordIndex, DtU16& value)`

Set/Get data in unsigned 16-bit integer format. These functions return `DtFalse` if the record length is not 16-bits, or if the record index is out of range.

• `virtual DtBoolean getDataU16(int recordIndex, DtU16& value) const`

- virtual [DtBoolean](#) setDataInt32(int recordIndex, DtInt32& value)
 Set/Get data in signed 32-bit integer format. These functions return DtFalse if the record length is not 32-bits, or if the record index is out of range.
- virtual [DtBoolean](#) getDataInt32(int recordIndex, DtInt32& value) const
- virtual [DtBoolean](#) setDataU32(int recordIndex, DtU32& value)
 Set/Get data in unsigned 32-bit integer format. These functions return DtFalse if the record length is not 32-bits, or if the record index is out of range.
- virtual [DtBoolean](#) getDataU32(int recordIndex, DtU32& value) const
- virtual [DtBoolean](#) setDataFloat32(int recordIndex, DtFloat32& value)
 Set/Get data in 32-bit floating point format. These functions return DtFalse if the record length is not 32-bits, or if the record index is out of range.
- virtual [DtBoolean](#) getDataFloat32(int recordIndex, DtFloat32& value) const
- virtual [DtBoolean](#) setDataFloat64(int recordIndex, DtFloat64& value)
 Set/Get data in 64-bit floating point format. These functions return DtFalse if the record length is not 64-bits, or if the record index is out of range.
- virtual [DtBoolean](#) getDataFloat64(int recordIndex, DtFloat64& value) const
- virtual int recordSetLength() const
 Get the total number of bytes required for the network representation of the record set
- virtual void print() const
 Print this record
- virtual void unloadNetRecordSet(const DtNetRecordSet& record)
 Load/unload data from a packet record
- virtual void loadNetRecordSet(DtNetRecordSet& record) const
- [DtDatumParam](#) myDatumID
- DtU32 mySerialNumber
- DtU16 myRecordLength
- DtU16 myRecordCount
- char* myData

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file tranctlpdu.h:

class DtTransferControlRequestPdu: public DtRecordSetPdu

Inheritance:

DtTransferControlRequestPdu - DtRecordSetPdu - DtPdu - DtBaseMsg

Public Methods

	DtTransferControlRequestPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtTransferControlRequestPdu (const DtNetTransferControlRequestPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtTransferControlRequestPdu (const DtTransferControlRequestPdu & orig)
 virtual	~DtTransferControlRequestPdu () <i>destructor</i>
 DtTransferControlRequestPdu &	operator= (const DtTransferControlRequestPdu & orig) <i>Assignment operator</i>
 virtual void	setOriginatingEntityID (const DtEntityIdentifier & entity) <i>Set/Get originating entity identifier</i>
 virtual const DtEntityIdentifier &	originatingEntityID () const
 virtual void	setReceivingEntityID (const DtEntityIdentifier & entity) <i>Set/Get receiving entity identifier</i>
 virtual const DtEntityIdentifier &	receivingEntityID () const
 virtual void	setRequestID (DtU32 id) <i>Set/Get request ID</i>
 virtual DtU32	requestID () const
 virtual void	setReqRelService (enum DtRequiredReliabilityService service) <i>Set/Get required reliability service</i>
 virtual DtRequiredReliabilityService	reqRelService () const
 virtual void	setTransferType (DtTransferType type) <i>Set/Get transfer type</i>
 virtual DtTransferType	transferType () const

virtual void	setTransferredEntityID (const DtEntityIdentifier & entity) <i>Set/Get transferred entity identifier</i>
virtual const DtEntityIdentifier &	transferredEntityID () const
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtTransferControlRequestPdu</i>
static void	addCallback (DtExerciseConn * conn, DtTransferControlRequestPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtTransferControlRequestPduCb cb, void* usr)

Protected Fields

static int	theStaticSize <i>Size of PDU with zero data bytes</i>
DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>

Protected Methods

virtual DtNetRecordSetPdu*	netRecordSetPduStruct () const <i>Returns a pointer to the beginning of the record set data struct</i>
virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
virtual DtNetTransferControlRequestPdu*	netPduStruct () const <i>Casts the netPdu element to our network struct Non-virtual since you can't override based on return type</i>
DtTransferControlRequestPdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type</i>

Inherited from [DtRecordSetPdu](#):

Public Methods

virtual DtBoolean	setNumRecordSets (DtU32 numSets)
virtual DtU32	numRecordSets () const
virtual DtBoolean	setRecordSet (int index, const DtRecordSet & recSet)
virtual DtBoolean	getRecordSet (int index, DtRecordSet & recSet) const

Protected Fields

int **mySaveRecCount**

int **mySaveBuffCount**

Protected Methods

[DtBoolean](#) **validateRecordSets**(DtNetRecordSetPdu& netStruct)

virtual char* **var**() const

virtual int **netRecordSetLength**() const

virtual int **recordSetOffset**(int recordSetIndex) const

Inherited from [DtPdu](#):

Public Methods

virtual void* **packet**() const

virtual int **status**() const

virtual void **setVersion**(int vers)

virtual int **protocolVersion**() const

virtual void **setExerciseld**(int id)

virtual int **exerciseld**() const

virtual [DtPduKind](#) **kind**() const

virtual [DtProtocolFamily](#) **protocolFamily**() const

virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

virtual DtTime **timeStamp**() const

virtual [DtTimeStampType](#) **timeStampType**() const

virtual DtTime **guessTimeValid**(DtTime referenceTime) const

virtual void **print**() const

virtual void **printHeader**() const

inline int **exerciseld**() const

inline [DtPduKind](#) **kind**() const

inline DtNetPduHeader* **netHeader**() const

inline void* **packet**() const

- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader()** const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtTransferControlRequestPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtTransferControlRequestPdu**(const DtNetTransferControlRequestPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtTransferControlRequestPdu**(const [DtTransferControlRequestPdu](#)& orig)
- virtual ~DtTransferControlRequestPdu**()
destructor
- [DtTransferControlRequestPdu](#)**& operator=(const [DtTransferControlRequestPdu](#)& orig)
Assignment operator
- virtual void setOriginatingEntityID**(const [DtEntityIdentifier](#)& entity)
Set/Get originating entity identifier
- virtual const [DtEntityIdentifier](#)**& originatingEntityID() const
- virtual void setReceivingEntityID**(const [DtEntityIdentifier](#)& entity)
Set/Get receiving entity identifier

- `virtual const DtEntityIdentifier& receivingEntityID() const`
- `virtual void setRequestID(DtU32 id)`
Set/Get request ID
- `virtual DtU32 requestID() const`
- `virtual void setReqRelService(enum DtRequiredReliabilityService service)`
Set/Get required reliability service
- `virtual DtRequiredReliabilityService reqRelService() const`
- `virtual void setTransferType(DtTransferType type)`
Set/Get transfer type
- `virtual DtTransferType transferType() const`
- `virtual void setTransferredEntityID(const DtEntityIdentifier& entity)`
Set/Get transferred entity identifier
- `virtual const DtEntityIdentifier& transferredEntityID() const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtTransferControlRequestPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtTransferControlRequestPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtTransferControlRequestPduCb cb, void* usr)`
- `virtual DtNetRecordSetPdu* netRecordSetPduStruct() const`
Returns a pointer to the beginning of the record set data struct
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetTransferControlRequestPdu* netPduStruct() const`
Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- `DtTransferControlRequestPdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- `static int theStaticSize`
Size of PDU with zero data bytes
- `DtEntityIdentifier myEntityID`
Temporary objects for reference returns

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtTransferType

Transfer Control Request PDU enums

-  DtOtherTransfer
 -  DtCtlReqTransferEntity
 -  DtAppReqTransferEntity
 -  DtReqExchangeEntity
 -  DtCtlReqTransferEnvProc
 -  DtAppReqTransferEnvProc
 -  DtReqExchangeEnvProc
-

Documentation

Transfer Control Request PDU enums

-  DtOtherTransfer
-  DtCtlReqTransferEntity
-  DtAppReqTransferEntity
-  DtReqExchangeEntity
-  DtCtlReqTransferEnvProc
-  DtAppReqTransferEnvProc
-  DtReqExchangeEnvProc

In file setrecordpdu.h:

class DtSetRecordPdu: public DtRecordSetPdu

Inheritance:

DtSetRecordPdu - DtRecordSetPdu - DtPdu - DtBaseMsg

Public Methods

	DtSetRecordPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtSetRecordPdu (const DtNetSetRecordPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtSetRecordPdu (const DtSetRecordPdu & orig)
 virtual	~DtSetRecordPdu () <i>destructor</i>
 DtSetRecordPdu &	operator= (const DtSetRecordPdu & orig) <i>Assignment operator</i>
 virtual void	setOriginatingEntityID (const DtEntityIdentifier & entity) <i>Set/Get originating entity identifier</i>
 virtual const DtEntityIdentifier &	originatingEntityID () const
 virtual void	setReceivingEntityID (const DtEntityIdentifier & entity) <i>Set/Get receiving entity identifier</i>
 virtual const DtEntityIdentifier &	receivingEntityID () const
 virtual void	setRequestID (DtU32 id) <i>Set/Get request ID</i>
 virtual DtU32	requestID () const
 virtual void	setReqRelService (enum DtRequiredReliabilityService service) <i>Set/Get required reliability service</i>
 virtual DtRequiredReliabilityService	reqRelService () const
 virtual void	printData () const <i>Print the data contained by this PDU</i>
 static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtSetRecordPdu</i>
 static void	addCallback (DtExerciseConn * conn, DtSetRecordPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
 static void	removeCallback (DtExerciseConn * conn, DtSetRecordPduCb cb, void* usr)

Protected Fields

- static int **theStaticSize**
Size of PDU with zero data bytes
- [DtEntityIdentifier](#) **myEntityID**
Temporary objects for reference returns

Protected Methods

- virtual DtNetRecordSetPdu* **netRecordSetPduStruct** () const
Returns a pointer to the beginning of the record set data struct
 - virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind
 - virtual DtNetSetRecordPdu* **netPduStruct** () const
Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
 - [DtSetRecordPdu](#)* **self** () const
Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
-

Inherited from [DtRecordSetPdu](#):

Public Methods

- virtual [DtBoolean](#) **setNumRecordSets**(DtU32 numSets)
- virtual DtU32 **numRecordSets**() const
- virtual [DtBoolean](#) **setRecordSet**(int index, const [DtRecordSet](#)& recSet)
- virtual [DtBoolean](#) **getRecordSet**(int index, [DtRecordSet](#)& recSet) const

Protected Fields

- int **mySaveRecCount**
- int **mySaveBuffCount**

Protected Methods

- [DtBoolean](#) **validateRecordSets**(DtNetRecordSetPdu& netStruct)
 - virtual char* **var**() const
 - virtual int **netRecordSetLength**() const
 - virtual int **recordSetOffset**(int recordSetIndex) const
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtSetRecordPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Constructors

- DtSetRecordPdu**(const DtNetSetRecordPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

- DtSetRecordPdu**(const [DtSetRecordPdu](#)& orig)

- virtual **~DtSetRecordPdu**()

destructor

- [DtSetRecordPdu](#)& operator=(const [DtSetRecordPdu](#)& orig)

Assignment operator

- virtual void **setOriginatingEntityID**(const [DtEntityIdentifier](#)& entity)

Set/Get originating entity identifier

- virtual const [DtEntityIdentifier](#)& originatingEntityID() const

- virtual void **setReceivingEntityID**(const [DtEntityIdentifier](#)& entity)

Set/Get receiving entity identifier

- virtual const [DtEntityIdentifier](#)& receivingEntityID() const

- virtual void **setRequestID**(DtU32 id)

Set/Get request ID

- virtual DtU32 **requestID**() const

- virtual void **setReqRelService**(enum [DtRequiredReliabilityService](#) service)

Set/Get required reliability service

- virtual [DtRequiredReliabilityService](#) **reqRelService**() const

- virtual void **printData**() const

Print the data contained by this PDU

- static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtSetRecordPdu. Caller is responsible for deletion.

- `static void addCallback(DtExerciseConn* conn, DtSetRecordPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtSetRecordPduCb cb, void* usr)`
- `virtual DtNetRecordSetPdu* netRecordSetPduStruct() const`
Returns a pointer to the beginning of the record set data struct
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetSetRecordPdu* netPduStruct() const`
Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- `DtSetRecordPdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- `static int theStaticSize`
Size of PDU with zero data bytes
- `DtEntityIdentifier myEntityID`
Temporary objects for reference returns

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file recordpdu.h:

class DtRecordPdu: public DtRecordSetPdu

Inheritance:

DtRecordPdu - DtRecordSetPdu - DtPdu - DtBaseMsg

Public Methods

	DtRecordPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtRecordPdu (const DtNetRecordPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtRecordPdu (const DtRecordPdu & orig)
 virtual	~DtRecordPdu () <i>destructor</i>
 DtRecordPdu &	operator= (const DtRecordPdu & orig) <i>Assignment operator</i>
 virtual void	setOriginatingEntityID (const DtEntityIdentifier & entity) <i>Set/Get originating entity identifier</i>
 virtual const DtEntityIdentifier &	originatingEntityID () const
 virtual void	setReceivingEntityID (const DtEntityIdentifier & entity) <i>Set/Get receiving entity identifier</i>
 virtual const DtEntityIdentifier &	receivingEntityID () const
 virtual void	setRequestID (DtU32 id) <i>Set/Get request ID</i>
 virtual DtU32	requestID () const
 virtual void	setReqRelService (enum DtRequiredReliabilityService service) <i>Set/Get required reliability service</i>
 virtual DtRequiredReliabilityService	reqRelService () const
 virtual void	setEventType (DtEventType type) <i>Set/Get the event type</i>
 virtual DtEventType	eventType () const
 virtual void	setResponseSerialNumber (DtU32 num) <i>Set/Get the response serial number</i>
 virtual DtU32	responseSerialNumber () const
 virtual void	printData () const <i>Print the data contained by this PDU</i>

- static [DtPdu](#)* **create** (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Create a DtRecordPdu
- static void **addCallback** ([DtExerciseConn](#)* conn, DtRecordPduCb cb, void* usr)
Add (register)/Remove (deregister) function to be called when pdu is received
- static void **removeCallback** ([DtExerciseConn](#)* conn, DtRecordPduCb cb, void* usr)

Protected Fields

- static int **theStaticSize**
Size of PDU with zero data bytes
- [DtEntityIdentifier](#) **myEntityID**
Temporary objects for reference returns

Protected Methods

- virtual DtNetRecordSetPdu* **netRecordSetPduStruct** () const
Returns a pointer to the beginning of the record set data struct
- virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind
- virtual DtNetRecordPdu* **netPduStruct** () const
Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- [DtRecordPdu](#)* **self** () const
Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type

Inherited from [DtRecordSetPdu](#):

Public Methods

- virtual [DtBoolean](#) **setNumRecordSets**(DtU32 numSets)
- virtual DtU32 **numRecordSets**() const
- virtual [DtBoolean](#) **setRecordSet**(int index, const [DtRecordSet](#)& recSet)
- virtual [DtBoolean](#) **getRecordSet**(int index, [DtRecordSet](#)& recSet) const

Protected Fields

- int **mySaveRecCount**
- int **mySaveBuffCount**

Protected Methods

- [DtBoolean](#) **validateRecordSets**(DtNetRecordSetPdu& netStruct)
 - virtual char* **var**() const
 - virtual int **netRecordSetLength**() const
 - virtual int **recordSetOffset**(int recordSetIndex) const
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)

- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtRecordPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtRecordPdu(const DtNetRecordPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtRecordPdu(const [DtRecordPdu](#)& orig)
- virtual ~DtRecordPdu()
destructor
- [DtRecordPdu](#)& operator=(const [DtRecordPdu](#)& orig)
Assignment operator
- virtual void setOriginatingEntityID(const [DtEntityIdentifier](#)& entity)
Set/Get originating entity identifier
- virtual const [DtEntityIdentifier](#)& originatingEntityID() const
- virtual void setReceivingEntityID(const [DtEntityIdentifier](#)& entity)
Set/Get receiving entity identifier
- virtual const [DtEntityIdentifier](#)& receivingEntityID() const
- virtual void setRequestID(DtU32 id)
Set/Get request ID
- virtual DtU32 requestID() const
- virtual void setReqRelService(enum [DtRequiredReliabilityService](#) service)
Set/Get required reliability service

- `virtual DtRequiredReliabilityService reqRelService() const`
- `virtual void setEventType(DtEventType type)`
Set/Get the event type
- `virtual DtEventType eventType() const`
- `virtual void setResponseSerialNumber(DtU32 num)`
Set/Get the response serial number
- `virtual DtU32 responseSerialNumber() const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtRecordPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtRecordPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtRecordPduCb cb, void* usr)`
- `virtual DtNetRecordSetPdu* netRecordSetPduStruct() const`
Returns a pointer to the beginning of the record set data struct
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetRecordPdu* netPduStruct() const`
Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- `DtRecordPdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- `static int theStaticSize`
Size of PDU with zero data bytes
- `DtEntityIdentifier myEntityID`
Temporary objects for reference returns

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file recquerypdu.h:

class DtRecordQueryPdu: public DtPdu

Inheritance:

DtRecordQueryPdu - DtPdu - DtBaseMsg

Public Methods

	DtRecordQueryPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtRecordQueryPdu (const DtNetRecordQueryPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtRecordQueryPdu (const DtRecordQueryPdu & orig)
 virtual	~DtRecordQueryPdu () <i>destructor</i>
 DtRecordQueryPdu &	operator= (const DtRecordQueryPdu & orig) <i>Assignment operator</i>
 virtual void	setOriginatingEntityID (const DtEntityIdentifier & entity) <i>Set/Get originating entity identifier</i>
 virtual const DtEntityIdentifier &	originatingEntityID () const
 virtual void	setReceivingEntityID (const DtEntityIdentifier & entity) <i>Set/Get receiving entity identifier</i>
 virtual const DtEntityIdentifier &	receivingEntityID () const
 virtual void	setRequestID (DtU32 id) <i>Set/Get request ID</i>
 virtual DtU32	requestID () const
 virtual void	setReqRelService (enum DtRequiredReliabilityService service) <i>Set/Get required reliability service</i>
 virtual DtRequiredReliabilityService	reqRelService () const
 virtual void	setEventType (DtEventType type) <i>Set/Get the event type</i>
 virtual DtEventType	eventType () const
 virtual void	setTime (DtU32 value) <i>Set/Get the time</i>
 virtual DtU32	time () const
 virtual DtBoolean	setNumRecords (DtU32 num) <i>Set/Get the number of records to be queried for</i>

virtual DtU32	numRecords () const
virtual DtBoolean	setRecordID (int index, DtDatumParam value) <i>Set/Get a record ID</i>
virtual DtBoolean	getRecordID (int index, DtDatumParam & value) const
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtRecordQueryPdu</i>
static void	addCallback (DtExerciseConn * conn, DtRecordQueryPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtRecordQueryPduCb cb, void* usr)

Protected Fields

static int	theStaticSize <i>Size of PDU with zero data bytes</i>
DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
virtual DtNetRecordQueryPdu*	netPduStruct () const <i>Casts the netPdu element to our network struct Non-virtual since you can't override based on return type</i>
DtRecordQueryPdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type</i>
virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>

Inherited from [DtPdu](#):

Public Methods

virtual void*	packet () const
virtual int	status () const
virtual void	setVersion (int vers)
virtual int	protocolVersion () const
virtual void	setExerciseld (int id)
virtual int	exerciseld () const

- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const

- virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtRecordQueryPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors

- `DtRecordQueryPdu(const DtNetRecordQueryPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtRecordQueryPdu(const DtRecordQueryPdu& orig)`
- `virtual ~DtRecordQueryPdu()`
destructor
- `DtRecordQueryPdu& operator=(const DtRecordQueryPdu& orig)`
Assignment operator
- `virtual void setOriginatingEntityID(const DtEntityIdentifier& entity)`
Set/Get originating entity identifier
- `virtual const DtEntityIdentifier& originatingEntityID() const`
- `virtual void setReceivingEntityID(const DtEntityIdentifier& entity)`
Set/Get receiving entity identifier
- `virtual const DtEntityIdentifier& receivingEntityID() const`
- `virtual void setRequestID(DtU32 id)`
Set/Get request ID
- `virtual DtU32 requestID() const`
- `virtual void setReqRelService(enum DtRequiredReliabilityService service)`
Set/Get required reliability service
- `virtual DtRequiredReliabilityService reqRelService() const`
- `virtual void setEventType(DtEventType type)`
Set/Get the event type
- `virtual DtEventType eventType() const`
- `virtual void setTime(DtU32 value)`
Set/Get the time
- `virtual DtU32 time() const`
- `virtual DtBoolean setNumRecords(DtU32 num)`
Set/Get the number of records to be queried for. Set may return DtFalse if the number of records is too large.
- `virtual DtU32 numRecords() const`
- `virtual DtBoolean setRecordID(int index, DtDatumParam value)`
Set/Get a record ID. These methods return DtFalse if the index is out of range.
- `virtual DtBoolean getRecordID(int index, DtDatumParam& value) const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer =`

DtUSE_INTERNAL_BUFFER)

Create a DtRecordQueryPdu. Caller is responsible for deletion.

• `static void addCallback(DtExerciseConn* conn, DtRecordQueryPduCb cb, void* usr)`

Add (register)/Remove (deregister) function to be called when pdu is received

• `static void removeCallback(DtExerciseConn* conn, DtRecordQueryPduCb cb, void* usr)`

• `virtual DtPduKind internalGetPduKind() const`

Returns the internal PDU kind

• `virtual DtNetRecordQueryPdu* netPduStruct() const`

Casts the netPdu element to our network struct Non-virtual since you can't override based on return type

• `DtRecordQueryPdu* self() const`

Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type

• `virtual char* var() const`

Returns a pointer to the first byte of variable length data

• `static int theStaticSize`

Size of PDU with zero data bytes

• `DtEntityIdentifier myEntityID`

Temporary objects for reference returns

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file receivepdu.h:

class DtReceiverPdu: public DtPdu

Inheritance:

DtReceiverPdu - DtPdu - DtBaseMsg

Public Methods

	DtReceiverPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtReceiverPdu (const DtNetReceiverPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static void	addCallback (DtExerciseConn *conn, DtReceiverPduCb cb, void *usr)
 static void	removeCallback (DtExerciseConn *conn, DtReceiverPduCb cb, void *usr)
 const DtEntityIdentifier &	entityId () const
 unsigned int	radioid () const
 DtRadioReceiverState	receiverState () const
 float	receivedPower () const
 const DtEntityIdentifier &	transmitEntityId () const
 unsigned int	transmitRadioid () const
 void	setEntityId (const DtEntityIdentifier &id)
 void	setRadioid (unsigned int id)
 void	setReceiverState (DtRadioReceiverState state)
 void	setReceivedPower (float power)
 void	setTransmitEntityId (const DtEntityIdentifier &id)
 void	setTransmitRadioid (unsigned int id)
 void	printData () const

Protected Fields

 [DtEntityIdentifier](#) **theld**

 [DtEntityIdentifier](#) **transld**

Protected Methods

 [DtPduKind](#) **internalGetPduKind** () const

Inherited from [DtPdu](#):

Public Methods

 [DtPdu](#) & **operator=**(const [DtPdu](#) &pdu)

 virtual void* **packet**() const

 virtual int **status**() const

 virtual void **setVersion**(int vers)

 virtual int **protocolVersion**() const

 virtual void **setExerciseld**(int id)

 virtual int **exerciseld**() const

 virtual [DtPduKind](#) **kind**() const

 virtual [DtProtocolFamily](#) **protocolFamily**() const

 virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

 virtual DtTime **timeStamp**() const

 virtual [DtTimeStampType](#) **timeStampType**() const

 virtual DtTime **guessTimeValid**(DtTime referenceTime) const

 virtual void **print**() const

 virtual void **printHeader**() const

 inline int **exerciseld**() const

 inline [DtPduKind](#) **kind**() const

 inline DtNetPduHeader* **netHeader**() const

 inline void* **packet**() const

 inline [DtProtocolFamily](#) **protocolFamily**() const

- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtReceiverPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtReceiverPdu(const DtNetReceiverPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void addCallback([DtExerciseConn](#) *conn, DtReceiverPduCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtReceiverPduCb cb, void *usr)
- const [DtEntityIdentifier](#)& entityId() const
- unsigned int radioId() const
- [DtRadioReceiverState](#) receiverState() const
- float receivedPower() const
- const [DtEntityIdentifier](#)& transmitEntityId() const
- unsigned int transmitRadioId() const

- `void setEntityId(const DtEntityIdentifier &id)`
- `void setRadioId(unsigned int id)`
- `void setReceiverState(DtRadioReceiverState state)`
- `void setReceivedPower(float power)`
- `void setTransmitEntityId(const DtEntityIdentifier &id)`
- `void setTransmitRadioId(unsigned int id)`
- `void printData() const`
- `DtPduKind internalGetPduKind() const`
- `DtEntityIdentifier theId`
- `DtEntityIdentifier transId`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtRadioReceiverState

-
-  DtRadioOff
 -  DtRadioOnNotReceiving
 -  DtRadioOn
-

Documentation

-  DtRadioOff
-  DtRadioOnNotReceiving
-  DtRadioOn

In file pointobjstpdu.h:

class DtPointObjectStatePdu: public DtPdu

Inheritance:

DtPointObjectStatePdu - DtPdu - DtBaseMsg

Public Methods

	DtPointObjectStatePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtPointObjectStatePdu (const DtNetPointObjectStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtPointObjectStatePdu (const DtPointObjectStatePdu & orig)
 virtual	~DtPointObjectStatePdu () <i>destructor</i>
 DtPointObjectStatePdu &	operator= (const DtPointObjectStatePdu & orig) <i>Assignment operator</i>
 virtual void	setObjectID (const DtEntityIdentifier & id) <i>Set/Get object ID</i>
 virtual const DtEntityIdentifier &	objectID () const
 virtual void	setReferencedObjectID (const DtEntityIdentifier & id) <i>Set/Get referenced object ID</i>
 virtual const DtEntityIdentifier &	referencedObjectID () const
 virtual void	setUpdateNumber (DtU16 field) <i>Set/Get update number</i>
 virtual DtU16	updateNumber () const
 virtual void	setForceID (DtForceType force) <i>Set/Get the force ID</i>
 virtual DtForceType	forceID () const
 virtual void	setModifications (DtU8 value) <i>Set/Get the modifications (by 8-bit integer)</i>
 virtual DtU8	modifications () const
 virtual void	setModificationsField (DtModificationField field, DtBoolean value) <i>Set/Get the modifications (by field)</i>
 virtual DtBoolean	modificationsField (DtModificationField field) const
 virtual void	setObjectType (const DtObjectType & type) <i>Set/Get the object type</i>

virtual const DtObjectType &	objectType () const
virtual void	setLocation (DtConstVector coords) <i>Set/Get location</i>
virtual DtConstVector	location () const
virtual void	setOrientation (const DtTaitBryan & orientation) <i>Set/Get orientation</i>
virtual const DtTaitBryan &	orientation () const
virtual void	setAppearance (DtU16 app) <i>Set/Get appearance, the lower 16 bits (as an integer)</i>
virtual DtU16	appearance () const
virtual void	setAppearanceAttribute (DtAppearAttribute attr, DtAppearAttrVal val) <i>Set/Get appearance by attribute for the lower 16 bits</i>
virtual DtAppearAttrVal	appearanceAttribute (DtAppearAttribute attr) const
virtual void	setSpecificAppearance (DtU32 app) <i>Set/Get specific appearance, the upper 32 bits (as a 32 bit integer)</i>
virtual DtU32	specificAppearance () const
virtual void	setSpecificAppearanceAttribute (const DtAppearAttribute attr, const DtAppearAttrVal val) <i>Set/Get specific appearance by attribute, the upper 32 bits</i>
virtual DtAppearAttrVal	specificAppearanceAttribute (const DtAppearAttribute attr) const
virtual void	setRequesterID (const DtSimulationAddress& type) <i>Set/Get the requester ID</i>
virtual const DtSimulationAddress&	requesterID () const
virtual void	setReceivingID (const DtSimulationAddress& type) <i>Set/Get the receiving ID</i>
virtual const DtSimulationAddress&	receivingID () const
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtPointObjectStatePdu</i>
static void	addCallback (DtExerciseConn * conn, DtPointObjectStatePduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtPointObjectStatePduCb cb, void* usr)

Protected Fields

static int	theStaticSize <i>Size of PDU with zero parameter records</i>
DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>
DtObjectType	myObjectType
DtVector	myVector
DtTaitBryan	myEulerAngles

Protected Methods

 virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
 virtual DtNetPointObjectStatePdu*	netPduStruct () const <i>Casts the netPdu element to our network struct</i>
 DtPointObjectStatePdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type</i>

Inherited from [DtPdu](#):

Public Methods

-  virtual void* **packet**() const
-  virtual int **status**() const
-  virtual void **setVersion**(int vers)
-  virtual int **protocolVersion**() const
-  virtual void **setExerciseld**(int id)
-  virtual int **exerciseld**() const
-  virtual [DtPduKind](#) **kind**() const
-  virtual [DtProtocolFamily](#) **protocolFamily**() const
-  virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
-  virtual DtTime **timeStamp**() const
-  virtual [DtTimeStampType](#) **timeStampType**() const
-  virtual DtTime **guessTimeValid**(DtTime referenceTime) const
-  virtual void **print**() const
-  virtual void **printHeader**() const
-  inline int **exerciseld**() const
-  inline [DtPduKind](#) **kind**() const
-  inline DtNetPduHeader* **netHeader**() const
-  inline void* **packet**() const
-  inline [DtProtocolFamily](#) **protocolFamily**() const
-  inline int **protocolVersion**() const

- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**

int myMsgSize

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtPointObjectStatePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtPointObjectStatePdu(const DtNetPointObjectStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtPointObjectStatePdu(const [DtPointObjectStatePdu](#)& orig)
- virtual ~DtPointObjectStatePdu()
destructor
- [DtPointObjectStatePdu](#)& operator=(const [DtPointObjectStatePdu](#)& orig)
Assignment operator
- virtual void setObjectID(const [DtEntityIdentifier](#)& id)
Set/Get object ID
- virtual const [DtEntityIdentifier](#)& objectID() const
- virtual void setReferencedObjectID(const [DtEntityIdentifier](#)& id)
Set/Get referenced object ID
- virtual const [DtEntityIdentifier](#)& referencedObjectID() const
- virtual void setUpdateNumber(DtU16 field)
Set/Get update number
- virtual DtU16 updateNumber() const

- virtual void setForceID([DtForceType](#) force)
Set/Get the force ID
- virtual [DtForceType](#) forceID() const
- virtual void setModifications(DtU8 value)
Set/Get the modifications (by 8-bit integer)
- virtual DtU8 modifications() const
- virtual void setModificationsField([DtModificationField](#) field, [DtBoolean](#) value)
Set/Get the modifications (by field)
- virtual [DtBoolean](#) modificationsField([DtModificationField](#) field) const
- virtual void setObjectType(const [DtObjectType](#)& type)
Set/Get the object type
- virtual const [DtObjectType](#)& objectType() const
- virtual void setLocation(DtConstVector coords)
Set/Get location
- virtual DtConstVector location() const
- virtual void setOrientation(const [DtTaitBryan](#)& orientation)
Set/Get orientation
- virtual const [DtTaitBryan](#)& orientation() const
- virtual void setAppearance(DtU16 app)
Set/Get appearance, the lower 16 bits (as an integer)
- virtual DtU16 appearance() const
- virtual void setAppearanceAttribute([DtAppearAttribute](#) attr, [DtAppearAttrVal](#) val)
Set/Get appearance by attribute for the lower 16 bits
- virtual [DtAppearAttrVal](#) appearanceAttribute([DtAppearAttribute](#) attr) const
- virtual void setSpecificAppearance(DtU32 app)
Set/Get specific appearance, the upper 32 bits (as a 32 bit integer)
- virtual DtU32 specificAppearance() const
- virtual void setSpecificAppearanceAttribute(const [DtAppearAttribute](#) attr, const [DtAppearAttrVal](#) val)
Set/Get specific appearance by attribute, the upper 32 bits
- virtual [DtAppearAttrVal](#) specificAppearanceAttribute(const [DtAppearAttribute](#) attr) const
- virtual void setRequesterID(const DtSimulationAddress& type)
Set/Get the requester ID

- `virtual const DtSimulationAddress& requesterID() const`
- `virtual void setReceivingID(const DtSimulationAddress& type)`
Set/Get the receiving ID
- `virtual const DtSimulationAddress& receivingID() const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtPointObjectStatePdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtPointObjectStatePduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtPointObjectStatePduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetPointObjectStatePdu* netPduStruct() const`
Casts the netPdu element to our network struct
- `DtPointObjectStatePdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
- `static int theStaticSize`
Size of PDU with zero parameter records
- `DtEntityIdentifier myEntityID`
Temporary objects for reference returns
- `DtObjectType myObjectType`
- `DtVector myVector`
- `DtTaitBryan myEulerAngles`
- `DtSimulationAddress mySimulationAddress`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtForceType

Section 4

-  DtForceOther
 -  DtForceFriendly
 -  DtForceOpposing
 -  DtForceNeutral
-

Documentation

Section 4

-  DtForceOther
-  DtForceFriendly
-  DtForceOpposing
-  DtForceNeutral

In file disenums.h:

enum DtModificationField

-
-  DtModLocation
 -  DtModOrientation
-

Documentation

-  DtModLocation
-  DtModOrientation

In file objectType.h:

class DtObjectType

class DtObjectType: DtObjectType is a class which contains information identifying the type of a synthetic environment object

Public Methods

- **DtObjectType ()**
Constructors
- **DtObjectType** (const [DtObjectType](#)& orig)
- [DtObjectType](#)& **operator=** (const [DtObjectType](#)& orig)
Assignment operator
- [DtObjectType](#)& **operator=** (const DtNetObjectType& orig)
- virtual **~DtObjectType ()**
Destructor
- virtual void **setDomain** (DtU8 value)
Set/Get the domain
- virtual DtU8 **domain** () const
- virtual void **setKind** (DtU8 value)
Set/Get the kind
- virtual DtU8 **kind** () const
- virtual void **setCategory** (DtU8 value)
Set/Get the category
- virtual DtU8 **category** () const
- virtual void **setSubcategory** (DtU8 value)
Set/Get the subcategory
- virtual DtU8 **subcategory** () const
- virtual void **print** () const
Print the information contained in the record
- virtual void **unloadNetRecord** (const DtNetObjectType& record)
Load/Unload data from a packet record
- virtual void **loadNetRecord** (DtNetObjectType& record) const

Protected Fields

- DtU8 **myDomain**
Data members:
- DtU8 **myKind**
- DtU8 **myCategory**
- DtU8 **mySubcategory**

Documentation

class DtObjectType: DtObjectType is a class which contains information identifying the type of a synthetic environment object

`DtObjectType()`

Constructors

`DtObjectType(const DtObjectType& orig)`

`DtObjectType& operator=(const DtObjectType& orig)`

Assignment operator

`DtObjectType& operator=(const DtNetObjectType& orig)`

`virtual ~DtObjectType()`

Destructor

`virtual void setDomain(DtU8 value)`

Set/Get the domain

`virtual DtU8 domain() const`

`virtual void setKind(DtU8 value)`

Set/Get the kind

`virtual DtU8 kind() const`

`virtual void setCategory(DtU8 value)`

Set/Get the category

`virtual DtU8 category() const`

`virtual void setSubcategory(DtU8 value)`

Set/Get the subcategory

`virtual DtU8 subcategory() const`

`virtual void print() const`

Print the information contained in the record

`virtual void unloadNetRecord(const DtNetObjectType& record)`

Load/Unload data from a packet record

`virtual void loadNetRecord(DtNetObjectType& record) const`

`DtU8 myDomain`

Data members:

`DtU8 myKind`

`DtU8 myCategory`

`DtU8 mySubcategory`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtAppearAttribute

if you put spaces between the "<<" and the surrounding strings, enumstr gets confused and thinks the shift is another value

-  DtAppPaintScheme
 -  DtAppMobilityKill
 -  DtAppFirePowerKill
 -  DtAppDamage
 -  DtAppSmoke
 -  DtAppTrailingEffects
 -  DtAppHatch
 -  DtAppLights
 -  DtAppFlaming
 -  DtAppFrozen
 -  DtAppPowerPlantStatus
 -  DtAppFinalPdu
 -  DtAppDummy
-

Documentation

if you put spaces between the "<<" and the surrounding strings, enumstr gets confused and thinks the shift is another value

-  DtAppPaintScheme
-  DtAppMobilityKill
-  DtAppFirePowerKill
-  DtAppDamage
-  DtAppSmoke
-  DtAppTrailingEffects
-  DtAppHatch
-  DtAppLights

- DtAppFlaming
- DtAppFrozen
- DtAppPowerPlantStatus
- DtAppFinalPdu
- DtAppDummy

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtAppearAttrVal

Create an enumeration type, but since many of the names will have the same value, define them all as consts below

 DtAppearInvalid

Documentation

Create an enumeration type, but since many of the names will have the same value, define them all as consts below.

 DtAppearInvalid

In file minestatedpu.h:

class DtMinefieldStatePdu: public DtPdu

Inheritance:

DtMinefieldStatePdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtMinefieldStatePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtMinefieldStatePdu (const DtNetMinefieldStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtMinefieldStatePdu (const DtMinefieldStatePdu & orig)
 virtual	~DtMinefieldStatePdu () <i>destructor</i>
 DtMinefieldStatePdu &	operator= (const DtMinefieldStatePdu & orig) <i>Assignment operator</i>
 virtual void	setMinefieldID (const DtEntityIdentifier & id) <i>Set/Get minefield ID</i>
 virtual const DtEntityIdentifier &	minefieldID () const
 virtual void	setSequenceNumber (DtU16 field) <i>Set/Get minefield sequence number</i>
 virtual DtU16	sequenceNumber () const
 virtual void	setForceID (DtForceType force) <i>Set/Get the force ID</i>
 virtual DtForceType	forceID () const
 virtual DtBoolean	setNumPerimeterPoints (DtU8 num) <i>Set/Get the number of perimeter points</i>
 virtual DtU8	numPerimeterPoints () const
 virtual void	setMinefieldType (const DtEntityType & type) <i>Set/Get minefield type</i>
 virtual const DtEntityType &	minefieldType () const
 virtual DtBoolean	setNumMineTypes (DtU16 num) <i>Set/Get the number of mine types</i>
 virtual DtU16	numMineTypes () const
 virtual void	setLocation (DtConstVector coords) <i>Set/Get location of minefield</i>

virtual DtConstVector	location () const
virtual void	setOrientation (const DtTaitBryan & orientation) <i>Set/Get orientation</i>
virtual const DtTaitBryan &	orientation () const
virtual void	setAppearance (DtU16 app) <i>Set/Get appearance (as a 16 bit integer)</i>
virtual DtU16	appearance () const
virtual void	setAppearanceAttribute (DtAppearAttribute attr, DtAppearAttrVal val) <i>Set/Get appearance by attribute</i>
virtual DtAppearAttrVal	appearanceAttribute (DtAppearAttribute attr) const
virtual void	setMinefieldProtocolMode (DtMinefieldProtocol mode) <i>Set/Get which network protocol mode is being used in the minefield transmission</i>
virtual DtMinefieldProtocol	minefieldProtocolMode () const
virtual DtBoolean	setPerimeterPoint (int index, const DtMinefieldPerimeterRecord & init) <i>Set/Get a perimeter point coordinate record, returns DtFalse if the index is out of range</i>
virtual DtBoolean	getPerimeterPoint (int index, DtMinefieldPerimeterRecord & record) const
virtual DtBoolean	setMineType (int index, const DtEntityType & init) <i>Set/Get a mine type record</i>
virtual DtBoolean	getMineType (int index, DtEntityType & record) const
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtMinefieldStatePdu</i>
static void	addCallback (DtExerciseConn * conn, DtMinefieldStatePduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtMinefieldStatePduCb cb, void* usr)

Protected Fields

static int	theStaticSize <i>Size of PDU with zero parameter records</i>
DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>
DtEntityType	myEntityType
DtTaitBryan	myEulerAngles
DtVector	myVector
DtMinefieldPerimeterRecord	myPerimeterRecord

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
-----------------------------------	--

virtual DtNetMinefieldStatePdu*	netPduStruct () const <i>Casts the netPdu element to our network struct</i>
virtual DtMinefieldStatePdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type</i>
virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>
virtual int	perimeterPointRecordOffset (int recordIndex) const <i>Returns a buffer offset to a perimeter point record</i>
virtual int	mineTypeRecordOffset (int recordIndex) const <i>Returns a buffer offset to a mine type record</i>

Inherited from [DtPdu](#):

Public Methods

virtual void*	packet () const
virtual int	status () const
virtual void	setVersion (int vers)
virtual int	protocolVersion () const
virtual void	setExerciseld (int id)
virtual int	exerciseld () const
virtual DtPduKind	kind () const
virtual DtProtocolFamily	protocolFamily () const
virtual void	setTimeStamp (DtTime time, DtTimeStampType type = DtTimeStampRelative)
virtual DtTime	timeStamp () const
virtual DtTimeStampType	timeStampType () const
virtual DtTime	guessTimeValid (DtTime referenceTime) const
virtual void	print () const
virtual void	printHeader () const
inline int	exerciseld () const
inline DtPduKind	kind () const
inline DtNetPduHeader*	netHeader () const
inline void*	packet () const
inline DtProtocolFamily	protocolFamily () const
inline int	protocolVersion () const

- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**

int myMsgSize

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtMinefieldStatePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtMinefieldStatePdu(const DtNetMinefieldStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtMinefieldStatePdu(const [DtMinefieldStatePdu](#)& orig)
- virtual ~DtMinefieldStatePdu()
destructor
- [DtMinefieldStatePdu](#)& operator=(const [DtMinefieldStatePdu](#)& orig)
Assignment operator
- virtual void setMinefieldID(const [DtEntityIdentifier](#)& id)
Set/Get minefield ID
- virtual const [DtEntityIdentifier](#)& minefieldID() const
- virtual void setSequenceNumber(DtU16 field)
Set/Get minefield sequence number
- virtual DtU16 sequenceNumber() const
- virtual void setForceID([DtForceType](#) force)
Set/Get the force ID
- virtual [DtForceType](#) forceID() const

- virtual [DtBoolean](#) setNumPerimeterPoints(DtU8 num)
Set/Get the number of perimeter points
- virtual DtU8 numPerimeterPoints() const
- virtual void setMinefieldType(const [DtEntityType](#)& type)
Set/Get minefield type
- virtual const [DtEntityType](#)& minefieldType() const
- virtual [DtBoolean](#) setNumMineTypes(DtU16 num)
Set/Get the number of mine types
- virtual DtU16 numMineTypes() const
- virtual void setLocation(DtConstVector coords)
Set/Get location of minefield
- virtual DtConstVector location() const
- virtual void setOrientation(const [DtTaitBryan](#)& orientation)
Set/Get orientation
- virtual const [DtTaitBryan](#)& orientation() const
- virtual void setAppearance(DtU16 app)
Set/Get appearance (as a 16 bit integer)
- virtual DtU16 appearance() const
- virtual void setAppearanceAttribute([DtAppearAttribute](#) attr, [DtAppearAttrVal](#) val)
Set/Get appearance by attribute
- virtual [DtAppearAttrVal](#) appearanceAttribute([DtAppearAttribute](#) attr) const
- virtual void setMinefieldProtocolMode([DtMinefieldProtocol](#) mode)
Set/Get which network protocol mode is being used in the minefield transmission
- virtual [DtMinefieldProtocol](#) minefieldProtocolMode() const
- virtual [DtBoolean](#) setPerimeterPoint(int index, const [DtMinefieldPerimeterRecord](#)& init)
Set/Get a perimeter point coordinate record, returns DtFalse if the index is out of range
- virtual [DtBoolean](#) getPerimeterPoint(int index, [DtMinefieldPerimeterRecord](#)& record) const
- virtual [DtBoolean](#) setMineType(int index, const [DtEntityType](#)& init)
Set/Get a mine type record. Returns DtFalse if the index is out of range.
- virtual [DtBoolean](#) getMineType(int index, [DtEntityType](#)& record) const
- virtual void printData() const
Print the data contained by this PDU
- static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer =

DtUSE_INTERNAL_BUFFER)

Create a DtMinefieldStatePdu. Caller is responsible for deletion.

static void addCallback([DtExerciseConn](#)* conn, DtMinefieldStatePduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

static void removeCallback([DtExerciseConn](#)* conn, DtMinefieldStatePduCb cb, void* usr)

virtual [DtPduKind](#) internalGetPduKind() const

Returns the internal PDU kind

virtual DtNetMinefieldStatePdu* netPduStruct() const

Casts the netPdu element to our network struct

[DtMinefieldStatePdu](#)* self() const

Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type

virtual char* var() const

Returns a pointer to the first byte of variable length data

virtual int perimeterPointRecordOffset(int recordIndex) const

Returns a buffer offset to a perimeter point record

virtual int mineTypeRecordOffset(int recordIndex) const

Returns a buffer offset to a mine type record

static int theStaticSize

Size of PDU with zero parameter records

[DtEntityIdentifier](#) myEntityID

Temporary objects for reference returns

[DtEntityType](#) myEntityType

[DtTaitBryan](#) myEulerAngles

[DtVector](#) myVector

[DtMinefieldPerimeterRecord](#) myPerimeterRecord

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtMinefieldProtocol

 DtHeartbeatMode

 DtQRP_Mode

Documentation

 DtHeartbeatMode

 DtQRP_Mode

In file mineperimeter.h:

class DtMinefieldPerimeterRecord

class DtMinefieldPerimeterRecord: DtMinefieldPerimeterRecord is a class which contains coordinates for the minefield perimeter points

Public Methods

	DtMinefieldPerimeterRecord () <i>Constructors</i>
	DtMinefieldPerimeterRecord (const DtMinefieldPerimeterRecord & orig)
 DtMinefieldPerimeterRecord &	operator= (const DtMinefieldPerimeterRecord & orig) <i>Assignment operator</i>
 DtMinefieldPerimeterRecord &	operator= (const DtNetMinefieldPerimeterRecord& orig)
 virtual	~DtMinefieldPerimeterRecord () <i>Destructor</i>
 virtual void	setX (DtFloat32 value) <i>Set/Get the X Component</i>
 virtual DtFloat32	x () const
 virtual void	setY (DtFloat32 value) <i>Set/Get the Y Component</i>
 virtual DtFloat32	y () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetMinefieldPerimeterRecord& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetMinefieldPerimeterRecord& record) const

Protected Fields

 DtFloat32	myX <i>Data members:</i>
 DtFloat32	myY

Documentation

class DtMinefieldPerimeterRecord:

DtMinefieldPerimeterRecord is a class which contains coordinates for the minefield perimeter points. This record is used in the minefield state and minefield query PDUs.

- `DtMinefieldPerimeterRecord()`
Constructors
- `DtMinefieldPerimeterRecord(const DtMinefieldPerimeterRecord& orig)`
- `DtMinefieldPerimeterRecord& operator=( const DtMinefieldPerimeterRecord& orig)`
Assignment operator
- `DtMinefieldPerimeterRecord& operator=( const DtNetMinefieldPerimeterRecord& orig)`
- `virtual ~DtMinefieldPerimeterRecord()`
Destructor
- `virtual void setX(DtFloat32 value)`
Set/Get the X Component
- `virtual DtFloat32 x() const`
- `virtual void setY(DtFloat32 value)`
Set/Get the Y Component
- `virtual DtFloat32 y() const`
- `virtual void print() const`
Print the information contained in the record
- `virtual void unloadNetRecord( const DtNetMinefieldPerimeterRecord& record)`
Load/Unload data from a packet record
- `virtual void loadNetRecord( DtNetMinefieldPerimeterRecord& record) const`
- `DtFloat32 myX`
Data members:
- `DtFloat32 myY`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file minersnackpdu.h:

class DtMinefieldResponseNackPdu: public DtPdu

Inheritance:

DtMinefieldResponseNackPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtMinefieldResponseNackPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtMinefieldResponseNackPdu (const DtNetMinefieldResponseNackPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtMinefieldResponseNackPdu (const DtMinefieldResponseNackPdu & orig)
virtual	~DtMinefieldResponseNackPdu () <i>destructor</i>
DtMinefieldResponseNackPdu &	operator= (const DtMinefieldResponseNackPdu & orig) <i>Assignment operator</i>
virtual void	setMinefieldID (const DtEntityIdentifier & id) <i>Set/Get minefield ID</i>
virtual const DtEntityIdentifier &	minefieldID () const
virtual void	setRequestingEntityID (const DtEntityIdentifier & id) <i>Set/Get requesting entity ID</i>
virtual const DtEntityIdentifier &	requestingEntityID () const
virtual void	setRequestID (DtU8 id) <i>Set/Get request ID</i>
virtual DtU8	requestID () const
virtual DtBoolean	setNumMissingPDUs (DtU8 num) <i>Set/Get the number of missing PDUs</i>
virtual DtU8	numMissingPDUs () const
virtual DtBoolean	setPDU_SequenceNumber (int index, DtU8 num) <i>Set/Get a PDU sequence number</i>
virtual DtBoolean	getPDU_SequenceNumber (int index, DtU8& num) const
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtMinefieldResponseNackPdu</i>

static void

addCallback ([DtExerciseConn](#)* conn, DtMinefieldResponseNackPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

static void

removeCallback ([DtExerciseConn](#)* conn, DtMinefieldResponseNackPduCb cb, void* usr)

Protected Fields

static int

theStaticSize

Size of PDU with zero parameter records

[DtEntityIdentifier](#)

myEntityID

Temporary objects for reference returns

Protected Methods

virtual [DtPduKind](#)

internalGetPduKind () const

Returns the internal PDU kind

virtual DtNetMinefieldResponseNackPdu*

netPduStruct () const

Casts the netPdu element to our network struct

virtual char*

var () const

Returns a pointer to the first byte of variable length data

[DtMinefieldResponseNackPdu](#)*

self () const

Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type

Inherited from [DtPdu](#):

Public Methods

virtual void* **packet**() const

virtual int **status**() const

virtual void **setVersion**(int vers)

virtual int **protocolVersion**() const

virtual void **setExerciseId**(int id)

virtual int **exerciseId**() const

virtual [DtPduKind](#) **kind**() const

virtual [DtProtocolFamily](#) **protocolFamily**() const

virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

virtual DtTime **timeStamp**() const

virtual [DtTimeStampType](#) **timeStampType**() const

virtual DtTime **guessTimeValid**(DtTime referenceTime) const

- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtMinefieldResponseNackPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtMinefieldResponseNackPdu**(const DtNetMinefieldResponseNackPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtMinefieldResponseNackPdu**(const [DtMinefieldResponseNackPdu](#)& orig)
- virtual ~DtMinefieldResponseNackPdu**()
destructor

- `DtMinefieldResponseNackPdu& operator=( const DtMinefieldResponseNackPdu& orig)`
 Assignment operator
- `virtual void setMinefieldID(const DtEntityIdentifier& id)`
 Set/Get minefield ID
- `virtual const DtEntityIdentifier& minefieldID() const`
- `virtual void setRequestingEntityID(const DtEntityIdentifier& id)`
 Set/Get requesting entity ID
- `virtual const DtEntityIdentifier& requestingEntityID() const`
- `virtual void setRequestID(DtU8 id)`
 Set/Get request ID
- `virtual DtU8 requestID() const`
- `virtual DtBoolean setNumMissingPDUs(DtU8 num)`
 Set/Get the number of missing PDUs
- `virtual DtU8 numMissingPDUs() const`
- `virtual DtBoolean setPDU_SequenceNumber(int index, DtU8 num)`
 Set/Get a PDU sequence number. Returns DtFalse if the index is out of range.
- `virtual DtBoolean getPDU_SequenceNumber(int index, DtU8& num) const`
- `virtual void printData() const`
 Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
 Create a DtMinefieldResponseNackPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtMinefieldResponseNackPduCb cb, void* usr)`
 Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtMinefieldResponseNackPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
 Returns the internal PDU kind
- `virtual DtNetMinefieldResponseNackPdu* netPduStruct() const`
 Casts the netPdu element to our network struct
- `virtual char* var() const`
 Returns a pointer to the first byte of variable length data
- `DtMinefieldResponseNackPdu* self() const`
 Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type

- `static int theStaticSize`
Size of PDU with zero parameter records
 - `DtEntityIdentifier myEntityID`
Temporary objects for reference returns
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file minequerypdu.h:

class [DtMinefieldQueryPdu](#): public [DtPdu](#)

Inheritance:

DtMinefieldQueryPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtMinefieldQueryPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtMinefieldQueryPdu (const DtNetMinefieldQueryPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtMinefieldQueryPdu (const <u>DtMinefieldQueryPdu</u> & orig)
virtual	~DtMinefieldQueryPdu () <i>destructor</i>
<u>DtMinefieldQueryPdu</u> &	operator= (const <u>DtMinefieldQueryPdu</u> & orig) <i>Assignment operator</i>
virtual void	setMinefieldID (const <u>DtEntityIdentifier</u> & id) <i>Set/Get minefield ID</i>
virtual const <u>DtEntityIdentifier</u> &	minefieldID () const
virtual void	setRequestingEntityID (const <u>DtEntityIdentifier</u> & id) <i>Set/Get requesting entity ID</i>
virtual const <u>DtEntityIdentifier</u> &	requestingEntityID () const
virtual void	setRequestID (DtU8 id) <i>Set/Get request ID</i>
virtual DtU8	requestID () const
virtual <u>DtBoolean</u>	setNumPerimeterPoints (DtU8 num) <i>Set/Get the number of perimeter points</i>
virtual DtU8	numPerimeterPoints () const
virtual <u>DtBoolean</u>	setNumSensorTypes (DtU8 num) <i>Set/Get the number of sensor types</i>
virtual DtU8	numSensorTypes () const
virtual void	setDataFilter (DtU32 value) <i>Set/Get the data filter (as a 32-bit integer)</i>
virtual DtU32	dataFilter () const

virtual void	setDataFilterField (DtMinefieldDataFilterField field, DtBoolean value) <i>Set/Get data filter attributes (DtTrue indicates that the data for that field is requested, DtFalse indicates that the data for that field is not requested)</i>
virtual DtBoolean	getDataFilterField (DtMinefieldDataFilterField field)
virtual void	setRequestedMineType (const DtEntityType & type) <i>Set/Get requested mine type</i>
virtual const DtEntityType &	requestedMineType () const
virtual DtBoolean	setPerimeterPoint (int index, const DtMinefieldPerimeterRecord & init) <i>Set/Get a perimeter point coordinate record</i>
virtual DtBoolean	getPerimeterPoint (int index, DtMinefieldPerimeterRecord & record) const
virtual DtBoolean	getSensorTypeCategory (int index, DtSensorTypeCategory & value) <i>Get the sensor type category for a record</i>
virtual DtBoolean	setSensorTypeSubcategory (int index, DtOpticalSensorTypeSubcategory value) <i>Set a sensor type subcategory for a record</i>
virtual DtBoolean	setSensorTypeSubcategory (int index, DtFLIR_SensorTypeSubcategory value)
virtual DtBoolean	setSensorTypeSubcategory (int index, DtRADAR_SensorTypeSubcategory value)
virtual DtBoolean	setSensorTypeSubcategory (int index, DtMagneticSensorTypeSubcategory value)
virtual DtBoolean	setSensorTypeSubcategory (int index, DtLaserSensorTypeSubcategory value)
virtual DtBoolean	setSensorTypeSubcategory (int index, DtSONARSensorTypeSubcategory value)
virtual DtBoolean	setSensorTypeSubcategory (int index, DtPhysicalSensorTypeSubcategory value)
virtual DtBoolean	setSensorTypeSubcategory (int index, DtMultispectralSensorTypeSubcategory value)
virtual DtBoolean	getSensorTypeSubcategory (int index, DtOpticalSensorTypeSubcategory & value) <i>Get a sensor type subcategory for a record</i>
virtual DtBoolean	getSensorTypeSubcategory (int index, DtFLIR_SensorTypeSubcategory & value)
virtual DtBoolean	getSensorTypeSubcategory (int index, DtRADAR_SensorTypeSubcategory & value)
virtual DtBoolean	getSensorTypeSubcategory (int index, DtMagneticSensorTypeSubcategory & value)
virtual DtBoolean	getSensorTypeSubcategory (int index, DtLaserSensorTypeSubcategory & value)
virtual DtBoolean	getSensorTypeSubcategory (int index, DtSONARSensorTypeSubcategory & value)
virtual DtBoolean	getSensorTypeSubcategory (int index, DtPhysicalSensorTypeSubcategory & value)
virtual DtBoolean	getSensorTypeSubcategory (int index, DtMultispectralSensorTypeSubcategory & value)
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket * initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtMinefieldQueryPdu</i>
static void	addCallback (DtExerciseConn * conn, DtMinefieldQueryPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtMinefieldQueryPduCb cb, void* usr)

Protected Fields

 static int	theStaticSize <i>Size of PDU with zero parameter records</i>
 DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>
 DtEntityType	myEntityType
 DtMinefieldPerimeterRecord	myPerimeterRecord

Protected Methods

 virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
 virtual DtNetMinefieldQueryPdu*	netPduStruct () const <i>Casts the netPdu element to our network struct</i>
 DtMinefieldQueryPdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type</i>
 virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>
 virtual int	perimeterPointRecordOffset (int recordIndex) const <i>Returns a buffer offset to a perimeter point record</i>
 virtual int	sensorTypeRecordOffset (int recordIndex) const <i>Returns a buffer offset to a sensor type record</i>
 virtual void	setSensorType (int index, DtU16 value) <i>Set/Get the sensor type</i>
 virtual DtU16	sensorType (int index) const

Inherited from [DtPdu](#):

Public Methods

 virtual void* packet () const
 virtual int status () const
 virtual void setVersion (int vers)
 virtual int protocolVersion () const
 virtual void setExerciseld (int id)
 virtual int exerciseld () const
 virtual DtPduKind kind () const
 virtual DtProtocolFamily protocolFamily () const
 virtual void setTimeStamp (DtTime time, DtTimeStampType type = DtTimeStampRelative)

- virtual DtTime **timeStamp()** const
- virtual [DtTimeStampType](#) **timeStampType()** const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtMinefieldQueryPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtMinefieldQueryPdu**(const DtNetMinefieldQueryPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtMinefieldQueryPdu**(const [DtMinefieldQueryPdu](#)& orig)
- virtual **~DtMinefieldQueryPdu**()

destructor

[DtMinefieldQueryPdu](#)& operator=(const [DtMinefieldQueryPdu](#)& orig)

Assignment operator

virtual void setMinefieldID(const [DtEntityIdentifier](#)& id)

Set/Get minefield ID

virtual const [DtEntityIdentifier](#)& minefieldID() const

virtual void setRequestingEntityID(const [DtEntityIdentifier](#)& id)

Set/Get requesting entity ID

virtual const [DtEntityIdentifier](#)& requestingEntityID() const

virtual void setRequestID(DtU8 id)

Set/Get request ID

virtual DtU8 requestID() const

virtual [DtBoolean](#) setNumPerimeterPoints(DtU8 num)

Set/Get the number of perimeter points

virtual DtU8 numPerimeterPoints() const

virtual [DtBoolean](#) setNumSensorTypes(DtU8 num)

Set/Get the number of sensor types

virtual DtU8 numSensorTypes() const

virtual void setDataFilter(DtU32 value)

Set/Get the data filter (as a 32-bit integer)

virtual DtU32 dataFilter() const

virtual void setDataFilterField([DtMinefieldDataFilterField](#) field, [DtBoolean](#) value)

Set/Get data filter attributes (DtTrue indicates that the data for that field is requested, DtFalse indicates that the data for that field is not requested)

virtual [DtBoolean](#) getDataFilterField([DtMinefieldDataFilterField](#) field)

virtual void setRequestedMineType(const [DtEntityType](#)& type)

Set/Get requested mine type

virtual const [DtEntityType](#)& requestedMineType() const

virtual [DtBoolean](#) setPerimeterPoint(int index, const [DtMinefieldPerimeterRecord](#)& init)

Set/Get a perimeter point coordinate record. Returns DtFalse if the index is out of range.

virtual [DtBoolean](#) getPerimeterPoint(int index, [DtMinefieldPerimeterRecord](#)& record)

const

virtual [DtBoolean](#) getSensorTypeCategory(int index, [DtSensorTypeCategory](#)& value)

Get the sensor type category for a record. Returns DtFalse if the index is out of range. The category is set by setSensorTypeRecord.

virtual [DtBoolean](#) setSensorTypeSubcategory(int index, [DtOpticalSensorTypeSubcategory](#) value)

Set a sensor type subcategory for a record. The correct category is automatically set by this function according to type of the enumeration in the argument list. DtFalse is returned if the index is out of range.

virtual [DtBoolean](#) setSensorTypeSubcategory(int index, [DtFLIR_SensorTypeSubcategory](#) value)

virtual [DtBoolean](#) setSensorTypeSubcategory(int index, [DtRADAR_SensorTypeSubcategory](#) value)

virtual [DtBoolean](#) setSensorTypeSubcategory(int index, [DtMagneticSensorTypeSubcategory](#) value)

virtual [DtBoolean](#) setSensorTypeSubcategory(int index, [DtLaserSensorTypeSubcategory](#) value)

virtual [DtBoolean](#) setSensorTypeSubcategory(int index, [DtSONARSensorTypeSubcategory](#) value)

virtual [DtBoolean](#) setSensorTypeSubcategory(int index, [DtPhysicalSensorTypeSubcategory](#) value)

virtual [DtBoolean](#) setSensorTypeSubcategory(int index, [DtMultispectralSensorTypeSubcategory](#) value)

virtual [DtBoolean](#) getSensorTypeSubcategory(int index, [DtOpticalSensorTypeSubcategory](#)& value)

Get a sensor type subcategory for a record. DtFalse is returned if the index is out of range or if the category of the record does not match the category indicated by the type of the enumeration in the argument list.

virtual [DtBoolean](#) getSensorTypeSubcategory(int index, [DtFLIR_SensorTypeSubcategory](#)& value)

virtual [DtBoolean](#) getSensorTypeSubcategory(int index, [DtRADAR_SensorTypeSubcategory](#)& value)

virtual [DtBoolean](#) getSensorTypeSubcategory(int index, [DtMagneticSensorTypeSubcategory](#)& value)

virtual [DtBoolean](#) getSensorTypeSubcategory(int index, [DtLaserSensorTypeSubcategory](#)& value)

virtual [DtBoolean](#) getSensorTypeSubcategory(int index, [DtSONARSensorTypeSubcategory](#)& value)

virtual [DtBoolean](#) getSensorTypeSubcategory(int index, [DtPhysicalSensorTypeSubcategory](#)& value)

virtual [DtBoolean](#) getSensorTypeSubcategory(int index, [DtMultispectralSensorTypeSubcategory](#)& value)

virtual void printData() const

Print the data contained by this PDU

static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtMinefieldQueryPdu. Caller is responsible for deletion.

- `static void addCallback(DtExerciseConn* conn, DtMinefieldQueryPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtMinefieldQueryPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetMinefieldQueryPdu* netPduStruct() const`
Casts the netPdu element to our network struct
- `DtMinefieldQueryPdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
- `virtual char* var() const`
Returns a pointer to the first byte of variable length data
- `virtual int perimeterPointRecordOffset(int recordIndex) const`
Returns a buffer offset to a perimeter point record
- `virtual int sensorTypeRecordOffset(int recordIndex) const`
Returns a buffer offset to a sensor type record
- `virtual void setSensorType(int index, DtU16 value)`
Set/Get the sensor type. (As a 16-bit integer)
- `virtual DtU16 sensorType(int index) const`
- `static int theStaticSize`
Size of PDU with zero parameter records
- `DtEntityIdentifier myEntityID`
Temporary objects for reference returns
- `DtEntityType myEntityType`
- `DtMinefieldPerimeterRecord myPerimeterRecord`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtMinefieldDataFilterField

Minefield PDU enums

-  DtMineGroundBurialDepthOffset
 -  DtMineWaterBurialDepthOffset
 -  DtMineSnowBurialDepthOffset
 -  DtMineOrientation
 -  DtMineThermalContrast
 -  DtMineReflectance
 -  DtMineEmplacementAge
 -  DtMineTripDetonationWireLocation
 -  DtMineFusing
 -  DtMineScalarDetectionCoefficient
 -  DtMinePaintScheme
-

Documentation

Minefield PDU enums

-  DtMineGroundBurialDepthOffset
-  DtMineWaterBurialDepthOffset
-  DtMineSnowBurialDepthOffset
-  DtMineOrientation
-  DtMineThermalContrast
-  DtMineReflectance
-  DtMineEmplacementAge
-  DtMineTripDetonationWireLocation
-  DtMineFusing
-  DtMineScalarDetectionCoefficient
-  DtMinePaintScheme

In file disenums.h:

enum DtSensorTypeCategory

-  DtOtherSensorTypeCategory
 -  DtOpticalSensorType
 -  DtFLIR_SensorType
 -  DtRADAR_SensorType
 -  DtMagneticSensorType
 -  DtLaserSensorType
 -  DtSONAR_SensorType
 -  DtPhysicalSensorType
 -  DtMultispectralSensorType
-

Documentation

-  DtOtherSensorTypeCategory
-  DtOpticalSensorType
-  DtFLIR_SensorType
-  DtRADAR_SensorType
-  DtMagneticSensorType
-  DtLaserSensorType
-  DtSONAR_SensorType
-  DtPhysicalSensorType
-  DtMultispectralSensorType

In file disenums.h:

enum DtOpticalSensorTypeSubcategory

-  DtUnaidedEyeActivelySearching
 -  DtUnaidedEyeNotActivelySearching
 -  DtBinoculars
 -  DtImageIntensifier
 -  DtHMMWV_OccActivelySearching
 -  DtHMMWV_OccNotActivelySearching
 -  DtTruckOccActivelySearching
 -  DtTruckOccNotActivelySearching
 -  DtTrackedVehicleOccClosedHatchActSearching
 -  DtTrackedVehicleOccClosedHatchNotActSearching
 -  DtTrackedVehicleOccOpenHatchActSearching
 -  DtTrackedVehicleOccOpenHatchNotActSearching
-

Documentation

-  DtUnaidedEyeActivelySearching
-  DtUnaidedEyeNotActivelySearching
-  DtBinoculars
-  DtImageIntensifier
-  DtHMMWV_OccActivelySearching
-  DtHMMWV_OccNotActivelySearching
-  DtTruckOccActivelySearching
-  DtTruckOccNotActivelySearching
-  DtTrackedVehicleOccClosedHatchActSearching
-  DtTrackedVehicleOccClosedHatchNotActSearching
-  DtTrackedVehicleOccOpenHatchActSearching



DtTrackedVehicleOccOpenHatchNotActSearching

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtFLIR_SensorTypeSubcategory

-  DtGeneric3_5
 -  DtGeneric8_12
 -  DtASTAMIDS_I
 -  DtASTAMIDS_II
 -  DtGSTAMIDS3_5
 -  DtGSTAMIDS8_12
 -  DtHSTAMIDS3_5
 -  DtHSTAMIDS8_12
 -  DtCOBRA3_5
 -  DtCOBRA8_12
-

Documentation

-  DtGeneric3_5
-  DtGeneric8_12
-  DtASTAMIDS_I
-  DtASTAMIDS_II
-  DtGSTAMIDS3_5
-  DtGSTAMIDS8_12
-  DtHSTAMIDS3_5
-  DtHSTAMIDS8_12
-  DtCOBRA3_5
-  DtCOBRA8_12

In file disenums.h:

enum DtRADAR_SensorTypeSubcategory

-
-  DtRADAR_Generic
 -  DtGenericGPR
 -  DtGSTAMIDS_I
 -  DtGSTAMIDS_II
 -  DtHSTAMIDS_I
 -  DtHSTAMIDS_II
-

Documentation

-  DtRADAR_Generic
-  DtGenericGPR
-  DtGSTAMIDS_I
-  DtGSTAMIDS_II
-  DtHSTAMIDS_I
-  DtHSTAMIDS_II

In file disenums.h:

enum DtMagneticSensorTypeSubcategory

 DtMagneticGeneric

 DtAN_PSS_11

 DtAN_PSS_12

 DtGSTAMIDS

Documentation

 DtMagneticGeneric

 DtAN_PSS_11

 DtAN_PSS_12

 DtGSTAMIDS

In file disenums.h:

enum DtLaserSensorTypeSubcategory

 DtLaserGeneric

 DtASTAMIDS

Documentation

 DtLaserGeneric

 DtASTAMIDS

In file disenums.h:

enum DtSONARSensorTypeSubcategory

 DtSONAR_Generic

Documentation

 DtSONAR_Generic

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtPhysicalSensorTypeSubcategory

-
-  DtGenericProbe
 -  DtProbeMetalContent
 -  DtProbeNoMetalContent
-

Documentation

-  DtGenericProbe
-  DtProbeMetalContent
-  DtProbeNoMetalContent

In file disenums.h:

enum DtMultispectralSensorTypeSubcategory

 DtMultispectralGeneric

Documentation

 DtMultispectralGeneric

[Alphabetic index](#) [Hierarchy of classes](#)

In file minedatapdu.h:

class DtMinefieldDataPdu: public DtPdu

Inheritance:

DtMinefieldDataPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtMinefieldDataPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtMinefieldDataPdu (const DtNetMinefieldDataPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtMinefieldDataPdu (const DtMinefieldDataPdu & orig)
 virtual	~DtMinefieldDataPdu () <i>destructor</i>
 DtMinefieldDataPdu &	operator= (const DtMinefieldDataPdu & orig) <i>Assignment operator</i>
 virtual void	setMinefieldID (const DtEntityIdentifier & id) <i>Set/Get minefield ID</i>
 virtual const DtEntityIdentifier &	minefieldID () const
 virtual void	setRequestingEntityID (const DtEntityIdentifier & id) <i>Set/Get requesting entity ID</i>
 virtual const DtEntityIdentifier &	requestingEntityID () const
 virtual void	setMinefieldSequenceNumber (DtU16 field) <i>Set/Get minefield sequence number</i>
 virtual DtU16	minefieldSequenceNumber () const
 virtual void	setRequestID (DtU8 value) <i>Set/Get the request ID</i>
 virtual DtU8	requestID () const
 virtual void	setPDU_SequenceNumber (DtU8 value) <i>Set/Get the PDU sequence number</i>
 virtual DtU8	PDU_SequenceNumber () const
 virtual void	setNumberOfPdus (DtU8 value) <i>Set/Get the number of PDUs</i>
 virtual DtU8	numberOfPdus () const
 virtual DtBoolean	setNumberOfMines (DtU8 value) <i>Set/Get the number of mines (this call erases any data which was set in the variable length fields)</i>

virtual DtU8

virtual [DtBoolean](#)

virtual DtU8

virtual void

virtual DtU32

virtual void

virtual [DtBoolean](#)

virtual void

virtual const [DtEntityType&](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

numberOfMines () const

setNumberOfSensorTypes (DtU8 value)

Set/Get the number of sensor types (this call erases any data which was set in the variable length fields)

numberOfSensorTypes () const

setDataFilter (DtU32 value)

Set/Get the data filter as a 32-bit value (this call erases any data which was set in the variable length fields)

dataFilter () const

setDataFilterBit ([DtMinefieldDataFilterField](#) field, [DtBoolean](#) value)

Set/Get a data filter bit (this call erases any data which was set in the variable length fields)

dataFilterBit ([DtMinefieldDataFilterField](#) field) const

setMineType (const [DtEntityType&](#) value)

Set/Get the mine type record

mineType () const

setSensorTypeCategory (int sensorIndex, [DtSensorTypeCategory](#) value)

Set/Get a sensor type category

getSensorTypeCategory (int sensorIndex, [DtSensorTypeCategory&](#) value) const

setSensorTypeSubcategory (int sensorIndex, DtU16 value)

Set/Get a sensor type sub-category

getSensorTypeSubcategory (int sensorIndex, DtU16& value) const

setMineLocation (int mineIndex, DtConstVector value)

Set/Get a mine location

getMineLocation (int mineIndex, DtVectorRef value) const

setGroundBurialDepthOffset (int mineIndex, DtFloat32 value)

Set/Get a ground burial depth offset

getGroundBurialDepthOffset (int mineIndex, DtFloat32& value) const

setWaterBurialDepthOffset (int mineIndex, DtFloat32 value)

Set/Get a water burial depth offset

getWaterBurialDepthOffset (int mineIndex, DtFloat32& value) const

setSnowBurialDepthOffset (int mineIndex, DtFloat32 value)

Set/Get a snow burial depth offset

getSnowBurialDepthOffset (int mineIndex, DtFloat32& value) const

setMineOrientation (int mineIndex, const [DtTaitBryan&](#) value)

Set/Get mine orientation

getMineOrientation (int mineIndex, [DtTaitBryan&](#) value) const

setThermalContrast (int mineIndex, DtFloat32 value)

Set/Get a thermal contrast value

getThermalContrast (int mineIndex, DtFloat32& value) const

setReflectance (int mineIndex, DtFloat32 value)

Set/Get a reflectance value

getReflectance (int mineIndex, DtFloat32& value) const

setMineEmplacementTime (int mineIndex, DtTime value)

Set/Get a mine emplacement time

- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual void
- static [DtPdu](#)*
- static void
- static void

getMineEmplacementTime (int mineIndex, DtTime& value) const

setMineEntityID (int mineIndex, DtU16 value)
Set/Get a mine entity ID

getMineEntityID (int mineIndex, DtU16& value) const

setPrimaryFusing (int mineIndex, [DtMinefieldFusing](#) value)
Set/Get primary fusing

getPrimaryFusing (int mineIndex, [DtMinefieldFusing](#)& value) const

setSecondaryFusing (int mineIndex, [DtMinefieldFusing](#) value)
Set/Get secondary fusing

getSecondaryFusing (int mineIndex, [DtMinefieldFusing](#)& value) const

setAntiHandlingDevice (int mineIndex, [DtMinefieldAntiHandlingDevice](#) value)
Set/Get anti-handling device

getAntiHandlingDevice (int mineIndex, [DtMinefieldAntiHandlingDevice](#)& value) const

setScalarDetectionCoefficient (int sensorIndex, int mineIndex, DtU8 value)
Set/Get scalar detection coefficient

getScalarDetectionCoefficient (int sensorIndex, int mineIndex, DtU8& value) const

setMinefieldAlgae (int mineIndex, [DtMinefieldAlgae](#) value)
Set/Get minefield algae

getMinefieldAlgae (int mineIndex, [DtMinefieldAlgae](#)& value) const

setPaintScheme (int mineIndex, [DtMinefieldPaintScheme](#) value)
Set/Get paint scheme

getPaintScheme (int mineIndex, [DtMinefieldPaintScheme](#)& value) const

setNumTripWires (int mineIndex, DtU8 num)
Set/Get the number of trip/detonation wires

getNumTripWires (int mineIndex, DtU8& num) const

setNumVertices (int mineIndex, int wireIndex, DtU8 num)
Set/Get the number of vertices

getNumVertices (int mineIndex, int wireIndex, DtU8& num) const

setVertex (int mineIndex, int wireIndex, int vertexIndex, DtConstVector value)
Set/Get a vertex

getVertex (int mineIndex, int wireIndex, int vertexIndex, DtVectorRef value) const

printData () const
Print the data contained by this PDU

create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Create a DtMinefieldDataPdu

addCallback ([DtExerciseConn](#)* conn, DtMinefieldDataPduCb cb, void* usr)
Add (register)/Remove (deregister) function to be called when pdu is received

removeCallback ([DtExerciseConn](#)* conn, DtMinefieldDataPduCb cb, void* usr)

Protected Fields

- static int **theStaticSize**
Size of PDU with zero parameter records

 [DtEntityIdentifier](#) **myEntityID**
Temporary objects for reference returns

 [DtEntityType](#) **myEntityType**

Protected Methods

 virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

 virtual DtNetMinefieldDataPdu* **netPduStruct** () const
Casts the netPdu element to our network struct

 [DtMinefieldDataPdu](#)* **self** () const
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type

 virtual char* **var** () const
Returns a pointer to the first byte of variable length data

 virtual int **sensorTypeRecordOffset** (int sensorIndex) const
Returns a buffer offset to the sensor type records

 virtual int **sensorTypePaddingOffset** () const
Returns a buffer offset to the padding following sensor type information

 virtual int **sensorTypePaddingBytes** () const
Returns the number of padding bytes following sensor type information

 virtual int **mineLocationRecordOffset** (int mineIndex) const
Returns a buffer offset to the mine location records

 virtual int **groundBurialDepthRecordOffset** (int mineIndex) const
Returns a buffer offset to the ground burial depth records

 virtual int **waterBurialDepthRecordOffset** (int mineIndex) const
Returns a buffer offset to the water burial depth records

 virtual int **snowBurialDepthRecordOffset** (int mineIndex) const
Returns a buffer offset to the snow burial depth records

 virtual int **mineOrientationRecordOffset** (int mineIndex) const
Returns a buffer offset to the mine orientation records

 virtual int **thermalContrastRecordOffset** (int mineIndex) const
Returns a buffer offset to the thermal contrast records

 virtual int **reflectanceRecordOffset** (int mineIndex) const
Returns a buffer offset to the reflectance records

 virtual int **mineEmplacementTimeRecordOffset** (int mineIndex) const
Returns a buffer offset to the mine emplacement time records

 virtual int **mineEntityID_RecordOffset** (int mineIndex) const
Returns a buffer offset to the mine entity ID records

 virtual int **fusingRecordOffset** (int mineIndex) const
Returns a buffer offset to the fusing records

 virtual int **scalarDetectionCoefficientRecordOffset** (int sensorIndex, int mineIndex) const
Returns a buffer offset to the scalar detection coefficient records

 virtual int **paintSchemeRecordOffset** (int mineIndex) const
Returns a buffer offset to the paint scheme records

 virtual int **paintSchemePaddingOffset** () const
Returns a buffer offset to the padding following paint scheme records

 virtual int **paintSchemePaddingBytes** () const
Returns the number of padding bytes following paint scheme records

 virtual int **numberOfTripWiresRecordOffset** (int mineIndex) const
Returns a buffer offset to the number of wires records

virtual int	numberOfTripWiresPaddingOffset () const <i>Returns a buffer offset to the padding following number of wires records</i>
virtual int	numberOfTripWiresPaddingBytes () const <i>Returns the number of padding bytes following number of wires records</i>
virtual int	variableDataLengthAfterReset () const <i>Returns the length of the variable data section in the PDU after the variable data has been reset</i>
virtual int	numberOfVerticesRecordOffset (int mineIndex, int wireIndex) const <i>Returns a buffer offset to the number of vertices records</i>
virtual int	numberOfVerticesPaddingOffset () const <i>Returns a buffer offset to the padding following number of vertices records</i>
virtual int	numberOfVerticesPaddingBytes () const <i>Returns the number of padding bytes following number of vertices records</i>
virtual int	vertexRecordOffset (int mineIndex, int wireIndex, int vertexIndex) const <i>Returns a buffer offset to the vertex records</i>
virtual int	variableDataLength () const <i>Returns the length of the variable data section in the PDU</i>

Inherited from [DtPdu](#):

Public Methods

virtual void*	packet () const
virtual int	status () const
virtual void	setVersion (int vers)
virtual int	protocolVersion () const
virtual void	setExerciseld (int id)
virtual int	exerciseld () const
virtual DtPduKind	kind () const
virtual DtProtocolFamily	protocolFamily () const
virtual void	setTimeStamp (DtTime time, DtTimeStampType type = DtTimeStampRelative)
virtual DtTime	timeStamp () const
virtual DtTimeStampType	timeStampType () const
virtual DtTime	guessTimeValid (DtTime referenceTime) const
virtual void	print () const
virtual void	printHeader () const
inline int	exerciseld () const
inline DtPduKind	kind () const
inline DtNetPduHeader*	netHeader () const

- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtMinefieldDataPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtMinefieldDataPdu**(const DtNetMinefieldDataPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtMinefieldDataPdu**(const [DtMinefieldDataPdu](#)& orig)
- virtual ~DtMinefieldDataPdu**()
destructor
- [DtMinefieldDataPdu](#)& operator=**(const [DtMinefieldDataPdu](#)& orig)
Assignment operator
- virtual void setMinefieldID**(const [DtEntityIdentifier](#)& id)
Set/Get minefield ID
- virtual const [DtEntityIdentifier](#)& minefieldID**() const
- virtual void setRequestingEntityID**(const [DtEntityIdentifier](#)& id)
Set/Get requesting entity ID

- virtual const [DtEntityIdentifier](#)& requestingEntityID() const
- virtual void setMinefieldSequenceNumber(DtU16 field)
Set/Get minefield sequence number
- virtual DtU16 minefieldSequenceNumber() const
- virtual void setRequestID(DtU8 value)
Set/Get the request ID
- virtual DtU8 requestID() const
- virtual void setPDU_SequenceNumber(DtU8 value)
Set/Get the PDU sequence number
- virtual DtU8 PDU_SequenceNumber() const
- virtual void setNumberOfPdus(DtU8 value)
Set/Get the number of PDUs
- virtual DtU8 numberOfPdus() const
- virtual [DtBoolean](#) setNumberOfMines(DtU8 value)
Set/Get the number of mines (this call erases any data which was set in the variable length fields)
- virtual DtU8 numberOfMines() const
- virtual [DtBoolean](#) setNumberOfSensorTypes(DtU8 value)
Set/Get the number of sensor types (this call erases any data which was set in the variable length fields)
- virtual DtU8 numberOfSensorTypes() const
- virtual void setDataFilter(DtU32 value)
Set/Get the data filter as a 32-bit value (this call erases any data which was set in the variable length fields)
- virtual DtU32 dataFilter() const
- virtual void setDataFilterBit([DtMinefieldDataFilterField](#) field, [DtBoolean](#) value)
Set/Get a data filter bit (this call erases any data which was set in the variable length fields)
- virtual [DtBoolean](#) dataFilterBit([DtMinefieldDataFilterField](#) field) const
- virtual void setMineType(const [DtEntityType](#)& value)
Set/Get the mine type record
- virtual const [DtEntityType](#)& mineType() const
- virtual [DtBoolean](#) setSensorTypeCategory(int sensorIndex, [DtSensorTypeCategory](#) value)
Set/Get a sensor type category. Returns DtFalse if the index is out of range.
- virtual [DtBoolean](#) getSensorTypeCategory(int sensorIndex, [DtSensorTypeCategory](#)& value) const
- virtual [DtBoolean](#) setSensorTypeSubcategory(int sensorIndex, DtU16 value)
Set/Get a sensor type sub-category. Returns DtFalse if the index is out of range.

- virtual [DtBoolean](#) getSensorTypeSubcategory(int sensorIndex, DtU16& value) const
- virtual [DtBoolean](#) setMineLocation(int mineIndex, DtConstVector value)
Set/Get a mine location. Returns DtFalse if the index is out of range.
- virtual [DtBoolean](#) getMineLocation(int mineIndex, DtVectorRef value) const
- virtual [DtBoolean](#) setGroundBurialDepthOffset(int mineIndex, DtFloat32 value)
Set/Get a ground burial depth offset. Returns DtFalse if the index is out of range or if an appropriate flag bit is not set.
- virtual [DtBoolean](#) getGroundBurialDepthOffset(int mineIndex, DtFloat32& value) const
- virtual [DtBoolean](#) setWaterBurialDepthOffset(int mineIndex, DtFloat32 value)
Set/Get a water burial depth offset. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getWaterBurialDepthOffset(int mineIndex, DtFloat32& value) const
- virtual [DtBoolean](#) setSnowBurialDepthOffset(int mineIndex, DtFloat32 value)
Set/Get a snow burial depth offset. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getSnowBurialDepthOffset(int mineIndex, DtFloat32& value) const
- virtual [DtBoolean](#) setMineOrientation(int mineIndex, const [DtTaitBryan](#)& value)
Set/Get mine orientation. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getMineOrientation(int mineIndex, [DtTaitBryan](#)& value) const
- virtual [DtBoolean](#) setThermalContrast(int mineIndex, DtFloat32 value)
Set/Get a thermal contrast value. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getThermalContrast(int mineIndex, DtFloat32& value) const
- virtual [DtBoolean](#) setReflectance(int mineIndex, DtFloat32 value)
Set/Get a reflectance value. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getReflectance(int mineIndex, DtFloat32& value) const
- virtual [DtBoolean](#) setMineEmplacementTime(int mineIndex, DtTime value)
Set/Get a mine emplacement time. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getMineEmplacementTime(int mineIndex, DtTime& value) const
- virtual [DtBoolean](#) setMineEntityID(int mineIndex, DtU16 value)
Set/Get a mine entity ID. Returns DtFalse if the index is out of range.
- virtual [DtBoolean](#) getMineEntityID(int mineIndex, DtU16& value) const
- virtual [DtBoolean](#) setPrimaryFusing(int mineIndex, [DtMinefieldFusing](#) value)
Set/Get primary fusing. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.

- virtual [DtBoolean](#) getPrimaryFusing(int mineIndex, [DtMinefieldFusing&](#) value) const
- virtual [DtBoolean](#) setSecondaryFusing(int mineIndex, [DtMinefieldFusing](#) value)
Set/Get secondary fusing. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getSecondaryFusing(int mineIndex, [DtMinefieldFusing&](#) value) const
- virtual [DtBoolean](#) setAntiHandlingDevice(int mineIndex, [DtMinefieldAntiHandlingDevice](#) value)
Set/Get anti-handling device. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getAntiHandlingDevice(int mineIndex, [DtMinefieldAntiHandlingDevice&](#) value) const
- virtual [DtBoolean](#) setScalarDetectionCoefficient(int sensorIndex, int mineIndex, DtU8 value)
Set/Get scalar detection coefficient. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getScalarDetectionCoefficient(int sensorIndex, int mineIndex, DtU8& value) const
- virtual [DtBoolean](#) setMinefieldAlgae(int mineIndex, [DtMinefieldAlgae](#) value)
Set/Get minefield algae. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getMinefieldAlgae(int mineIndex, [DtMinefieldAlgae&](#) value) const
- virtual [DtBoolean](#) setPaintScheme(int mineIndex, [DtMinefieldPaintScheme](#) value)
Set/Get paint scheme. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getPaintScheme(int mineIndex, [DtMinefieldPaintScheme&](#) value) const
- virtual [DtBoolean](#) setNumTripWires(int mineIndex, DtU8 num)
Set/Get the number of trip/detonation wires. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getNumTripWires(int mineIndex, DtU8& num) const
- virtual [DtBoolean](#) setNumVertices(int mineIndex, int wireIndex, DtU8 num)
Set/Get the number of vertices. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getNumVertices(int mineIndex, int wireIndex, DtU8& num) const
- virtual [DtBoolean](#) setVertex(int mineIndex, int wireIndex, int vertexIndex, DtConstVector value)
Set/Get a vertex. Returns DtFalse if the index is out of range or if the appropriate flag bit is not set.
- virtual [DtBoolean](#) getVertex(int mineIndex, int wireIndex, int vertexIndex, DtVectorRef value) const
- virtual void printData() const
Print the data contained by this PDU
- static [DtPdu*](#) create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtMinefieldDataPdu. Caller is responsible for deletion.

- `static void addCallback(DtExerciseConn* conn, DtMinefieldDataPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtMinefieldDataPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetMinefieldDataPdu* netPduStruct() const`
Casts the netPdu element to our network struct
- `DtMinefieldDataPdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
- `virtual char* var() const`
Returns a pointer to the first byte of variable length data
- `virtual int sensorTypeRecordOffset(int sensorIndex) const`
Returns a buffer offset to the sensor type records
- `virtual int sensorTypePaddingOffset() const`
Returns a buffer offset to the padding following sensor type information
- `virtual int sensorTypePaddingBytes() const`
Returns the number of padding bytes following sensor type information
- `virtual int mineLocationRecordOffset(int mineIndex) const`
Returns a buffer offset to the mine location records
- `virtual int groundBurialDepthRecordOffset(int mineIndex) const`
Returns a buffer offset to the ground burial depth records
- `virtual int waterBurialDepthRecordOffset(int mineIndex) const`
Returns a buffer offset to the water burial depth records
- `virtual int snowBurialDepthRecordOffset(int mineIndex) const`
Returns a buffer offset to the snow burial depth records
- `virtual int mineOrientationRecordOffset(int mineIndex) const`
Returns a buffer offset to the mine orientation records
- `virtual int thermalContrastRecordOffset(int mineIndex) const`
Returns a buffer offset to the thermal contrast records
- `virtual int reflectanceRecordOffset(int mineIndex) const`
Returns a buffer offset to the reflectance records
- `virtual int mineEmplacementTimeRecordOffset(int mineIndex) const`
Returns a buffer offset to the mine emplacement time records
- `virtual int mineEntityID_RecordOffset(int mineIndex) const`
Returns a buffer offset to the mine entity ID records

- 
virtual int fusingRecordOffset(int mineIndex) const
 Returns a buffer offset to the fusing records
- 
virtual int scalarDetectionCoefficientRecordOffset(int sensorIndex, int mineIndex) const
 Returns a buffer offset to the scalar detection coefficient records
- 
virtual int paintSchemeRecordOffset(int mineIndex) const
 Returns a buffer offset to the paint scheme records
- 
virtual int paintSchemePaddingOffset() const
 Returns a buffer offset to the padding following paint scheme records
- 
virtual int paintSchemePaddingBytes() const
 Returns the number of padding bytes following paint scheme records
- 
virtual int numberOfTripWiresRecordOffset(int mineIndex) const
 Returns a buffer offset to the number of wires records
- 
virtual int numberOfTripWiresPaddingOffset() const
 Returns a buffer offset to the padding following number of wires records
- 
virtual int numberOfTripWiresPaddingBytes() const
 Returns the number of padding bytes following number of wires records
- 
virtual int variableDataLengthAfterReset() const
 Returns the length of the variable data section in the PDU after the variable data has been reset. For each mine, the number of trip wires has been set to zero so there is no variable length data following.
- 
virtual int numberOfVerticesRecordOffset(int mineIndex, int wireIndex) const
 Returns a buffer offset to the number of vertices records
- 
virtual int numberOfVerticesPaddingOffset() const
 Returns a buffer offset to the padding following number of vertices records
- 
virtual int numberOfVerticesPaddingBytes() const
 Returns the number of padding bytes following number of vertices records
- 
virtual int vertexRecordOffset(int mineIndex, int wireIndex, int vertexIndex) const
 Returns a buffer offset to the vertex records
- 
virtual int variableDataLength() const
 Returns the length of the variable data section in the PDU
- 
static int theStaticSize
 Size of PDU with zero parameter records
- 
[DtEntityIdentifier](#) myEntityID
 Temporary objects for reference returns
- 
[DtEntityType](#) myEntityType

This class has no child classes.

In file disenums.h:

enum DtMinefieldFusing

-
-  DtNoFuse
 -  DtOtherFuse
 -  DtPressureFuse
 -  DtMagneticFuse
 -  DtTiltRodFuse
 -  DtCommandFuse
 -  DtTripWireFuse
-

Documentation

-  DtNoFuse
-  DtOtherFuse
-  DtPressureFuse
-  DtMagneticFuse
-  DtTiltRodFuse
-  DtCommandFuse
-  DtTripWireFuse

In file disenums.h:

enum DtMinefieldAntiHandlingDevice

 DtNoAntiHandlingDevice

 DtAntiHandlingDevice

Documentation

 DtNoAntiHandlingDevice

 DtAntiHandlingDevice

In file disenums.h:

enum DtMinefieldAlgae

-  DtNoAlgae
 -  DtLightAlgae
 -  DtModerateAlgae
 -  DtHeavyAlgae
-

Documentation

-  DtNoAlgae
-  DtLightAlgae
-  DtModerateAlgae
-  DtHeavyAlgae

In file disenums.h:

enum DtMinefieldPaintScheme

-  DtOtherPaintScheme
 -  DtStandardPaintScheme
 -  DtCamouflageDesertPaintScheme
 -  DtCamouflageJunglePaintScheme
 -  DtCamouflageSnowPaintScheme
 -  DtCamouflageGravelPaintScheme
 -  DtCamouflagePavementPaintScheme
 -  DtCamouflageSandPaintScheme
 -  DtNaturalWoodPaintScheme
 -  DtClearPaintScheme
 -  DtRedPaintScheme
 -  DtBluePaintScheme
 -  DtGreenPaintScheme
 -  DtOlivePaintScheme
 -  DtWhitePaintScheme
 -  DtTanPaintScheme
 -  DtBlackPaintScheme
 -  DtYellowPaintScheme
 -  DtBrownPaintScheme
-

Documentation

-  DtOtherPaintScheme
-  DtStandardPaintScheme
-  DtCamouflageDesertPaintScheme
-  DtCamouflageJunglePaintScheme

- DtCamouflageSnowPaintScheme
- DtCamouflageGravelPaintScheme
- DtCamouflagePavementPaintScheme
- DtCamouflageSandPaintScheme
- DtNaturalWoodPaintScheme
- DtClearPaintScheme
- DtRedPaintScheme
- DtBluePaintScheme
- DtGreenPaintScheme
- DtOlivePaintScheme
- DtWhitePaintScheme
- DtTanPaintScheme
- DtBlackPaintScheme
- DtYellowPaintScheme
- DtBrownPaintScheme

[Alphabetic index](#) [Hierarchy of classes](#)

In file linobjstpdu.h:

class DtLinearObjectStatePdu: public DtPdu

Inheritance:

DtLinearObjectStatePdu - DtPdu - DtBaseMsg

Public Methods

	DtLinearObjectStatePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtLinearObjectStatePdu (const DtNetLinearObjectStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtLinearObjectStatePdu (const DtLinearObjectStatePdu & orig)
 virtual	~DtLinearObjectStatePdu () <i>destructor</i>
 DtLinearObjectStatePdu &	operator= (const DtLinearObjectStatePdu & orig) <i>Assignment operator</i>
 virtual void	setObjectID (const DtEntityIdentifier & id) <i>Set/Get object ID</i>
 virtual const DtEntityIdentifier &	objectID () const
 virtual void	setReferencedObjectID (const DtEntityIdentifier & id) <i>Set/Get referenced object ID</i>
 virtual const DtEntityIdentifier &	referencedObjectID () const
 virtual void	setUpdateNumber (DtU16 field) <i>Set/Get update number</i>
 virtual DtU16	updateNumber () const
 virtual void	setForceID (DtForceType force) <i>Set/Get the force ID</i>
 virtual DtForceType	forceID () const
 virtual DtBoolean	setNumSegments (DtU8 num) <i>Set/Get the number of segments</i>
 virtual DtU8	numSegments () const
 virtual void	setRequesterID (const DtSimulationAddress& type) <i>Set/Get the requester ID</i>
 virtual const DtSimulationAddress&	requesterID () const
 virtual void	setReceivingID (const DtSimulationAddress& type) <i>Set/Get the receiving ID</i>

virtual const DtSimulationAddress&	receivingID () const
virtual void	setObjectType (const DtObjectType & type) <i>Set/Get the object type</i>
virtual const DtObjectType &	objectType () const
virtual DtBoolean	setLinearSegmentParameter (int index, const DtLinearSegmentParameter & init) <i>Set/Get a linear segment parameter record</i>
virtual DtBoolean	getLinearSegmentParameter (int index, DtLinearSegmentParameter & record) const
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtLinearObjectStatePdu</i>
static void	addCallback (DtExerciseConn * conn, DtLinearObjectStatePduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtLinearObjectStatePduCb cb, void* usr)

Protected Fields

static int	theStaticSize <i>Size of PDU with zero parameter records</i>
DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>
DtObjectType	myObjectType
DtSimulationAddress	mySimulationAddress

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
virtual DtNetLinearObjectStatePdu*	netPduStruct () const <i>Casts the netPdu element to our network struct</i>
DtLinearObjectStatePdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type</i>
virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>

Inherited from [DtPdu](#):

Public Methods

virtual void*	packet () const
virtual int	status () const

- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)
- virtual void **insertBytes**(void *address, int numBytes)
- virtual void **deleteBytes**(int offset, int numBytes)
- virtual void **deleteBytes**(void *address, int numBytes)

- virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtLinearObjectStatePdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Constructors

- DtLinearObjectStatePdu**(const DtNetLinearObjectStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

- DtLinearObjectStatePdu**(const [DtLinearObjectStatePdu](#)& orig)

- virtual **~DtLinearObjectStatePdu**()

destructor

- [DtLinearObjectStatePdu](#)& operator=(const [DtLinearObjectStatePdu](#)& orig)

Assignment operator

- virtual void **setObjectID**(const [DtEntityIdentifier](#)& id)

Set/Get object ID

- virtual const [DtEntityIdentifier](#)& **objectID**() const

- virtual void **setReferencedObjectID**(const [DtEntityIdentifier](#)& id)

Set/Get referenced object ID

- virtual const [DtEntityIdentifier](#)& **referencedObjectID**() const

- virtual void **setUpdateNumber**(DtU16 field)

Set/Get update number

- virtual DtU16 **updateNumber**() const

- virtual void **setForceID**([DtForceType](#) force)

Set/Get the force ID

- virtual [DtForceType](#) **forceID**() const

- virtual [DtBoolean](#) **setNumSegments**(DtU8 num)

Set/Get the number of segments

- virtual DtU8 **numSegments**() const

- virtual void **setRequesterID**(const DtSimulationAddress& type)

Set/Get the requester ID

- virtual const DtSimulationAddress& **requesterID**() const

- virtual void **setReceivingID**(const DtSimulationAddress& type)

Set/Get the receiving ID

- `virtual const DtSimulationAddress& receivingID() const`
- `virtual void setObjectType(const DtObjectType& type)`
Set/Get the object type
- `virtual const DtObjectType& objectType() const`
- `virtual DtBoolean setLinearSegmentParameter(int index, const DtLinearSegmentParameter& init)`
Set/Get a linear segment parameter record. Returns DtFalse if the index is out of range.
- `virtual DtBoolean getLinearSegmentParameter(int index, DtLinearSegmentParameter& record) const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtLinearObjectStatePdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtLinearObjectStatePduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtLinearObjectStatePduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetLinearObjectStatePdu* netPduStruct() const`
Casts the netPdu element to our network struct
- `DtLinearObjectStatePdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
- `virtual char* var() const`
Returns a pointer to the first byte of variable length data
- `static int theStaticSize`
Size of PDU with zero parameter records
- `DtEntityIdentifier myEntityID`
Temporary objects for reference returns
- `DtObjectType myObjectType`
- `DtSimulationAddress mySimulationAddress`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file linsegparam.h:

class DtLinearSegmentParameter

class DtLinearSegmentParameter: DtLinearSegmentParameter is a class which contains the linear segment parameters for the linear object state PDU

Public Methods

	DtLinearSegmentParameter () <i>Constructors</i>
	DtLinearSegmentParameter (const DtLinearSegmentParameter & orig)
DtLinearSegmentParameter &	operator= (const DtLinearSegmentParameter & orig) <i>Assignment operator</i>
DtLinearSegmentParameter &	operator= (const DtNetLinearSegmentParameter& orig)
virtual	~DtLinearSegmentParameter () <i>Destructor</i>
virtual void	setSegmentNumber (DtU8 value) <i>Set/Get the segment number</i>
virtual DtU8	segmentNumber () const
virtual void	setModifications (DtU8 value) <i>Set/Get the modifications (by 8-bit integer)</i>
virtual DtU8	modifications () const
virtual void	setModificationsField (DtModificationField field, DtBoolean value) <i>Set/Get the modifications (by field)</i>
virtual DtBoolean	modificationsField (DtModificationField field) const
virtual void	setAppearance (DtU16 app) <i>Set/Get appearance, the lower 16 bits (as an integer)</i>
virtual DtU16	appearance () const
virtual void	setAppearanceAttribute (DtAppearAttribute attr, DtAppearAttrVal val) <i>Set/Get appearance by attribute, the lower 16 bits</i>
virtual DtAppearAttrVal	appearanceAttribute (DtAppearAttribute attr) const
virtual void	setSpecificAppearance (DtU32 app) <i>Set/Get specific appearance, the upper 32 bits (as a 32 bit integer)</i>
virtual DtU32	specificAppearance () const
virtual void	setSpecificAppearanceAttribute (DtAppearAttribute attr, DtAppearAttrVal val) <i>Set/Get specific appearance by attribute, the upper 32 bits</i>
virtual DtAppearAttrVal	specificAppearanceAttribute (DtAppearAttribute attr) const
virtual void	setLocation (DtConstVector coords) <i>Set/Get location</i>
virtual DtConstVector	location () const

virtual void	setOrientation (const DtTaitBryan & orientation) <i>Set/Get orientation</i>
virtual const DtTaitBryan &	orientation () const
virtual void	setSegmentLength (DtU16 value) <i>Set/Get the segment length</i>
virtual DtU16	segmentLength () const
virtual void	setSegmentWidth (DtU16 value) <i>Set/Get the segment width</i>
virtual DtU16	segmentWidth () const
virtual void	setSegmentHeight (DtU16 value) <i>Set/Get the segment height</i>
virtual DtU16	segmentHeight () const
virtual void	setSegmentDepth (DtU16 value) <i>Set/Get the segment depth</i>
virtual DtU16	segmentDepth () const
virtual void	print () const <i>Print the information contained in the record</i>
virtual void	unloadNetRecord (const DtNetLinearSegmentParameter& record) <i>Load/Unload data from a packet record</i>
virtual void	loadNetRecord (DtNetLinearSegmentParameter& record) const

Protected Fields

DtU8	mySegmentNumber <i>Data members:</i>
DtU8	myModifications
DtU32	mySpecificAppearance
DtU16	myAppearance
DtVector	myLocation
DtTaitBryan	myOrientation
DtU16	mySegmentLength
DtU16	mySegmentWidth
DtU16	mySegmentHeight
DtU16	mySegmentDepth

Documentation

class DtLinearSegmentParameter: DtLinearSegmentParameter is a class which contains the linear segment parameters for the linear object state PDU

DtLinearSegmentParameter ()
Constructors

- `DtLinearSegmentParameter(const DtLinearSegmentParameter& orig)`
- `DtLinearSegmentParameter& operator=( const DtLinearSegmentParameter& orig)`
Assignment operator
- `DtLinearSegmentParameter& operator=( const DtNetLinearSegmentParameter& orig)`
- `virtual ~DtLinearSegmentParameter()`
Destructor
- `virtual void setSegmentNumber(DtU8 value)`
Set/Get the segment number
- `virtual DtU8 segmentNumber() const`
- `virtual void setModifications(DtU8 value)`
Set/Get the modifications (by 8-bit integer)
- `virtual DtU8 modifications() const`
- `virtual void setModificationsField( DtModificationField field, DtBoolean value)`
Set/Get the modifications (by field)
- `virtual DtBoolean modificationsField( DtModificationField field) const`
- `virtual void setAppearance(DtU16 app)`
Set/Get appearance, the lower 16 bits (as an integer)
- `virtual DtU16 appearance() const`
- `virtual void setAppearanceAttribute( DtAppearAttribute attr, DtAppearAttrVal val)`
Set/Get appearance by attribute, the lower 16 bits
- `virtual DtAppearAttrVal appearanceAttribute( DtAppearAttribute attr) const`
- `virtual void setSpecificAppearance(DtU32 app)`
Set/Get specific appearance, the upper 32 bits (as a 32 bit integer)
- `virtual DtU32 specificAppearance() const`
- `virtual void setSpecificAppearanceAttribute( DtAppearAttribute attr, DtAppearAttrVal val)`
Set/Get specific appearance by attribute, the upper 32 bits
- `virtual DtAppearAttrVal specificAppearanceAttribute( DtAppearAttribute attr) const`
- `virtual void setLocation(DtConstVector coords)`
Set/Get location
- `virtual DtConstVector location() const`
- `virtual void setOrientation(const DtTaitBryan& orientation)`
Set/Get orientation
- `virtual const DtTaitBryan& orientation() const`

- `virtual void setSegmentLength(DtU16 value)`
Set/Get the segment length
- `virtual DtU16 segmentLength() const`
- `virtual void setSegmentWidth(DtU16 value)`
Set/Get the segment width
- `virtual DtU16 segmentWidth() const`
- `virtual void setSegmentHeight(DtU16 value)`
Set/Get the segment height
- `virtual DtU16 segmentHeight() const`
- `virtual void setSegmentDepth(DtU16 value)`
Set/Get the segment depth
- `virtual DtU16 segmentDepth() const`
- `virtual void print() const`
Print the information contained in the record
- `virtual void unloadNetRecord( const DtNetLinearSegmentParameter& record)`
Load/Unload data from a packet record
- `virtual void loadNetRecord( DtNetLinearSegmentParameter& record) const`
- `DtU8 mySegmentNumber`
Data members:
- `DtU8 myModifications`
- `DtU32 mySpecificAppearance`
- `DtU16 myAppearance`
- `DtVector myLocation`
- `DtTaitBryan myOrientation`
- `DtU16 mySegmentLength`
- `DtU16 mySegmentWidth`
- `DtU16 mySegmentHeight`
- `DtU16 mySegmentDepth`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file letspipdu.h:

class DtLeTspiPdu: public DtPdu

Inheritance:

DtLeTspiPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtLeTspiPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtLeTspiPdu (const DtNetLeTspiPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtLeTspiPdu (const DtLeTspiPdu & orig)
 virtual	~DtLeTspiPdu () <i>destructor</i>
 DtLeTspiPdu &	operator= (const DtLeTspiPdu & orig) <i>Assignment operator</i>
 virtual void	setSubProtocol (DtSubProtocol value) <i>Set/Get the sub-protocol number</i>
 virtual DtSubProtocol	subProtocol () const
 virtual void	setEntityID (const DtLeEntityID & id) <i>Set/Get entity ID</i>
 virtual const DtLeEntityID &	entityID () const
 virtual DtU8	tspiField () const <i>Get TSPI Flag field (this field is automatically maintained by the PDU as other PDU fields are set)</i>
 virtual DtBoolean	tspiFlag (DtLE_TSPI_FieldFlag flag) const <i>Get the value for a specific TSPI flag from the flag field</i>
 virtual void	setLocation (const DtLeEntityLocationWorldCoordinates & value) <i>Set/Get entity location</i>
 virtual const DtLeEntityLocationWorldCoordinates &	location () const
 virtual void	setVelocity (const DtLeEntityLinearVelocity & value) <i>Set/Get entity linear velocity</i>
 virtual DtBoolean	getVelocity (DtLeEntityLinearVelocity & value) const
 virtual void	setOrientation (const DtLeEntityOrientation & value) <i>Set/Get entity orientation</i>
 virtual DtBoolean	getOrientation (DtLeEntityOrientation & value) const

- virtual void
- virtual [DtBoolean](#)
- virtual void
- virtual [DtBoolean](#)
- virtual void
- virtual [DtBoolean](#)
- virtual void
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual DtU8
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual void
- virtual void
- static [DtPdu](#)*
- static void
- static void

setPositionError (const [DtLePositionError](#)& value)
Set/Get position error

getPositionError ([DtLePositionError](#)& value) const

setOrientationError (const [DtLeOrientationError](#)& value)
Set/Get orientation error

getOrientationError ([DtLeOrientationError](#)& value) const

setDeadReckoningParameters (const [DtLeDeadReckoningParameters](#)& value)
Set/Get dead reckoning parameters

getDeadReckoningParameters ([DtLeDeadReckoningParameters](#)& value) const

setMeasuredSpeed (DtU16 value)
Set/Get measured speed

getMeasuredSpeed (DtU16& value) const

setSystemSpecificDataLength (DtU8 value)
Set/Get the system specific data length

systemSpecificDataLength () const

setSystemSpecificData (DtU8* dataPtr)
Set/Get the system specific data (will copy systemSpecificDataLength() bytes on set/get)

getSystemSpecificData (DtU8* dataPtr)

printData () const
Print the data contained by this PDU

printHeader () const
Print the header (override to include the new sub-protocol field)

create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Create a DtLeTspiPdu

addCallback ([DtExerciseConn](#)* conn, DtLeTspiPduCb cb, void* usr)
Add (register)/Remove (deregister) function to be called when pdu is received

removeCallback ([DtExerciseConn](#)* conn, DtLeTspiPduCb cb, void* usr)

Public

- The following accessor functions refer to records which are optional

Protected Fields

- static int **theStaticSize**
Size of PDU with zero parameter records
- [DtLeEntityID](#) **myEntityID**
Temporary objects for reference returns
- [DtLeEntityLocationWorldCoordinates](#) **myLocation**

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
virtual DtNetLeTspiPdu*	netPduStruct () const <i>Casts the netPdu element to our network struct</i>
DtLeTspiPdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type</i>
virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>
virtual void	setTspiField (DtU8 value) <i>Set TSPI Flags</i>
virtual void	setTspiFlag (DtLE_TSPI_FieldFlag flag) <i>Set a flag of the TSPI field</i>
virtual void	clearTspiFlags (DtLE_TSPI_FieldFlag flag) <i>Clear the flag and all following flags in the TSPI field</i>
virtual int	tspiFlagOffset () const <i>Returns the offset (in bytes) of a field from variable data region</i>
virtual int	locationOffset () const
virtual int	velocityOffset () const
virtual int	orientationOffset () const
virtual int	positionErrorOffset () const
virtual int	orientationErrorOffset () const
virtual int	deadReckoningParametersOffset () const
virtual int	measuredSpeedOffset () const
virtual int	systemSpecificOffset () const

Inherited from [DtPdu](#):

Public Methods

virtual void* packet () const
virtual int status () const
virtual void setVersion (int vers)
virtual int protocolVersion () const
virtual void setExerciseld (int id)
virtual int exerciseld () const
virtual DtPduKind kind () const
virtual DtProtocolFamily protocolFamily () const

- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtLeTspiPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtLeTspiPdu**(const DtNetLeTspiPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtLeTspiPdu**(const [DtLeTspiPdu](#)& orig)
- virtual ~DtLeTspiPdu**()
destructor

• [DtLeTspiPdu](#)& operator=(const [DtLeTspiPdu](#)& orig)
Assignment operator

• virtual void setSubProtocol([DtSubProtocol](#) value)
Set/Get the sub-protocol number

• virtual [DtSubProtocol](#) subProtocol() const

• virtual void setEntityID(const [DtLeEntityID](#)& id)
Set/Get entity ID

• virtual const [DtLeEntityID](#)& entityID() const

• virtual DtU8 tspiField() const
Get TSPI Flag field (this field is automatically maintained by the PDU as other PDU fields are set)

• virtual [DtBoolean](#) tspiFlag([DtLE_TSPI_FieldFlag](#) flag) const
Get the value for a specific TSPI flag from the flag field

• virtual void setLocation(const [DtLeEntityLocationWorldCoordinates](#)& value)
Set/Get entity location

• virtual const [DtLeEntityLocationWorldCoordinates](#)& location() const

•

The following accessor functions refer to records which are optional. The set methods create the record in the PDU if it does not already exist. The get methods return DtFalse if the record does not exist.

• virtual void setVelocity(const [DtLeEntityLinearVelocity](#)& value)
Set/Get entity linear velocity

• virtual [DtBoolean](#) getVelocity([DtLeEntityLinearVelocity](#)& value) const

• virtual void setOrientation(const [DtLeEntityOrientation](#)& value)
Set/Get entity orientation

• virtual [DtBoolean](#) getOrientation([DtLeEntityOrientation](#)& value) const

• virtual void setPositionError(const [DtLePositionError](#)& value)
Set/Get position error

• virtual [DtBoolean](#) getPositionError([DtLePositionError](#)& value) const

• virtual void setOrientationError(const [DtLeOrientationError](#)& value)
Set/Get orientation error

• virtual [DtBoolean](#) getOrientationError([DtLeOrientationError](#)& value) const

• virtual void setDeadReckoningParameters(const [DtLeDeadReckoningParameters](#)& value)
Set/Get dead reckoning parameters

• virtual [DtBoolean](#) getDeadReckoningParameters([DtLeDeadReckoningParameters](#)& value) const

• virtual void setMeasuredSpeed(DtU16 value)

Set/Get measured speed

virtual [DtBoolean](#) getMeasuredSpeed(DtU16& value) const

virtual [DtBoolean](#) setSystemSpecificDataLength(DtU8 value)

Set/Get the system specific data length

virtual DtU8 systemSpecificDataLength() const

virtual [DtBoolean](#) setSystemSpecificData(DtU8* dataPtr)

Set/Get the system specific data (will copy systemSpecificDataLength() bytes on set/get). Returns DtFalse if the length is zero.

virtual [DtBoolean](#) getSystemSpecificData(DtU8* dataPtr)

virtual void printData() const

Print the data contained by this PDU

virtual void printHeader() const

Print the header (override to include the new sub-protocol field)

static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtLeTspiPdu. Caller is responsible for deletion.

static void addCallback([DtExerciseConn](#)* conn, DtLeTspiPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

static void removeCallback([DtExerciseConn](#)* conn, DtLeTspiPduCb cb, void* usr)

virtual [DtPduKind](#) internalGetPduKind() const

Returns the internal PDU kind

virtual DtNetLeTspiPdu* netPduStruct() const

Casts the netPdu element to our network struct

[DtLeTspiPdu](#)* self() const

Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type

virtual char* var() const

Returns a pointer to the first byte of variable length data

virtual void setTspiField(DtU8 value)

Set TSPI Flags

virtual void setTspiFlag([DtLE_TSPI_FieldFlag](#) flag)

Set a flag of the TSPI field

virtual void clearTspiFlags([DtLE_TSPI_FieldFlag](#) flag)

Clear the flag and all following flags in the TSPI field

virtual int tspiFlagOffset() const

Returns the offset (in bytes) of a field from variable data region

- virtual int locationOffset() const
 - virtual int velocityOffset() const
 - virtual int orientationOffset() const
 - virtual int positionErrorOffset() const
 - virtual int orientationErrorOffset() const
 - virtual int deadReckoningParametersOffset() const
 - virtual int measuredSpeedOffset() const
 - virtual int systemSpecificOffset() const
 - static int theStaticSize
Size of PDU with zero parameter records
 - [DtLeEntityID](#) myEntityID
Temporary objects for reference returns
 - [DtLeEntityLocationWorldCoordinates](#) myLocation
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtSubProtocol

Live entity protocol PDU enums

 DtSubProtocolInvalid

Documentation

Live entity protocol PDU enums

 DtSubProtocolInvalid

[Alphabetic index](#) [Hierarchy of classes](#)

In file leentityid.h:

class DtLeEntityID

class DtLeEntityID: DtLeEntityID is a class which contains the entity identification information for the live entity PDUs

Public Methods

- **DtLeEntityID ()**
Constructors
- **DtLeEntityID** (const [DtLeEntityID](#)& orig)
- [DtLeEntityID](#)& **operator=** (const [DtLeEntityID](#)& orig)
Assignment operator
- [DtLeEntityID](#)& **operator=** (const DtNetLeEntityID& orig)
- virtual **~DtLeEntityID ()**
Destructor
- virtual void **setSite** (DtU8 value)
Set/Get site
- virtual DtU8 **site** () const
- virtual void **setApplication** (DtU8 value)
Set/Get application
- virtual DtU8 **application** () const
- virtual void **setEntity** (DtU16 value)
Set/Get entity
- virtual DtU16 **entity** () const
- virtual void **print** () const
Print the information contained in the record
- virtual void **unloadNetRecord** (const DtNetLeEntityID& record)
Load/Unload data from a packet record
- virtual void **loadNetRecord** (DtNetLeEntityID& record) const

Protected Fields

- DtU8 **mySite**
Data members:
 - DtU8 **myApplication**
 - DtU16 **myEntity**
-

Documentation

class DtLeEntityID: DtLeEntityID is a class which contains the entity identification information for the live entity PDUs

DtLeEntityID()

Constructors

DtLeEntityID(const [DtLeEntityID](#)& orig)

[DtLeEntityID](#)& operator=(const [DtLeEntityID](#)& orig)

Assignment operator

[DtLeEntityID](#)& operator=(const DtNetLeEntityID& orig)

virtual ~DtLeEntityID()

Destructor

virtual void setSite(DtU8 value)

Set/Get site

virtual DtU8 site() const

virtual void setApplication(DtU8 value)

Set/Get application

virtual DtU8 application() const

virtual void setEntity(DtU16 value)

Set/Get entity

virtual DtU16 entity() const

virtual void print() const

Print the information contained in the record

virtual void unloadNetRecord(const DtNetLeEntityID& record)

Load/Unload data from a packet record

virtual void loadNetRecord(DtNetLeEntityID& record) const

DtU8 mySite

Data members:

DtU8 myApplication

DtU16 myEntity

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtLE_TSPI_FieldFlag

- DtLE_TSPI_Reserved
 - DtLE_TSPI_SystemSpecific
 - DtLE_TSPI_MeasuredSpeed
 - DtLE_TSPI_DeadReckoning
 - DtLE_TSPI_OrientationError
 - DtLE_TSPI_PositionError
 - DtLE_TSPI_EntityOrientation
 - DtLE_TSPI_EntityLinearVelocity
-

Documentation

- DtLE_TSPI_Reserved
- DtLE_TSPI_SystemSpecific
- DtLE_TSPI_MeasuredSpeed
- DtLE_TSPI_DeadReckoning
- DtLE_TSPI_OrientationError
- DtLE_TSPI_PositionError
- DtLE_TSPI_EntityOrientation
- DtLE_TSPI_EntityLinearVelocity

In file lellocworld.h:

class DtLeEntityLocationWorldCoordinates

class DtLeEntityLocationWorldCoordinates: DtLeEntityLocationWorldCoordinates is a class which contains the entity location information for the live entity PDUs

Public Methods

	DtLeEntityLocationWorldCoordinates () <i>Constructors</i>
	DtLeEntityLocationWorldCoordinates (const DtLeEntityLocationWorldCoordinates& orig)
 DtLeEntityLocationWorldCoordinates &	operator= (const DtLeEntityLocationWorldCoordinates& orig) <i>Assignment operator</i>
 DtLeEntityLocationWorldCoordinates &	operator= (const DtNetLeEntityLocationWorldCoordinates& orig)
 virtual	~DtLeEntityLocationWorldCoordinates () <i>Destructor</i>
 virtual void	setReferencePoint (DtU16 value) <i>Set/Get the reference point</i>
 virtual DtU16	referencePoint () const
 virtual void	setDeltaX (DtU16 value) <i>Set/Get delta X</i>
 virtual DtU16	deltaX () const
 virtual void	setDeltaY (DtU16 value) <i>Set/Get delta Y</i>
 virtual DtU16	deltaY () const
 virtual void	setDeltaZ (DtU16 value) <i>Set/Get delta Z</i>
 virtual DtU16	deltaZ () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetLeEntityLocationWorldCoordinates& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetLeEntityLocationWorldCoordinates& record) const

Protected Fields

 DtU16	myReferencePoint <i>Data members:</i>
 DtU16	myDeltaX
 DtU16	myDeltaY

Documentation

class DtLeEntityLocationWorldCoordinates: DtLeEntityLocationWorldCoordinates is a class which contains the entity location information for the live entity PDUs

`DtLeEntityLocationWorldCoordinates()`

Constructors

`DtLeEntityLocationWorldCoordinates( const DtLeEntityLocationWorldCoordinates& orig)`

`DtLeEntityLocationWorldCoordinates& operator=( const`

`DtLeEntityLocationWorldCoordinates& orig)`

Assignment operator

`DtLeEntityLocationWorldCoordinates& operator=( const`

`DtNetLeEntityLocationWorldCoordinates& orig)`

`virtual ~DtLeEntityLocationWorldCoordinates()`

Destructor

`virtual void setReferencePoint(DtU16 value)`

Set/Get the reference point

`virtual DtU16 referencePoint() const`

`virtual void setDeltaX(DtU16 value)`

Set/Get delta X

`virtual DtU16 deltaX() const`

`virtual void setDeltaY(DtU16 value)`

Set/Get delta Y

`virtual DtU16 deltaY() const`

`virtual void setDeltaZ(DtU16 value)`

Set/Get delta Z

`virtual DtU16 deltaZ() const`

`virtual void print() const`

Print the information contained in the record

`virtual void unloadNetRecord( const DtNetLeEntityLocationWorldCoordinates& record)`

Load/Unload data from a packet record

`virtual void loadNetRecord( DtNetLeEntityLocationWorldCoordinates& record) const`

`DtU16 myReferencePoint`

Data members:

`DtU16 myDeltaX`

● DtU16 myDeltaY

● DtU16 myDeltaZ

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file leentityvel.h:

class DtLeEntityLinearVelocity

class DtLeEntityLinearVelocity: DtLeEntityLinearVelocity is a class which contains the entity linear velocity information for the live entity PDUs

Public Methods

	DtLeEntityLinearVelocity () <i>Constructors</i>
	DtLeEntityLinearVelocity (const DtLeEntityLinearVelocity & orig)
 DtLeEntityLinearVelocity &	operator= (const DtLeEntityLinearVelocity & orig) <i>Assignment operator</i>
 DtLeEntityLinearVelocity &	operator= (const DtNetLeEntityLinearVelocity& orig)
 virtual	~DtLeEntityLinearVelocity () <i>Destructor</i>
 virtual void	setX (DtU16 value) <i>Set/Get X</i>
 virtual DtU16	x () const
 virtual void	setY (DtU16 value) <i>Set/Get Y</i>
 virtual DtU16	y () const
 virtual void	setZ (DtU16 value) <i>Set/Get Z</i>
 virtual DtU16	z () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetLeEntityLinearVelocity& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetLeEntityLinearVelocity& record) const

Protected Fields

 DtU16	myX <i>Data members:</i>
 DtU16	myY
 DtU16	myZ

Documentation

class DtLeEntityLinearVelocity: DtLeEntityLinearVelocity is a class which contains the entity linear velocity information for the live entity PDUs

- `DtLeEntityLinearVelocity()`
Constructors
- `DtLeEntityLinearVelocity(const DtLeEntityLinearVelocity& orig)`
- `DtLeEntityLinearVelocity& operator=( const DtLeEntityLinearVelocity& orig)`
Assignment operator
- `DtLeEntityLinearVelocity& operator=( const DtNetLeEntityLinearVelocity& orig)`
- `virtual ~DtLeEntityLinearVelocity()`
Destructor
- `virtual void setX(DtU16 value)`
Set/Get X
- `virtual DtU16 x() const`
- `virtual void setY(DtU16 value)`
Set/Get Y
- `virtual DtU16 y() const`
- `virtual void setZ(DtU16 value)`
Set/Get Z
- `virtual DtU16 z() const`
- `virtual void print() const`
Print the information contained in the record
- `virtual void unloadNetRecord( const DtNetLeEntityLinearVelocity& record)`
Load/Unload data from a packet record
- `virtual void loadNetRecord( DtNetLeEntityLinearVelocity& record) const`
- `DtU16 myX`
Data members:
- `DtU16 myY`
- `DtU16 myZ`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file leentityori.h:

class DtLeEntityOrientation

class DtLeEntityOrientation: DtLeEntityOrientation is a class which contains the entity orientation information for the live entity PDUs

Public Methods

	DtLeEntityOrientation () <i>Constructors</i>
	DtLeEntityOrientation (const DtLeEntityOrientation & orig)
 DtLeEntityOrientation &	operator= (const DtLeEntityOrientation & orig) <i>Assignment operator</i>
 DtLeEntityOrientation &	operator= (const DtNetLeEntityOrientation& orig)
 virtual	~DtLeEntityOrientation () <i>Destructor</i>
 virtual void	setPsi (DtU16 value) <i>Set/Get Psi</i>
 virtual DtU16	psi () const
 virtual void	setTheta (DtU16 value) <i>Set/Get Theta</i>
 virtual DtU16	theta () const
 virtual void	setPhi (DtU16 value) <i>Set/Get Phi</i>
 virtual DtU16	phi () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetLeEntityOrientation& record) <i>Load/Unload data from a packet record (for 8-bit fields)</i>
 virtual void	loadNetRecord (DtNetLeEntityOrientation& record) const
 virtual void	unloadNetRecord16 (const DtNetLeEntityOrientation16& record) <i>Load/Unload data from a packet record (for 16-bit fields)</i>
 virtual void	loadNetRecord16 (DtNetLeEntityOrientation16& record) const

Protected Fields

 DtU16	myPsi <i>Data members:</i>
 DtU16	myTheta
 DtU16	myPhi

Documentation

class DtLeEntityOrientation:

DtLeEntityOrientation is a class which contains the entity orientation information for the live entity PDUs. This record uses 8-bit values for the TSPI PDU and 16-bit values for the detonation PDU.

`DtLeEntityOrientation()`

Constructors

`DtLeEntityOrientation(const DtLeEntityOrientation& orig)`

`DtLeEntityOrientation& operator=( const DtLeEntityOrientation& orig)`

Assignment operator

`DtLeEntityOrientation& operator=( const DtNetLeEntityOrientation& orig)`

`virtual ~DtLeEntityOrientation()`

Destructor

`virtual void setPsi(DtU16 value)`

Set/Get Psi

`virtual DtU16 psi() const`

`virtual void setTheta(DtU16 value)`

Set/Get Theta

`virtual DtU16 theta() const`

`virtual void setPhi(DtU16 value)`

Set/Get Phi

`virtual DtU16 phi() const`

`virtual void print() const`

Print the information contained in the record

`virtual void unloadNetRecord( const DtNetLeEntityOrientation& record)`

Load/Unload data from a packet record (for 8-bit fields)

`virtual void loadNetRecord( DtNetLeEntityOrientation& record) const`

`virtual void unloadNetRecord16( const DtNetLeEntityOrientation16& record)`

Load/Unload data from a packet record (for 16-bit fields)

`virtual void loadNetRecord16( DtNetLeEntityOrientation16& record) const`

`DtU16 myPsi`

Data members:

`DtU16 myTheta`

`DtU16 myPhi`

This class has no child classes.

[*Alphabetic index*](#) [*Hierarchy of classes*](#)

In file leposeerror.h:

class DtLePositionError

class DtLePositionError: DtLePositionError is a class which contains the entity location information for the live entity PDUs

Public Methods

-  **DtLePositionError ()**
Constructors
-  **DtLePositionError** (const [DtLePositionError](#)& orig)
-  [DtLePositionError](#)& **operator=** (const [DtLePositionError](#)& orig)
Assignment operator
-  [DtLePositionError](#)& **operator=** (const DtNetLePositionError& orig)
-  virtual **~DtLePositionError ()**
Destructor
-  virtual void **setHorizontalError** (DtU16 value)
Set/Get horizontal error
-  virtual DtU16 **horizontalError** () const
-  virtual void **setVerticalError** (DtU16 value)
Set/Get vertical error
-  virtual DtU16 **verticalError** () const
-  virtual void **print** () const
Print the information contained in the record
-  virtual void **unloadNetRecord** (const DtNetLePositionError& record)
Load/Unload data from a packet record
-  virtual void **loadNetRecord** (DtNetLePositionError& record) const

Protected Fields

-  DtU16 **myHorizontalError**
Data members:
 -  DtU16 **myVerticalError**
-

Documentation

class DtLePositionError: DtLePositionError is a class which contains the entity location information for the live entity PDUs

-  **DtLePositionError ()**
Constructors

- `DtLePositionError(const DtLePositionError& orig)`
 - `DtLePositionError& operator=( const DtLePositionError& orig)`
Assignment operator
 - `DtLePositionError& operator=( const DtNetLePositionError& orig)`
 - `virtual ~DtLePositionError()`
Destructor
 - `virtual void setHorizontalError(DtU16 value)`
Set/Get horizontal error
 - `virtual DtU16 horizontalError() const`
 - `virtual void setVerticalError(DtU16 value)`
Set/Get vertical error
 - `virtual DtU16 verticalError() const`
 - `virtual void print() const`
Print the information contained in the record
 - `virtual void unloadNetRecord( const DtNetLePositionError& record)`
Load/Unload data from a packet record
 - `virtual void loadNetRecord( DtNetLePositionError& record) const`
 - `DtU16 myHorizontalError`
Data members:
 - `DtU16 myVerticalError`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file leorierror.h:

class DtLeOrientationError

class DtLeOrientationError: DtLeOrientationError is a class which contains the entity orientation error information for the live entity PDUs

Public Methods

	DtLeOrientationError () <i>Constructors</i>
	DtLeOrientationError (const DtLeOrientationError & orig)
 DtLeOrientationError &	operator= (const DtLeOrientationError & orig) <i>Assignment operator</i>
 DtLeOrientationError &	operator= (const DtNetLeOrientationError& orig)
 virtual	~DtLeOrientationError () <i>Destructor</i>
 virtual void	setAzimuthError (DtU16 value) <i>Set/Get azimuth error</i>
 virtual DtU16	azimuthError () const
 virtual void	setElevationError (DtU16 value) <i>Set/Get elevation error</i>
 virtual DtU16	elevationError () const
 virtual void	setRotationError (DtU16 value) <i>Set/Get rotation error</i>
 virtual DtU16	rotationError () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetLeOrientationError& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetLeOrientationError& record) const

Protected Fields

 DtU16	myAzimuthError <i>Data members:</i>
 DtU16	myElevationError
 DtU16	myRotationError

Documentation

class DtLeOrientationError: DtLeOrientationError is a class which contains the entity orientation error information for the live entity PDUs

- `DtLeOrientationError()`
Constructors
- `DtLeOrientationError(const DtLeOrientationError& orig)`
- `DtLeOrientationError& operator=( const DtLeOrientationError& orig)`
Assignment operator
- `DtLeOrientationError& operator=( const DtNetLeOrientationError& orig)`
- `virtual ~DtLeOrientationError()`
Destructor
- `virtual void setAzimuthError(DtU16 value)`
Set/Get azimuth error
- `virtual DtU16 azimuthError() const`
- `virtual void setElevationError(DtU16 value)`
Set/Get elevation error
- `virtual DtU16 elevationError() const`
- `virtual void setRotationError(DtU16 value)`
Set/Get rotation error
- `virtual DtU16 rotationError() const`
- `virtual void print() const`
Print the information contained in the record
- `virtual void unloadNetRecord( const DtNetLeOrientationError& record)`
Load/Unload data from a packet record
- `virtual void loadNetRecord( DtNetLeOrientationError& record) const`
- `DtU16 myAzimuthError`
Data members:
- `DtU16 myElevationError`
- `DtU16 myRotationError`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file ledeadreck.h:

class DtLeDeadReckoningParameters

class DtLeDeadReckoningParameters: DtLeDeadReckoningParameters is a class which contains the entity dead reckoning parameter information for the live entity PDUs

Public Methods

	DtLeDeadReckoningParameters () <i>Constructors</i>
	DtLeDeadReckoningParameters (const DtLeDeadReckoningParameters & orig)
 DtLeDeadReckoningParameters &	operator= (const DtLeDeadReckoningParameters & orig) <i>Assignment operator</i>
 DtLeDeadReckoningParameters &	operator= (const DtNetLeDeadReckoningParameters& orig)
 virtual	~DtLeDeadReckoningParameters () <i>Destructor</i>
 virtual void	setDeadReckoningAlgorithm (DtDeadReckonTypes value) <i>Set/Get dead reckoning algorithm</i>
 DtDeadReckonTypes	deadReckoningAlgorithm () const
 virtual void	setLinearAcceleration (DtU32 value) <i>Set/Get entity linear acceleration</i>
 virtual DtU32	linearAcceleration () const
 virtual void	setAngularVelocity (DtU32 value) <i>Set/Get entity angular velocity</i>
 virtual DtU32	angularVelocity () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetLeDeadReckoningParameters& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetLeDeadReckoningParameters& record) const

Protected Fields

 DtDeadReckonTypes	myDeadReckoningAlgorithm <i>Data members:</i>
 DtU32	myLinearAcceleration
 DtU32	myAngularVelocity

Documentation

class DtLeDeadReckoningParameters: DtLeDeadReckoningParameters is a class which contains the entity dead reckoning parameter information for the live entity PDUs

- `DtLeDeadReckoningParameters()`
Constructors
- `DtLeDeadReckoningParameters(const DtLeDeadReckoningParameters& orig)`
- `DtLeDeadReckoningParameters& operator=(const DtLeDeadReckoningParameters& orig)`
Assignment operator
- `DtLeDeadReckoningParameters& operator=(const DtNetLeDeadReckoningParameters& orig)`
- `virtual ~DtLeDeadReckoningParameters()`
Destructor
- `virtual void setDeadReckoningAlgorithm(DtDeadReckonTypes value)`
Set/Get dead reckoning algorithm
- `virtual DtDeadReckonTypes deadReckoningAlgorithm() const`
- `virtual void setLinearAcceleration(DtU32 value)`
Set/Get entity linear acceleration. This PDU field is only 24-bits wide.
- `virtual DtU32 linearAcceleration() const`
- `virtual void setAngularVelocity(DtU32 value)`
Set/Get entity angular velocity. This PDU field is only 24-bits wide.
- `virtual DtU32 angularVelocity() const`
- `virtual void print() const`
Print the information contained in the record
- `virtual void unloadNetRecord(const DtNetLeDeadReckoningParameters& record)`
Load/Unload data from a packet record
- `virtual void loadNetRecord(DtNetLeDeadReckoningParameters& record) const`
- `DtDeadReckonTypes myDeadReckoningAlgorithm`
Data members:
- `DtU32 myLinearAcceleration`
- `DtU32 myAngularVelocity`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtDeadReckonTypes

Section 7

-  DtDrOther
 -  DtDrStatic
 -  DtDrDrmFpw
 -  DtDrDrmRpw
 -  DtDrDrmRvw
 -  DtDrDrmFvw
 -  DtDrDrmFpb
 -  DtDrDrmRpb
 -  DtDrDrmRvb
 -  DtDrDrmFvb
-

Documentation

Section 7

-  DtDrOther
-  DtDrStatic
-  DtDrDrmFpw
-  DtDrDrmRpw
-  DtDrDrmRvw
-  DtDrDrmFvw
-  DtDrDrmFpb
-  DtDrDrmRpb
-  DtDrDrmRvb
-  DtDrDrmFvb

In file lefirepdu.h:

class DtLeFirePdu: public DtPdu

Inheritance:

DtLeFirePdu - DtPdu - DtBaseMsg

Public Methods

	DtLeFirePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtLeFirePdu (const DtNetLeFirePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtLeFirePdu (const <u>DtLeFirePdu</u> & orig)
 virtual	~DtLeFirePdu () <i>destructor</i>
 <u>DtLeFirePdu</u> &	operator= (const <u>DtLeFirePdu</u> & orig) <i>Assignment operator</i>
 virtual void	setSubProtocol (<u>DtSubProtocol</u> value) <i>Set/Get the sub-protocol number</i>
 virtual <u>DtSubProtocol</u>	subProtocol () const
 virtual void	setFiringEntityID (const <u>DtLeEntityID</u> & id) <i>Set/Get firing entity ID</i>
 virtual const <u>DtLeEntityID</u> &	firingEntityID () const
 virtual DtU8	fireFlagsField () const <i>Get flags field (this field is automatically maintained by the PDU as other PDU fields are set)</i>
 virtual <u>DtBoolean</u>	fireFlag (<u>DtLE_FireFieldFlag</u> flag) const <i>Get the value for a specific flag from the flags field</i>
 virtual void	setTargetEntityID (const <u>DtLeEntityID</u> & id) <i>Set/Get target entity ID (optional)</i>
 virtual <u>DtBoolean</u>	getTargetEntityID (<u>DtLeEntityID</u> & id) const
 virtual void	setMunitionID_Site (DtU8 value) <i>Set/Get munition ID site (optional)</i>
 virtual <u>DtBoolean</u>	getMunitionID_Site (DtU8& value) const
 virtual void	setMunitionID_Application (DtU8 value) <i>Set/Get munition ID application (optional)</i>
 virtual <u>DtBoolean</u>	getMunitionID_Application (DtU8& value) const

virtual void	setMunitionID_Entity (DtU16 value) <i>Set/Get munition ID entity (optional)</i>
virtual DtBoolean	getMunitionID_Entity (DtU16& value) const
virtual void	setEventID_Site (DtU8 value) <i>Set/Get event ID site (optional)</i>
virtual DtBoolean	getEventID_Site (DtU8& value) const
virtual void	setEventID_Application (DtU8 value) <i>Set/Get event ID application (optional)</i>
virtual DtBoolean	getEventID_Application (DtU8& value) const
virtual void	setEventID_Event (DtU16 value) <i>Set/Get event ID event</i>
virtual DtU16	eventID_Event () const
virtual void	setLocation (const DtLeEntityLocationWorldCoordinates & value) <i>Set/Get entity location (optional)</i>
virtual DtBoolean	getLocation (DtLeEntityLocationWorldCoordinates & value) const
virtual void	setBurstMunitionType (DtEntityType & type) <i>Set/Get burst munition type</i>
virtual const DtEntityType &	burstMunitionType () const
virtual void	setBurstWarhead (DtU16 value) <i>Set/Get burst warhead (optional)</i>
virtual DtBoolean	getBurstWarhead (DtU16& value) const
virtual void	setBurstFuse (DtU16 value) <i>Set/Get burst fuse (optional)</i>
virtual DtBoolean	getBurstFuse (DtU16& value) const
virtual void	setBurstQuantity (DtU16 value) <i>Set/Get burst quantity (optional)</i>
virtual DtBoolean	getBurstQuantity (DtU16& value) const
virtual void	setBurstRate (DtU16 value) <i>Set/Get burst rate (optional)</i>
virtual DtBoolean	getBurstRate (DtU16& value) const
virtual void	setVelocity (const DtLeEntityLinearVelocity & value) <i>Set/Get entity linear velocity</i>
virtual const DtLeEntityLinearVelocity &	velocity () const
virtual void	setRange (DtU16 value) <i>Set/Get range</i>
virtual DtU16	range () const
virtual void	printData () const <i>Print the data contained by this PDU</i>
virtual void	printHeader () const <i>Print the header (override to include the new sub-protocol field)</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtLeFirePdu</i>
static void	addCallback (DtExerciseConn * conn, DtLeFirePduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>

static void

removeCallback ([DtExerciseConn](#)* conn, DtLeFirePduCb cb, void* usr)

Public

Some of following accessor functions refer to records which are optional

Protected Fields

static int **theStaticSize**
Size of PDU with zero parameter records

[DtLeEntityID](#) **myEntityID**
Temporary objects for reference returns

[DtLeEntityLinearVelocity](#) **myVelocity**

[DtEntityType](#) **myEntityType**

Protected Methods

virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind

virtual DtNetLeFirePdu* **netPduStruct** () const
Casts the netPdu element to our network struct

[DtLeFirePdu](#)* **self** () const
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type

virtual char* **var** () const
Returns a pointer to the first byte of variable length data

virtual void **setFireFlagsField** (DtU8 value)
Set Fire Flags

virtual void **setFireFlag** ([DtLE_FireFieldFlag](#) flag)
Set a flag of the flag field

virtual void **clearFireFlags** ([DtLE_FireFieldFlag](#) flag)
Clear the flag and all following flags in the fire flags field

virtual int **fireFlagFieldOffset** () const
Returns the offset (in bytes) of a field from variable data region

virtual int **targetEntityID_Offset** () const

virtual int **munitionID_Offset1** () const

virtual int **munitionID_Offset2** () const

virtual int **eventID_Offset1** () const

virtual int **eventID_Offset2** () const

virtual int **locationOffset** () const

virtual int **burstOffset1** () const

virtual int **burstOffset2** () const

virtual int **burstOffset3** () const

virtual int	velocityOffset () const
virtual int	rangeOffset () const

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const

inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

int **myOKStatus**

Protected Methods

virtual void **setStatus**(int okStatus)

virtual DtNetPduHeader* **netHeader**() const

virtual void **initPdu**(int size, DtBufferPtr buffer)

virtual void **initPdu**(const void* initial, DtBufferPtr buffer)

virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const

virtual void **sizeChanged**(int size)

virtual void **clearData**()

virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

virtual const char* **netRep**() const

virtual int **length**() const

inline const char* **netRep**() const

Protected Fields

void* **myNetMessage**

void* **myInternalBuff**

int **myIntBuffSize**

int **myMsgSize**

Protected Methods

void **initMsg**(int size, DtBufferPtr bufferPtr)

void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)

virtual void **changeSize**(int size)

- virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtLeFirePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Constructors
- `DtLeFirePdu(const DtNetLeFirePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtLeFirePdu(const DtLeFirePdu& orig)`
- `virtual ~DtLeFirePdu()`
destructor
- `DtLeFirePdu& operator=(const DtLeFirePdu& orig)`
Assignment operator
- `virtual void setSubProtocol(DtSubProtocol value)`
Set/Get the sub-protocol number
- `virtual DtSubProtocol subProtocol() const`
- `virtual void setFiringEntityID(const DtLeEntityID& id)`
Set/Get firing entity ID
- `virtual const DtLeEntityID& firingEntityID() const`
- `virtual DtU8 fireFlagsField() const`
Get flags field (this field is automatically maintained by the PDU as other PDU fields are set)
- `virtual DtBoolean fireFlag(DtLE\_FireFieldFlag flag) const`
Get the value for a specific flag from the flags field
- Some of following accessor functions refer to records which are optional. The set methods create the record in the PDU if it does not already exist. The get methods return `DtFalse` if the record does not exist.
- `virtual void setTargetEntityID(const DtLeEntityID& id)`
Set/Get target entity ID (optional)
- `virtual DtBoolean getTargetEntityID(DtLeEntityID& id) const`
- `virtual void setMunitionID_Site(DtU8 value)`

Set/Get munition ID site (optional)

virtual [DtBoolean](#) getMunitionID_Site(DtU8& value) const

virtual void setMunitionID_Application(DtU8 value)

Set/Get munition ID application (optional)

virtual [DtBoolean](#) getMunitionID_Application(DtU8& value) const

virtual void setMunitionID_Entity(DtU16 value)

Set/Get munition ID entity (optional)

virtual [DtBoolean](#) getMunitionID_Entity(DtU16& value) const

virtual void setEventID_Site(DtU8 value)

Set/Get event ID site (optional)

virtual [DtBoolean](#) getEventID_Site(DtU8& value) const

virtual void setEventID_Application(DtU8 value)

Set/Get event ID application (optional)

virtual [DtBoolean](#) getEventID_Application(DtU8& value) const

virtual void setEventID_Event(DtU16 value)

Set/Get event ID event

virtual DtU16 eventID_Event() const

virtual void setLocation(const [DtLeEntityLocationWorldCoordinates](#)& value)

Set/Get entity location (optional)

virtual [DtBoolean](#) getLocation([DtLeEntityLocationWorldCoordinates](#)& value) const

virtual void setBurstMunitionType([DtEntityType](#)& type)

Set/Get burst munition type

virtual const [DtEntityType](#)& burstMunitionType() const

virtual void setBurstWarhead(DtU16 value)

Set/Get burst warhead (optional)

virtual [DtBoolean](#) getBurstWarhead(DtU16& value) const

virtual void setBurstFuse(DtU16 value)

Set/Get burst fuse (optional)

virtual [DtBoolean](#) getBurstFuse(DtU16& value) const

virtual void setBurstQuantity(DtU16 value)

Set/Get burst quantity (optional)

virtual [DtBoolean](#) getBurstQuantity(DtU16& value) const

virtual void setBurstRate(DtU16 value)

Set/Get burst rate (optional)

- `virtual DtBoolean getBurstRate(DtU16& value) const`
- `virtual void setVelocity(const DtLeEntityLinearVelocity& value)`
Set/Get entity linear velocity
- `virtual const DtLeEntityLinearVelocity& velocity() const`
- `virtual void setRange(DtU16 value)`
Set/Get range
- `virtual DtU16 range() const`
- `virtual void printData() const`
Print the data contained by this PDU
- `virtual void printHeader() const`
Print the header (override to include the new sub-protocol field)
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtLeFirePdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtLeFirePduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtLeFirePduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetLeFirePdu* netPduStruct() const`
Casts the netPdu element to our network struct
- `DtLeFirePdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
- `virtual char* var() const`
Returns a pointer to the first byte of variable length data
- `virtual void setFireFlagsField(DtU8 value)`
Set Fire Flags
- `virtual void setFireFlag(DtLE\_FireFieldFlag flag)`
Set a flag of the flag field
- `virtual void clearFireFlags(DtLE\_FireFieldFlag flag)`
Clear the flag and all following flags in the fire flags field
- `virtual int fireFlagFieldOffset() const`
Returns the offset (in bytes) of a field from variable data region. Some fields have subfields which are optional, so there are multiple
- `virtual int targetEntityID_Offset() const`

- virtual int munitionID_Offset1() const
- virtual int munitionID_Offset2() const
- virtual int eventID_Offset1() const
- virtual int eventID_Offset2() const
- virtual int locationOffset() const
- virtual int burstOffset1() const
- virtual int burstOffset2() const
- virtual int burstOffset3() const
- virtual int velocityOffset() const
- virtual int rangeOffset() const
- static int theStaticSize
Size of PDU with zero parameter records
- [DtLeEntityID](#) myEntityID
Temporary objects for reference returns
- [DtLeEntityLinearVelocity](#) myVelocity
- [DtEntityType](#) myEntityType

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtLE_FireFieldFlag

- DtLE_FireLocation
 - DtLE_FireBurstQuantityRate
 - DtLE_FireBurstWarheadFuse
 - DtLE_FireEventSiteApp
 - DtLE_FireMunitionID
 - DtLE_FireMunitionSiteApp
 - DtLE_FireTargetID
-

Documentation

- DtLE_FireLocation
- DtLE_FireBurstQuantityRate
- DtLE_FireBurstWarheadFuse
- DtLE_FireEventSiteApp
- DtLE_FireMunitionID
- DtLE_FireMunitionSiteApp
- DtLE_FireTargetID

In file ledetonpdu.h:

class DtLeDetonationPdu: public DtPdu

Inheritance:

DtLeDetonationPdu - DtPdu - DtBaseMsg

Public Methods

	DtLeDetonationPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtLeDetonationPdu (const DtNetLeDetonationPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtLeDetonationPdu (const DtLeDetonationPdu & orig)
 virtual	~DtLeDetonationPdu () <i>destructor</i>
 DtLeDetonationPdu &	operator= (const DtLeDetonationPdu & orig) <i>Assignment operator</i>
 virtual void	setSubProtocol (DtSubProtocol value) <i>Set/Get the sub-protocol number</i>
 virtual DtSubProtocol	subProtocol () const
 virtual void	setFiringEntityID (const DtLeEntityID & id) <i>Set/Get firing entity ID</i>
 virtual const DtLeEntityID &	firingEntityID () const
 virtual DtU8	detonFlagsField () const <i>Get flags field 1 (this field is automatically maintained by the PDU as other PDU fields are set)</i>
 virtual DtBoolean	getDeton2FlagsField (DtU8& field) const <i>Get flags field 2 (this field is automatically maintained by the PDU as other PDU fields are set)</i>
 virtual DtBoolean	detonFlag (DtLE_DetonFieldFlag flag) const <i>Get the value for a specific flag from the first flags field</i>
 virtual DtBoolean	getDeton2Flag (DtLE_DetonField2Flag flag, DtBoolean & value) const <i>Get the value for a specific flag from the second flags field</i>
 virtual void	setTargetEntityID (const DtLeEntityID & id) <i>Set/Get target entity ID (optional)</i>
 virtual DtBoolean	getTargetEntityID (DtLeEntityID & id) const
 virtual void	setMunitionID_Site (DtU8 value) <i>Set/Get munition ID site (optional)</i>

virtual DtBoolean	getMunitionID_Site (DtU8& value) const
virtual void	setMunitionID_Application (DtU8 value) <i>Set/Get munition ID application (optional)</i>
virtual DtBoolean	getMunitionID_Application (DtU8& value) const
virtual void	setMunitionID_Entity (DtU16 value) <i>Set/Get munition ID entity (optional)</i>
virtual DtBoolean	getMunitionID_Entity (DtU16& value) const
virtual void	setEventID_Site (DtU8 value) <i>Set/Get event ID site (optional)</i>
virtual DtBoolean	getEventID_Site (DtU8& value) const
virtual void	setEventID_Application (DtU8 value) <i>Set/Get event ID application (optional)</i>
virtual DtBoolean	getEventID_Application (DtU8& value) const
virtual void	setEventID_Event (DtU16 value) <i>Set/Get event ID event</i>
virtual DtBoolean	getEventID_Event (DtU16& value) const
virtual void	setLocationWorld (const DtLeEntityLocationWorldCoordinates & value) <i>Set/Get entity location (world coordinates)</i>
virtual DtBoolean	getLocationWorld (DtLeEntityLocationWorldCoordinates & value) const
virtual void	setLocationEntity (const DtLeEntityLocationEntityCoordinates & value) <i>Set/Get entity location (entity coordinates)</i>
virtual DtBoolean	getLocationEntity (DtLeEntityLocationEntityCoordinates & value) const
virtual void	setVelocity (const DtLeEntityLinearVelocity & value) <i>Set/Get entity linear velocity</i>
virtual const DtLeEntityLinearVelocity &	velocity () const
virtual void	setOrientation (const DtLeEntityOrientation & value) <i>Set/Get entity orientation</i>
virtual DtBoolean	getOrientation (DtLeEntityOrientation & value) const
virtual void	setBurstMunitionType (DtEntityType & type) <i>Set/Get burst munition type</i>
virtual const DtEntityType &	burstMunitionType () const
virtual void	setBurstWarhead (DtU16 value) <i>Set/Get burst warhead (optional)</i>
virtual DtBoolean	getBurstWarhead (DtU16& value) const
virtual void	setBurstFuse (DtU16 value) <i>Set/Get burst fuse (optional)</i>
virtual DtBoolean	getBurstFuse (DtU16& value) const
virtual void	setBurstQuantity (DtU16 value) <i>Set/Get burst quantity (optional)</i>
virtual DtBoolean	getBurstQuantity (DtU16& value) const
virtual void	setBurstRate (DtU16 value) <i>Set/Get burst rate (optional)</i>
virtual DtBoolean	getBurstRate (DtU16& value) const

virtual void	setDetonationResult (DtU8 value) <i>Set/Get detonation result</i>
virtual DtU8	detonationResult () const
virtual void	printData () const <i>Print the data contained by this PDU</i>
virtual void	printHeader () const <i>Print the header (override to include the new sub-protocol field)</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtLeDetonationPdu</i>
static void	addCallback (DtExerciseConn * conn, DtLeDetonationPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtLeDetonationPduCb cb, void* usr)

Public

- Some of following accessor functions refer to records which are optional

Protected Fields

static int	theStaticSize <i>Size of PDU with zero parameter records</i>
DtLeEntityID	myEntityID <i>Temporary objects for reference returns</i>
DtLeEntityLinearVelocity	myVelocity
DtEntityType	myEntityType

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
virtual DtNetLeDetonationPdu*	netPduStruct () const <i>Casts the netPdu element to our network struct</i>
DtLeDetonationPdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type</i>
virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>
virtual void	setDetonFlagsField (DtU8 value) <i>Set first detonation flags field</i>
virtual void	setDeton2FlagsField (DtU8 value) <i>Set second detonation flags field</i>
virtual void	setDetonFlag (DtLE_DetonFieldFlag flag) <i>Set first detonation flag</i>
virtual void	setDeton2Flag (DtLE_DetonField2Flag flag) <i>Set second detonation flag</i>

virtual int	detonFlagFieldOffset () const <i>Returns the offset (in bytes) of a field from variable data region</i>
virtual int	deton2FlagFieldOffset () const
virtual int	targetEntityID_Offset () const
virtual int	munitionID_Offset1 () const
virtual int	munitionID_Offset2 () const
virtual int	eventID_Offset1 () const
virtual int	eventID_Offset2 () const
virtual int	locationWorldOffset () const
virtual int	velocityOffset () const
virtual int	munitionOrientationOffset () const
virtual int	burstOffset1 () const
virtual int	burstOffset2 () const
virtual int	burstOffset3 () const
virtual int	locationEntityOffset () const
virtual int	resultOffset () const

Inherited from [DtPdu](#):

Public Methods

virtual void*	packet () const
virtual int	status () const
virtual void	setVersion (int vers)
virtual int	protocolVersion () const
virtual void	setExerciseld (int id)
virtual int	exerciseld () const
virtual DtPduKind	kind () const
virtual DtProtocolFamily	protocolFamily () const
virtual void	setTimeStamp (DtTime time, DtTimeStampType type = DtTimeStampRelative)
virtual DtTime	timeStamp () const
virtual DtTimeStampType	timeStampType () const
virtual DtTime	guessTimeValid (DtTime referenceTime) const
virtual void	print () const

- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtLeDetonationPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtLeDetonationPdu**(const DtNetLeDetonationPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtLeDetonationPdu**(const [DtLeDetonationPdu](#)& orig)
- virtual ~DtLeDetonationPdu**()
destructor
- [DtLeDetonationPdu](#)**& operator=(const [DtLeDetonationPdu](#)& orig)
Assignment operator

virtual void setSubProtocol([DtSubProtocol](#) value)

Set/Get the sub-protocol number

virtual [DtSubProtocol](#) subProtocol() const

virtual void setFiringEntityID(const [DtLeEntityID](#)& id)

Set/Get firing entity ID

virtual const [DtLeEntityID](#)& firingEntityID() const

virtual DtU8 detonFlagsField() const

Get flags field 1 (this field is automatically maintained by the PDU as other PDU fields are set)

virtual [DtBoolean](#) getDeton2FlagsField(DtU8& field) const

Get flags field 2 (this field is automatically maintained by the PDU as other PDU fields are set). Returns DtFalse if this field is not present.

virtual [DtBoolean](#) detonFlag([DtLE_DetonFieldFlag](#) flag) const

Get the value for a specific flag from the first flags field

virtual [DtBoolean](#) getDeton2Flag([DtLE_DetonField2Flag](#) flag, [DtBoolean](#)& value) const

Get the value for a specific flag from the second flags field. Returns DtFalse if the field is not present in the PDU.

Some of following accessor functions refer to records which are optional. The set methods create the record in the PDU if it does not already exist. The get methods return DtFalse if the record does not exist.

virtual void setTargetEntityID(const [DtLeEntityID](#)& id)

Set/Get target entity ID (optional)

virtual [DtBoolean](#) getTargetEntityID([DtLeEntityID](#)& id) const

virtual void setMunitionID_Site(DtU8 value)

Set/Get munition ID site (optional)

virtual [DtBoolean](#) getMunitionID_Site(DtU8& value) const

virtual void setMunitionID_Application(DtU8 value)

Set/Get munition ID application (optional)

virtual [DtBoolean](#) getMunitionID_Application(DtU8& value) const

virtual void setMunitionID_Entity(DtU16 value)

Set/Get munition ID entity (optional)

virtual [DtBoolean](#) getMunitionID_Entity(DtU16& value) const

virtual void setEventID_Site(DtU8 value)

Set/Get event ID site (optional)

virtual [DtBoolean](#) getEventID_Site(DtU8& value) const

virtual void setEventID_Application(DtU8 value)

Set/Get event ID application (optional)

- virtual [DtBoolean](#) getEventID_Application(DtU8& value) const
- virtual void setEventID_Event(DtU16 value)
Set/Get event ID event
- virtual [DtBoolean](#) getEventID_Event(DtU16& value) const
- virtual void setLocationWorld(const [DtLeEntityLocationWorldCoordinates](#)& value)
Set/Get entity location (world coordinates). In this PDU, entity location is stored in world coordinates or entity coordinates but not both. Get returns DtFalse if the coordinates are stored in entity coordinates.
- virtual [DtBoolean](#) getLocationWorld([DtLeEntityLocationWorldCoordinates](#)& value) const
- virtual void setLocationEntity(const [DtLeEntityLocationEntityCoordinates](#)& value)
Set/Get entity location (entity coordinates). In this PDU, entity location is stored in world coordinates or entity coordinates but not both. Get returns DtFalse if the coordinates are stored in world coordinates.
- virtual [DtBoolean](#) getLocationEntity([DtLeEntityLocationEntityCoordinates](#)& value) const
- virtual void setVelocity(const [DtLeEntityLinearVelocity](#)& value)
Set/Get entity linear velocity
- virtual const [DtLeEntityLinearVelocity](#)& velocity() const
- virtual void setOrientation(const [DtLeEntityOrientation](#)& value)
Set/Get entity orientation
- virtual [DtBoolean](#) getOrientation([DtLeEntityOrientation](#)& value) const
- virtual void setBurstMunitionType([DtEntityType](#)& type)
Set/Get burst munition type
- virtual const [DtEntityType](#)& burstMunitionType() const
- virtual void setBurstWarhead(DtU16 value)
Set/Get burst warhead (optional)
- virtual [DtBoolean](#) getBurstWarhead(DtU16& value) const
- virtual void setBurstFuse(DtU16 value)
Set/Get burst fuse (optional)
- virtual [DtBoolean](#) getBurstFuse(DtU16& value) const
- virtual void setBurstQuantity(DtU16 value)
Set/Get burst quantity (optional)
- virtual [DtBoolean](#) getBurstQuantity(DtU16& value) const
- virtual void setBurstRate(DtU16 value)
Set/Get burst rate (optional)
- virtual [DtBoolean](#) getBurstRate(DtU16& value) const
- virtual void setDetonationResult(DtU8 value)

Set/Get detonation result

virtual DtU8 detonationResult() const

virtual void printData() const

Print the data contained by this PDU

virtual void printHeader() const

Print the header (override to include the new sub-protocol field)

static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtLeDetonationPdu. Caller is responsible for deletion.

static void addCallback(DtExerciseConn* conn, DtLeDetonationPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

static void removeCallback(DtExerciseConn* conn, DtLeDetonationPduCb cb, void* usr)

virtual DtPduKind internalGetPduKind() const

Returns the internal PDU kind

virtual DtNetLeDetonationPdu* netPduStruct() const

Casts the netPdu element to our network struct

DtLeDetonationPdu* self() const

Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type

virtual char* var() const

Returns a pointer to the first byte of variable length data

virtual void setDetonFlagsField(DtU8 value)

Set first detonation flags field

virtual void setDeton2FlagsField(DtU8 value)

Set second detonation flags field

virtual void setDetonFlag(DtLE_DetonFieldFlag flag)

Set first detonation flag

virtual void setDeton2Flag(DtLE_DetonField2Flag flag)

Set second detonation flag

virtual int detonFlagFieldOffset() const

Returns the offset (in bytes) of a field from variable data region. Some fields have subfields which are optional, so there are multiple

virtual int deton2FlagFieldOffset() const

virtual int targetEntityID_Offset() const

virtual int munitionID_Offset1() const

virtual int munitionID_Offset2() const

- virtual int eventID_Offset1() const
- virtual int eventID_Offset2() const
- virtual int locationWorldOffset() const
- virtual int velocityOffset() const
- virtual int munitionOrientationOffset() const
- virtual int burstOffset1() const
- virtual int burstOffset2() const
- virtual int burstOffset3() const
- virtual int locationEntityOffset() const
- virtual int resultOffset() const
- static int theStaticSize
Size of PDU with zero parameter records
- [DtLeEntityID](#) myEntityID
Temporary objects for reference returns
- [DtLeEntityLinearVelocity](#) myVelocity
- [DtEntityType](#) myEntityType

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtLE_DetonFieldFlag

-
-  DtLE_DetonField2
 -  DtLE_DetonLocationEntityCoords
 -  DtLE_DetonBurstQuantityRate
 -  DtLE_DetonBurstWarheadFuse
 -  DtLE_DetonEventSiteApp
 -  DtLE_DetonMunitionID
 -  DtLE_DetonMunitionSiteApp
 -  DtLE_DetonTargetID
-

Documentation

-  DtLE_DetonField2
-  DtLE_DetonLocationEntityCoords
-  DtLE_DetonBurstQuantityRate
-  DtLE_DetonBurstWarheadFuse
-  DtLE_DetonEventSiteApp
-  DtLE_DetonMunitionID
-  DtLE_DetonMunitionSiteApp
-  DtLE_DetonTargetID

In file disenums.h:

enum DtLE_DetonField2Flag

-
-  DtLE_DetonEventNumber
 -  DtLE_DetonMunitionOrientation
-

Documentation

-  DtLE_DetonEventNumber
-  DtLE_DetonMunitionOrientation

In file lelcentity.h:

class DtLeEntityLocationEntityCoordinates

class DtLeEntityLocationEntityCoordinates: DtLeEntityLocationEntityCoordinates is a class which contains the entity location information (in entity coordinates) for the live entity PDUs

Public Methods

	DtLeEntityLocationEntityCoordinates () <i>Constructors</i>
	DtLeEntityLocationEntityCoordinates (const DtLeEntityLocationEntityCoordinates & orig)
 DtLeEntityLocationEntityCoordinates &	operator= (const DtLeEntityLocationEntityCoordinates & orig) <i>Assignment operator</i>
 DtLeEntityLocationEntityCoordinates &	operator= (const DtNetLeEntityLocationEntityCoordinates& orig)
 virtual	~DtLeEntityLocationEntityCoordinates () <i>Destructor</i>
 virtual void	setX (DtU16 value) <i>Set/Get X</i>
 virtual DtU16	x () const
 virtual void	setY (DtU16 value) <i>Set/Get Y</i>
 virtual DtU16	y () const
 virtual void	setZ (DtU16 value) <i>Set/Get Z</i>
 virtual DtU16	z () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetLeEntityLocationEntityCoordinates& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetLeEntityLocationEntityCoordinates& record) const

Protected Fields

 DtU16 myX	<i>Data members:</i>
 DtU16 myY	
 DtU16 myZ	

Documentation

class DtLeEntityLocationEntityCoordinates: DtLeEntityLocationEntityCoordinates is a class which contains the entity location information (in entity coordinates) for the live entity PDUs

DtLeEntityLocationEntityCoordinates()

Constructors

DtLeEntityLocationEntityCoordinates(const [DtLeEntityLocationEntityCoordinates](#)& orig)

[DtLeEntityLocationEntityCoordinates](#)& operator=(const [DtLeEntityLocationEntityCoordinates](#)& orig)

Assignment operator

[DtLeEntityLocationEntityCoordinates](#)& operator=(const DtNetLeEntityLocationEntityCoordinates& orig)

virtual ~DtLeEntityLocationEntityCoordinates()

Destructor

virtual void setX(DtU16 value)

Set/Get X

virtual DtU16 x() const

virtual void setY(DtU16 value)

Set/Get Y

virtual DtU16 y() const

virtual void setZ(DtU16 value)

Set/Get Z

virtual DtU16 z() const

virtual void print() const

Print the information contained in the record

virtual void unloadNetRecord(const DtNetLeEntityLocationEntityCoordinates& record)

Load/Unload data from a packet record

virtual void loadNetRecord(DtNetLeEntityLocationEntityCoordinates& record) const

DtU16 myX

Data members:

DtU16 myY

DtU16 myZ

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file leartpartspdu.h:

class DtLeArticulatedPartsPdu: public DtPdu

Inheritance:

DtLeArticulatedPartsPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtLeArticulatedPartsPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtLeArticulatedPartsPdu (const DtNetLeArticulatedPartsPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtLeArticulatedPartsPdu (const DtLeArticulatedPartsPdu & orig)
virtual	~DtLeArticulatedPartsPdu () <i>destructor</i>
DtLeArticulatedPartsPdu &	operator= (const DtLeArticulatedPartsPdu & orig) <i>Assignment operator</i>
virtual void	setSubProtocol (DtSubProtocol value) <i>Set/Get the sub-protocol number</i>
virtual DtSubProtocol	subProtocol () const
virtual void	setEntityID (const DtLeEntityID & id) <i>Set/Get entity ID</i>
virtual const DtLeEntityID &	entityID () const
virtual DtU8	numRecords () const <i>Get the number of articulation parameter records (this field is automatically set by the PDU when the articulated part list or attached part list is set)</i>
virtual DtBoolean	setArtPartParameters (const DtArtPartList & partList) <i>Set/Get articulated parts parameters</i>
virtual DtBoolean	getArtPartParameters (DtArtPartList & partList) const
virtual DtBoolean	setAttPartParameters (const DtAttachedPartList & partList) <i>Set/Get attached parts parameters</i>
virtual DtBoolean	getAttPartParameters (DtAttachedPartList & partList) const
virtual void	printData () const <i>Print the data contained by this PDU</i>
virtual void	printHeader () const <i>Print the header (override to include the new sub-protocol field)</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtLeArticulatedPartsPdu</i>

- static void **addCallback** ([DtExerciseConn](#)* conn, DtLeArticulatedPartsPduCb cb, void* usr)
Add (register)/Remove (deregister) function to be called when pdu is received
- static void **removeCallback** ([DtExerciseConn](#)* conn, DtLeArticulatedPartsPduCb cb, void* usr)

Public

- The articulated parts and attached parts information are both optional*

Protected Fields

- static int **theStaticSize**
Size of PDU with zero parameter records
- int **myNumArtPartParams**
Number of articulated part parameters for this PDU
- int **myNumAttPartParams**
Number of attached part parameters for this PDU
- [DtLeEntityID](#) **myEntityID**
Temporary objects for reference returns

Protected Methods

- virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind
- virtual DtNetLeArticulatedPartsPdu* **netPduStruct** () const
Casts the netPdu element to our network struct
- [DtLeArticulatedPartsPdu](#)* **self** () const
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
- virtual char* **var** () const
Returns a pointer to the first byte of variable length data
- virtual void **setNumRecords** (DtU8 value)
Set number of records
- virtual int **numRecordsOffset** () const
Returns the offset (in bytes) of a field in the variable data region
- virtual int **artParmOffset** () const
- virtual int **attParmOffset** () const

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)

- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)

- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

`DtLeArticulatedPartsPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

Constructors

`DtLeArticulatedPartsPdu(const DtNetLeArticulatedPartsPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

`DtLeArticulatedPartsPdu(const DtLeArticulatedPartsPdu& orig)`

`virtual ~DtLeArticulatedPartsPdu()`

destructor

`DtLeArticulatedPartsPdu& operator=(const DtLeArticulatedPartsPdu& orig)`

Assignment operator

`virtual void setSubProtocol(DtSubProtocol value)`

Set/Get the sub-protocol number

`virtual DtSubProtocol subProtocol() const`

`virtual void setEntityID(const DtLeEntityID& id)`

Set/Get entity ID

`virtual const DtLeEntityID& entityID() const`

`virtual DtU8 numRecords() const`

Get the number of articulation parameter records (this field is automatically set by the PDU when the articulated part list or attached part list is set)

The articulated parts and attached parts information are both optional. Either or both can be set.

`virtual DtBoolean setArtPartParameters(const DtArtPartList& partList)`

Set/Get articulated parts parameters. Parameters are passed in the form of an articulated parts list. Returns `DtFalse` if the operation could not be performed.

`virtual DtBoolean getArtPartParameters(DtArtPartList& partList) const`

`virtual DtBoolean setAttPartParameters(const DtAttachedPartList& partList)`

Set/Get attached parts parameters. Parameters are passed in the form of an attached parts list. Returns `DtFalse` if the operation could not be performed.

`virtual DtBoolean getAttPartParameters(DtAttachedPartList& partList) const`

`virtual void printData() const`

Print the data contained by this PDU

`virtual void printHeader() const`

Print the header (override to include the new sub-protocol field)

`static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

Create a `DtLeArticulatedPartsPdu`. Caller is responsible for deletion.

-  `static void addCallback(DtExerciseConn* conn, DtLeArticulatedPartsPduCb cb, void* usr)`
 Add (register)/Remove (deregister) function to be called when pdu is received
-  `static void removeCallback(DtExerciseConn* conn, DtLeArticulatedPartsPduCb cb, void* usr)`
-  `virtual DtPduKind internalGetPduKind() const`
 Returns the internal PDU kind
-  `virtual DtNetLeArticulatedPartsPdu* netPduStruct() const`
 Casts the netPdu element to our network struct
-  `DtLeArticulatedPartsPdu* self() const`
 Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
-  `virtual char* var() const`
 Returns a pointer to the first byte of variable length data
-  `virtual void setNumRecords(DtU8 value)`
 Set number of records
-  `virtual int numRecordsOffset() const`
 Returns the offset (in bytes) of a field in the variable data region
-  `virtual int artParmOffset() const`
-  `virtual int attParmOffset() const`
-  `static int theStaticSize`
 Size of PDU with zero parameter records
-  `int myNumArtPartParams`
 Number of articulated part parameters for this PDU
-  `int myNumAttPartParams`
 Number of attached part parameters for this PDU
-  `DtLeEntityID myEntityID`
 Temporary objects for reference returns

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file artpartlist.h:

class DtArtPartList

class DtArtPartList: Instances of DtArtPartList are used to manage articulated parts

Public Methods

	DtArtPartList () <i>default constructor</i>
	DtArtPartList (const DtArtPartSpec* specList) <i>constructor</i>
virtual	~DtArtPartList () <i>destructor</i>
	DtArtPartList (const DtArtPartList & orig) <i>copy constructor</i>
DtArtPartList &	operator= (const DtArtPartList & orig) <i>assignment operator</i>
void	init (const DtArtPartSpec* partSpecs) <i>The init functions don't set values for the parameters, they merely set up the part and parameter types</i>
virtual void	init (int numParams, const DtNetArtParamRecord* netParams) <i>The init functions don't set values for the parameters, they merely set up the part and parameter types</i>
virtual void	setValsFromNet (int numParams, const DtNetArtParamRecord* netParams) <i>sets values from net representation, but does not change the set of valid parts and parameters</i>
virtual void	setFromNet (int numParams, const DtNetArtParamRecord* netParams) <i>initializes the part types and parameters, and sets values as well</i>
virtual void	fillNetParams (DtNetArtParamRecord* netParams) const <i>fills in netParams based on current values</i>
void	clear () <i>removes all parts and parameters from list Non-virtual since called from ctor/dtor</i>
virtual DtBoolean	setArtParamVal (int partType, int paramType, DtFloat32 val) <i>Set/Get the specified part type's specified parameter type value</i>
virtual DtBoolean	getArtParamVal (int partType, int paramType, DtFloat32& val) const
virtual int	numArtParts () const <i>Get the number of articulated parts</i>
virtual int	numArtParams () const <i>Get the number of articulated parameters</i>
virtual DtArtPart *	artPartFromPartNum (int partNum) const <i>Get the articulated part from the part number</i>
virtual DtArtPart *	artPartFromPartType (int partType) const <i>Get the first articulated part with the part type</i>
virtual int	partNumFromPartType (int partType) const <i>returns 0 if there is no such part, otherwise the part number</i>

- virtual const DtArtPartSpec* **artPartSpecs** () const
return the list of DtArtPartSpecs describing the parts in the list
- virtual void **print** () const
Print the articulated part list
- static void **fillSpecList** (int numParams, const DtNetArtParamRecord* netParams, DtArtPartSpec* specList)
- static [DtBoolean](#) **artNotAttached** (const DtNetArtParamRecord* param)
Returns true if param is an articulated parameter record, false if its an attached part record
- inline int **numArtParams** () const
- inline int **numArtParts** () const
- inline [DtArtPartList](#)* **self** () const

Public

- Fill the articulated part spec list from the net parameter record list*

Protected Fields

- [DtArtPart](#)** **myPartList**
- int **myNumArtParts**
- int **myNumArtParams**
- DtArtPartSpec* **mySpecList**

Protected Methods

- virtual [DtBoolean](#) **partNumInBounds** (int partNum) const
Returns whether the part number is within bounds
- [DtArtPartList](#)* **self** () const
Cast off const-ness for trivial internal change

Documentation

class DtArtPartList:

Instances of DtArtPartList are used to manage articulated parts. Articulated part spec lists are mainly used to generate these lists.

- DtArtPartList** ()
default constructor
- DtArtPartList** (const DtArtPartSpec* specList)
constructor
- virtual **~DtArtPartList** ()

destructor

`DtArtPartList(const DtArtPartList& orig)`

copy constructor

`DtArtPartList& operator=(const DtArtPartList& orig)`

assignment operator

`void init(const DtArtPartSpec* partSpecs)`

The init functions don't set values for the parameters, they merely set up the part and parameter types. Non-virtual since called from ctor.

`virtual void init(int numParams, const DtNetArtParamRecord* netParams)`

The init functions don't set values for the parameters, they merely set up the part and parameter types. numParams is the total number of records in netParams, which might include attached parameter records that will be ignored.

`virtual void setValsFromNet(int numParams, const DtNetArtParamRecord* netParams)`

sets values from net representation, but does not change the set of valid parts and parameters. numParams is the total number of records in netParams, which might include attached parameter records that will be ignored.

`virtual void setFromNet(int numParams, const DtNetArtParamRecord* netParams)`

initializes the part types and parameters, and sets values as well. numParams is the total number of records in netParams, which might include attached parameter records that will be ignored.

`virtual void fillNetParams(DtNetArtParamRecord* netParams) const`

fills in netParams based on current values

`void clear()`

removes all parts and parameters from list Non-virtual since called from ctor/dtor

`virtual DtBoolean setArtParamVal(int partType, int paramType, DtFloat32 val)`

Set/Get the specified part type's specified parameter type value. Returns DtFalse if the part or parameter are not found, otherwise returns DtTrue.

`virtual DtBoolean getArtParamVal(int partType, int paramType, DtFloat32& val) const`

`virtual int numArtParts() const`

Get the number of articulated parts

`virtual int numArtParams() const`

Get the number of articulated parameters

`virtual DtArtPart* artPartFromPartNum(int partNum) const`

Get the articulated part from the part number

`virtual DtArtPart* artPartFromPartType(int partType) const`

Get the first articulated part with the part type

`virtual int partNumFromPartType(int partType) const`

returns 0 if there is no such part, otherwise the part number

`virtual const DtArtPartSpec* artPartSpecs() const`

return the list of DtArtPartSpecs describing the parts in the list

virtual void print() const

Print the articulated part list

Fill the articulated part spec list from the net parameter record list. numParams is the total number of records in netParams, which might include attached parameter records that will be ignored.

static void fillSpecList(int numParams, const DtNetArtParamRecord* netParams, DtArtPartSpec* specList)

static DtBoolean artNotAttached(const DtNetArtParamRecord* param)

Returns true if param is an articulated parameter record, false if its an attached part record

virtual DtBoolean partNumInBounds(int partNum) const

Returns whether the part number is within bounds

DtArtPartList* self() const

Cast off const-ness for trivial internal change. Non-virtual since can't override on return type.

DtArtPart** myPartList

int myNumArtParts

int myNumArtParams

DtArtPartSpec* mySpecList

inline int numArtParams() const

inline int numArtParts() const

inline DtArtPartList* self() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file artpart.h:

class DtArtPart

One Articulated Part

Public Methods

virtual DtBoolean	setVal (int paramType, DtFloat32 value) <i>Set/Get the value for the specified parameter type</i>
virtual DtBoolean	getVal (int paramType, DtFloat32& value) const
virtual DtFloat32	val (int paramType) const <i>Get the value for the specified parameter type</i>
virtual int	partType () const <i>Get the part type</i>
virtual int	partNum () const <i>Get the part number</i>
virtual int	partAttachedTo () const <i>Get the part this part is attached to</i>
virtual int	numParams () const <i>Get the number of parameters</i>
virtual int	firstValidParamType () const <i>Get the first valid (added) parameter type</i>
virtual int	nextValidParamType (int paramType) const <i>Get the next valid parameter type</i>
virtual int	paramNum (int paramType) const <i>Returns the index of the parameter in a multiple part list as is found in HLA messages and DIS PDUs</i>
virtual const DtTaitBryan &	angles () const <i>Get the euler angles</i>
virtual void	getAngles (DtTaitBryan & euler) const
virtual DtConstVector	translations () const <i>Get the translations</i>
virtual void	getTranslations (DtVectorRef vec) const
virtual void	setApproximating (DtBoolean onOff) <i>Set/Get whether or not to do approximating</i>
virtual DtBoolean	approximating () const
virtual void	print () <i>Print the part data</i>
static int	artParamsRate (int paramType) <i>Get whether the parameter type is actually a rate</i>
static int	computePartType (int partPlusParam) <i>Get the part type from the part and parameter value</i>
static int	computeParameter (int partPlusParam) <i>Get the parameter type from the part and parameter value</i>

	DtArtPart () <i>default constructor</i>
inline DtBoolean	approximating () const
inline DtFloat32	lastSetVal (int paramType) const
inline int	numParams () const
inline int	partAttachedTo () const
inline int	partNum () const
inline int	partType () const
inline DtArtPart*	self () const
inline void	setApproximating (DtBoolean onOff)
inline void	setPartNum (int num)
inline DtTime	timeSet (int paramType) const

Protected Fields

int	myPartType
int	myPartNum
int	myPartAttachedTo
int	myNumParams
DtInternalArtParam**	myVals <i>myVals is an array of DtMAX_PARAMS_PER_PART + 1 items which are themselves pointers</i>
int	myFirstParam
int	myLastParam
DtBoolean	myApproxFlag
DtTaitBryan	myAngles
DtVector	myVector

Protected Methods

	DtArtPart (const DtArtPartSpec& spec) <i>constructor</i>
virtual	~DtArtPart () <i>destructor</i>
	DtArtPart (const DtArtPart & orig) <i>copy constructor</i>
DtArtPart &	operator= (const DtArtPart & orig) <i>assignment operator</i>
virtual void	setParamNum (int paramType, int num) <i>Set the parameter number of the specified parameter type</i>
virtual void	setPartNum (int num) <i>Set the part number</i>

- virtual void **addParam** (int paramType)
Add the specified parameter type to the part
- virtual int **findNextValidParamType** (int paramType) const
Find the next valid parameter type
- virtual DtFloat32 **approxVal** (int paramType) const
Get the approximated value
- virtual DtFloat32 **lastSetVal** (int paramType) const
Get the last set (absolute) value
- virtual DtTime **timeSet** (int paramType) const
Get the time at which the value was last set
- [DtArtPart](#)* **self** () const
Cast off const-ness for trivial internal change

Protected

- Constructors are protected since this class is managed by DtArtPartList
-

Documentation

One Articulated Part

- virtual [DtBoolean](#) setVal(int paramType, DtFloat32 value)
Set/Get the value for the specified parameter type. Returns DtFalse if the parameter type has not been added, returns DtTrue otherwise.
- virtual [DtBoolean](#) getVal(int paramType, DtFloat32& value) const
- virtual DtFloat32 val(int paramType) const
Get the value for the specified parameter type. Note this does not check existence of the parameter type and should only be called when you know that it does exist (as from nextValidParamType()).
- virtual int partType() const
Get the part type
- virtual int partNum() const
Get the part number
- virtual int partAttachedTo() const
Get the part this part is attached to
- virtual int numParams() const
Get the number of parameters
- virtual int firstValidParamType() const
Get the first valid (added) parameter type
- virtual int nextValidParamType(int paramType) const
Get the next valid parameter type
- virtual int paramNum(int paramType) const

Returns the index of the parameter in a multiple part list as is found in HLA messages and DIS PDUs

virtual const [DtTaitBryan](#)& angles() const

Get the euler angles. Euler is filled with the angles phi, theta and psi of the part with respect to its neutral orientation.

virtual void getAngles([DtTaitBryan](#)& euler) const

virtual DtConstVector translations() const

Get the translations. Vec is filled in with a vector representing the current translation, in x, y and z of the part from its neutral position.

virtual void getTranslations(DtVectorRef vec) const

virtual void setApproximating([DtBoolean](#) onOff)

Set/Get whether or not to do approximating

virtual [DtBoolean](#) approximating() const

virtual void print()

Print the part data

static int artParamIsRate(int paramType)

Get whether the parameter type is actually a rate

static int computePartType(int partPlusParam)

Get the part type from the part and parameter value

static int computeParameter(int partPlusParam)

Get the parameter type from the part and parameter value

Constructors are protected since this class is managed by DtArtPartList

DtArtPart()

default constructor

DtArtPart(const DtArtPartSpec& spec)

constructor

virtual ~DtArtPart()

destructor

DtArtPart(const [DtArtPart](#)& orig)

copy constructor

[DtArtPart](#)& operator=(const [DtArtPart](#)& orig)

assignment operator

virtual void setParamNum(int paramType, int num)

Set the parameter number of the specified parameter type

virtual void setPartNum(int num)

Set the part number

virtual void addParam(int paramType)

Add the specified parameter type to the part. Note that this assumes that parameters are always added in order from lowest to highest.

virtual int findNextValidParamType(int paramType) const

Find the next valid parameter type

virtual DtFloat32 approxVal(int paramType) const

Get the approximated value

virtual DtFloat32 lastSetVal(int paramType) const

Get the last set (absolute) value

virtual DtTime timeSet(int paramType) const

Get the time at which the value was last set

[DtArtPart](#)* self() const

Cast off const-ness for trivial internal change. Non-virtual since can't override on return type.

int myPartType

int myPartNum

int myPartAttachedTo

int myNumParams

DtInternalArtParam** myVals

myVals is an array of DtMAX_PARAMS_PER_PART + 1 items which are themselves pointers

int myFirstParam

int myLastParam

[DtBoolean](#) myApproxFlag

[DtTaitBryan](#) myAngles

[DtVector](#) myVector

inline [DtBoolean](#) approximating() const

inline DtFloat32 lastSetVal(int paramType) const

inline int numParams() const

inline int partAttachedTo() const

inline int partNum() const

inline int partType() const

inline [DtArtPart](#)* self() const

inline void setApproximating([DtBoolean](#) onOff)

inline void setPartNum(int num)

inline DtTime timeSet(int paramType) const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file attpartlist.h:

class DtAttachedPartList

class DtAttachedPartList: Instances of DtAttachedPartList are lists of attached parts

Public Methods

•	DtAttachedPartList () <i>default constructor</i>
•	DtAttachedPartList (const DtAttachedPartList & orig) <i>copy constructor</i>
• DtAttachedPartList &	operator= (const DtAttachedPartList & orig) <i>assignment operator</i>
• virtual	~DtAttachedPartList () <i>destructor</i>
• virtual DtBoolean	setNumAttachedParts (int num) <i>Set/Get the number of attached parts</i>
• virtual int	numAttachedParts () const
• virtual const int*	stationList () const <i>Returns a zero-termination list of station numbers that are valid in this attached part list</i>
• virtual void	init (const int* stations) <i>Stations is a zero-terminated array of stations</i>
• virtual DtBoolean	setStationNumber (int partIndex, int station) <i>Set/Get the station number for a part</i>
• virtual DtBoolean	getStationNumber (int partIndex, int& station) const
• virtual int	stationNumber (int partIndex) const <i>Returns the station number for a part</i>
• virtual DtBoolean	setEntityType (int partIndex, const DtEntityType & type) <i>Set/Get the entity type for a part</i>
• virtual DtBoolean	getEntityType (int partIndex, DtEntityType & type) const
• virtual const DtEntityType &	entityType (int partIndex) const <i>Returns the entityType for a part, by index</i>
• virtual DtAttachedPartStatus	setEntityTypeAtStation (int station, const DtEntityType & type) <i>Set/Get the entity type for a part</i>
• virtual DtAttachedPartStatus	getEntityTypeAtStation (int station, DtEntityType & type) const
• virtual const DtEntityType &	entityTypeAtStation (int station) const <i>Returns the entityType at a particular station</i>
• virtual void	unloadNetList (int numParams, const DtNetAttachedPart* partList) <i>After this call, this attached part list will reflect the information contained in netParams</i>
• virtual void	loadNetList (DtNetAttachedPart* partList) const <i>Load a list from network data representation</i>
• virtual int	netRecordLength () const <i>Get the network data length for this list</i>

- [DtBoolean](#) **equivalent** (const [DtAttachedPartList](#)& other) const
Returns true if we contain the exact same information as other
- virtual void **print ()** const
Print the information contained in the record
- inline const int* **stationList ()** const

Protected Fields

- int **myNumParts**
 - int* **myStationList**
zero terminated list that should always be of length myNumParts + 1
 - [DtEntityType](#)* **myTypeList**
NULL if myNumArtParts = 0, otherwise has length = myNumParts
-

Documentation

class DtAttachedPartList: Instances of DtAttachedPartList are lists of attached parts

- **DtAttachedPartList()**
default constructor
- **DtAttachedPartList(const [DtAttachedPartList](#)& orig)**
copy constructor
- **[DtAttachedPartList](#)& operator=(const [DtAttachedPartList](#)& orig)**
assignment operator
- **virtual ~DtAttachedPartList()**
destructor
- **virtual [DtBoolean](#) setNumAttachedParts(int num)**
Set/Get the number of attached parts. Set may return DtFalse if the number of requested parts is invalid.
- **virtual int numAttachedParts() const**
- **virtual const int* stationList() const**
Returns a zero-termination list of station numbers that are valid in this attached part list
- **virtual void init(const int* stations)**
Stations is a zero-terminated array of stations. Init initializes the attPartList with the stations in the array.
- **virtual [DtBoolean](#) setStationNumber(int partIndex, int station)**
Set/Get the station number for a part. Returns DtFalse if the index is out of range.
- **virtual [DtBoolean](#) getStationNumber(int partIndex, int& station) const**
- **virtual int stationNumber(int partIndex) const**
Returns the station number for a part. This version of the function requires that partIndex be valid (between 0 and numAttachedParts()).

- virtual [DtBoolean](#) setEntityType(int partIndex, const [DtEntityType&](#) type)
 Set/Get the entity type for a part. Returns DtFalse if the index is out of range.
- virtual [DtBoolean](#) getEntityType(int partIndex, [DtEntityType&](#) type) const
- virtual const [DtEntityType&](#) entityType(int partIndex) const
 Returns the entityType for a part, by index. This version of the function requires that partIndex be valid (between 0 and numAttachedParts()).
- virtual [DtAttachedPartStatus](#) setEntityTypeAtStation(int station, const [DtEntityType&](#) type)
 Set/Get the entity type for a part. Returns DtFalse if the index is out of range.
- virtual [DtAttachedPartStatus](#) getEntityTypeAtStation(int station, [DtEntityType&](#) type) const
- virtual const [DtEntityType&](#) entityTypeAtStation(int station) const
 Returns the entityType at a particular station. If the list does not contain an entry for the station, DtEntityType::nullEntityType is returned;
- virtual void unloadNetList(int numParams, const DtNetAttachedPart* partList)
 After this call, this attached part list will reflect the information contained in netParams. Both the set of valid stations, and the entity types attached to those stations might be changed. numParams is the total number of records in netParams, which might include articulated parameter records that will be ignored.
- virtual void loadNetList(DtNetAttachedPart* partList) const
 Load a list from network data representation. numAttachedParts() network part records will be written/read.
- virtual int netRecordLength() const
 Get the network data length for this list
- [DtBoolean](#) equivalent(const [DtAttachedPartList&](#) other) const
 Returns true if we contain the exact same information as other
- virtual void print() const
 Print the information contained in the record
- int myNumParts
- int* myStationList
 zero terminated list that should always be of length myNumParts + 1
- [DtEntityType*](#) myTypeList
 NULL if myNumArtParts = 0, otherwise has length = myNumParts
- inline const int* stationList() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file attpartlist.h:

enum DtAttachedPartStatus

-  DtAttachedPartSuccess
 -  DtNoSuchAttachStation
 -  DtPartNotAttached
 -  DtPartAttached
-

Documentation

-  DtAttachedPartSuccess
-  DtNoSuchAttachStation
-  DtPartNotAttached
-  DtPartAttached

In file leappearpdu.h:

class DtLeAppearancePdu: public DtPdu

Inheritance:

DtLeAppearancePdu - DtPdu - DtBaseMsg

Public Methods

	DtLeAppearancePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtLeAppearancePdu (const DtNetLeAppearancePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtLeAppearancePdu (const DtLeAppearancePdu & orig)
virtual	~DtLeAppearancePdu () <i>destructor</i>
DtLeAppearancePdu &	operator= (const DtLeAppearancePdu & orig) <i>Assignment operator</i>
virtual void	setSubProtocol (DtSubProtocol value) <i>Set/Get the sub-protocol number</i>
virtual DtSubProtocol	subProtocol () const
virtual void	setEntityID (const DtLeEntityID & id) <i>Set/Get entity ID</i>
virtual const DtLeEntityID &	entityID () const
virtual DtU8	appearanceField1 () const <i>Get the first appearance flag field (this field is automatically maintained by the PDU as other PDU fields are set)</i>
virtual DtBoolean	appearanceField1Flag (DtLE_AppearanceField1Flag flag) const <i>Get the value for a specific appearance flag from the first appearance field</i>
virtual DtBoolean	getAppearanceField2 (DtU8& value) const <i>Get the second appearance flag field (this field is automatically maintained by the PDU as other PDU fields are set)</i>
virtual DtBoolean	getAppearanceField2Flag (DtLE_AppearanceField2Flag flag, DtBoolean & value) const <i>Get the value for a specific appearance flag from the second appearance field</i>
virtual void	setForceID (DtForceType force) <i>Set/Get the force ID</i>
virtual DtBoolean	getForceID (DtForceType & force) const
virtual void	setEntityType (const DtEntityType & type) <i>Set/Get entity type</i>

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual void

static [DtPdu](#)*

static void

static void

getEntityType ([DtEntityType](#)& type) const

setAlternativeEntityType (const [DtEntityType](#)& type)

Set/Get alternative entity type

getAlternativeEntityType ([DtEntityType](#)& type) const

setMarkingCharacterSet ([DtCharacterSet](#) value)

Set/Get marking character set

getMarkingCharacterSet ([DtCharacterSet](#)& value) const

setMarkingText (DtU8* dataPtr)

Set/Get marking text (copies 11 bytes)

getMarkingText (DtU8* dataPtr) const

setCapabilities (DtU32 value)

Set/Get capabilities (as a 32-bit integer)

getCapabilities (DtU32& value) const

setCapability ([DtCapability](#) flag, [DtBoolean](#) value)

Set/Get a particular capability (as a Boolean)

getCapability ([DtCapability](#) flag, [DtBoolean](#)& value) const

setVisualAppearance (DtU32 value)

Set/Get visual appearance (as a 32 bit integer)

getVisualAppearance (DtU32& value) const

setVisualAppearanceAttribute ([DtAppearAttribute](#) attr, [DtAppearAttrVal](#) val)

Set/Get appearance by attribute

getVisualAppearanceAttribute ([DtAppearAttribute](#) attr, [DtAppearAttrVal](#)& val)

setIR_Appearance (DtU32 value)

Set/Get IR appearance (as a 32 bit integer)

getIR_Appearance (DtU32& value) const

setEM_Appearance (DtU32 value)

Set/Get EM appearance (as a 32 bit integer)

getEM_Appearance (DtU32& value) const

setAudioAppearance (DtU32 value)

Set/Get audio appearance (as a 32 bit integer)

getAudioAppearance (DtU32& value) const

printData () const

Print the data contained by this PDU

printHeader () const

Print the header (override to include the new sub-protocol field)

create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtLeAppearancePdu

addCallback ([DtExerciseConn](#)* conn, DtLeAppearancePduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

removeCallback ([DtExerciseConn](#)* conn, DtLeAppearancePduCb cb, void* usr)

Public



The following accessor functions refer to records which are optional

Protected Fields



static int **theStaticSize**
Size of PDU with zero parameter records



[DtLeEntityID](#) **myEntityID**
Temporary objects for reference returns

Protected Methods



virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind



virtual DtNetLeAppearancePdu* **netPduStruct** () const
Casts the netPdu element to our network struct



[DtLeAppearancePdu](#)* **self** () const
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type



virtual char* **var** () const
Returns a pointer to the first byte of variable length data



virtual void **setAppearanceField1** (DtU8 value)
Set appearance field 1



virtual void **setAppearanceField1Flag** ([DtLE_AppearanceField1Flag](#) flag)
Set a flag of appearance field 1



virtual void **clearField1Flags** ([DtLE_AppearanceField1Flag](#) flag)
Clear a flag and all following flags in appearance field 1



virtual void **setAppearanceField2** (DtU8 value)
Set appearance field 2



virtual void **setAppearanceField2Flag** ([DtLE_AppearanceField2Flag](#) flag)
Set a flag of appearance field 2



virtual int **appearanceField1Offset** () const
Returns the offset (in bytes) of a field in the variable data region



virtual int **appearanceField2Offset** () const



virtual int **forceID_Offset** () const



virtual int **entityTypeOffset** () const



virtual int **alternativeEntityTypeOffset** () const



virtual int **entityMarkingOffset** () const



virtual int **capabilitiesOffset** () const



virtual int **visualAppearanceOffset** () const



virtual int **IR_AppearanceOffset** () const



virtual int **EM_AppearanceOffset** () const



virtual int **audioAppearanceOffset** () const

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtLeAppearancePdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Constructors

- DtLeAppearancePdu**(const DtNetLeAppearancePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

- DtLeAppearancePdu**(const [DtLeAppearancePdu](#)& orig)

- virtual ~**DtLeAppearancePdu**()

destructor

- [DtLeAppearancePdu](#)& operator=(const [DtLeAppearancePdu](#)& orig)

Assignment operator

- virtual void **setSubProtocol**([DtSubProtocol](#) value)

Set/Get the sub-protocol number

- virtual [DtSubProtocol](#) **subProtocol**() const

- virtual void **setEntityID**(const [DtLeEntityID](#)& id)

Set/Get entity ID

- virtual const [DtLeEntityID](#)& **entityID**() const

- virtual DtU8 **appearanceField1**() const

Get the first appearance flag field (this field is automatically maintained by the PDU as other PDU fields are set)

- virtual [DtBoolean](#) **appearanceField1Flag**([DtLE_AppearanceField1Flag](#) flag) const

Get the value for a specific appearance flag from the first appearance field



The following accessor functions refer to records which are optional. The set methods create the record in the PDU if it does not already exist. The get methods return DtFalse if the record does not exist.

- virtual [DtBoolean](#) **getAppearanceField2**(DtU8& value) const

Get the second appearance flag field (this field is automatically maintained by the PDU as other PDU fields are set)

- virtual [DtBoolean](#) **getAppearanceField2Flag**([DtLE_AppearanceField2Flag](#) flag, [DtBoolean](#)& value) const

Get the value for a specific appearance flag from the second appearance field

- virtual void setForceID([DtForceType](#) force)
Set/Get the force ID
- virtual [DtBoolean](#) getForceID([DtForceType](#)& force) const
- virtual void setEntityType(const [DtEntityType](#)& type)
Set/Get entity type
- virtual [DtBoolean](#) getEntityType([DtEntityType](#)& type) const
- virtual void setAlternativeEntityType(const [DtEntityType](#)& type)
Set/Get alternative entity type
- virtual [DtBoolean](#) getAlternativeEntityType([DtEntityType](#)& type) const
- virtual void setMarkingCharacterSet([DtCharacterSet](#) value)
Set/Get marking character set
- virtual [DtBoolean](#) getMarkingCharacterSet([DtCharacterSet](#)& value) const
- virtual void setMarkingText(DtU8* dataPtr)
Set/Get marking text (copies 11 bytes)
- virtual [DtBoolean](#) getMarkingText(DtU8* dataPtr) const
- virtual void setCapabilities(DtU32 value)
Set/Get capabilities (as a 32-bit integer)
- virtual [DtBoolean](#) getCapabilities(DtU32& value) const
- virtual void setCapability([DtCapability](#) flag, [DtBoolean](#) value)
Set/Get a particular capability (as a Boolean)
- virtual [DtBoolean](#) getCapability([DtCapability](#) flag, [DtBoolean](#)& value) const
- virtual void setVisualAppearance(DtU32 value)
Set/Get visual appearance (as a 32 bit integer)
- virtual [DtBoolean](#) getVisualAppearance(DtU32& value) const
- virtual void setVisualAppearanceAttribute([DtAppearAttribute](#) attr, [DtAppearAttrVal](#) val)
Set/Get appearance by attribute
- virtual [DtBoolean](#) getVisualAppearanceAttribute([DtAppearAttribute](#) attr, [DtAppearAttrVal](#)& val)
- virtual void setIR_Appearance(DtU32 value)
Set/Get IR appearance (as a 32 bit integer)
- virtual [DtBoolean](#) getIR_Appearance(DtU32& value) const
- virtual void setEM_Appearance(DtU32 value)
Set/Get EM appearance (as a 32 bit integer)
- virtual [DtBoolean](#) getEM_Appearance(DtU32& value) const

- virtual void setAudioAppearance(DtU32 value)
 Set/Get audio appearance (as a 32 bit integer)
- virtual [DtBoolean](#) getAudioAppearance(DtU32& value) const
- virtual void printData() const
 Print the data contained by this PDU
- virtual void printHeader() const
 Print the header (override to include the new sub-protocol field)
- static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 Create a DtLeAppearancePdu. Caller is responsible for deletion.
- static void addCallback([DtExerciseConn](#)* conn, DtLeAppearancePduCb cb, void* usr)
 Add (register)/Remove (deregister) function to be called when pdu is received
- static void removeCallback([DtExerciseConn](#)* conn, DtLeAppearancePduCb cb, void* usr)
- virtual [DtPduKind](#) internalGetPduKind() const
 Returns the internal PDU kind
- virtual DtNetLeAppearancePdu* netPduStruct() const
 Casts the netPdu element to our network struct
- [DtLeAppearancePdu](#)* self() const
 Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
- virtual char* var() const
 Returns a pointer to the first byte of variable length data
- virtual void setAppearanceField1(DtU8 value)
 Set appearance field 1
- virtual void setAppearanceField1Flag([DtLE_AppearanceField1Flag](#) flag)
 Set a flag of appearance field 1
- virtual void clearField1Flags([DtLE_AppearanceField1Flag](#) flag)
 Clear a flag and all following flags in appearance field 1
- virtual void setAppearanceField2(DtU8 value)
 Set appearance field 2
- virtual void setAppearanceField2Flag([DtLE_AppearanceField2Flag](#) flag)
 Set a flag of appearance field 2
- virtual int appearanceField1Offset() const
 Returns the offset (in bytes) of a field in the variable data region
- virtual int appearanceField2Offset() const
- virtual int forceID_Offset() const

- virtual int entityTypeOffset() const
 - virtual int alternativeEntityTypeOffset() const
 - virtual int entityMarkingOffset() const
 - virtual int capabilitiesOffset() const
 - virtual int visualAppearanceOffset() const
 - virtual int IR_AppearanceOffset() const
 - virtual int EM_AppearanceOffset() const
 - virtual int audioAppearanceOffset() const
 - static int theStaticSize
Size of PDU with zero parameter records
 - [DtLeEntityID](#) myEntityID
Temporary objects for reference returns
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file a/senums.h:

enum DtLE_AppearanceField1Flag

-  DtLE_AppearanceFlagField2
 -  DtLE_AppearanceIR
 -  DtLE_AppearanceVisual
 -  DtLE_AppearanceCapabilities
 -  DtLE_AppearanceMarking
 -  DtLE_AppearanceAlternativeType
 -  DtLE_AppearanceType
 -  DtLE_AppearanceForceID
-

Documentation

-  DtLE_AppearanceFlagField2
-  DtLE_AppearanceIR
-  DtLE_AppearanceVisual
-  DtLE_AppearanceCapabilities
-  DtLE_AppearanceMarking
-  DtLE_AppearanceAlternativeType
-  DtLE_AppearanceType
-  DtLE_AppearanceForceID

In file disenums.h:

enum DtLE_AppearanceField2Flag

 DtLE_AppearanceAudio

 DtLE_AppearanceEM

Documentation

 DtLE_AppearanceAudio

 DtLE_AppearanceEM

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtCharacterSet

 DtUnusedCharacterSet

 DtASCII_CharacterSet

Documentation

 DtUnusedCharacterSet

 DtASCII_CharacterSet

In file disenums.h:

enum DtCapability

-  DtAmmunitionSupplyCap
 -  DtFuelSupplyCap
 -  DtRecoveryCap
 -  DtRepairCap
-

Documentation

-  DtAmmunitionSupplyCap
-  DtFuelSupplyCap
-  DtRecoveryCap
-  DtRepairCap

In file laserpdu.h:

class DtLaserPdu: public DtPdu

Inheritance:

DtLaserPdu - DtPdu - DtBaseMsg

Public Methods

	DtLaserPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtLaserPdu (const DtNetLaserPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static void	addCallback (DtExerciseConn *conn, DtLaserPduCb cb, void *usr)
 static void	removeCallback (DtExerciseConn *conn, DtLaserPduCb cb, void *usr)
 const DtEntityIdentifier &	lasingEntityId () const
 DtLaserCodeName	codeName () const
 const DtEntityIdentifier &	lasedEntityId () const
 DtLaserCode	laserCode () const
 float	power () const
 float	wavelength () const
 DtConstVector	lasedSpotRelative () const
 DtConstVector	lasedSpotWorld () const
 void	setLasingEntityId (const DtEntityIdentifier &id)
 void	setCodeName (DtLaserCodeName name)
 void	setLasedEntityId (const DtEntityIdentifier &id)
 void	setLaserCode (DtLaserCode code)
 void	setPower (float power)
 void	setWavelength (float wavelength)
 void	setLasedSpotRelative (DtConstVector loc)
 void	setLasedSpotWorld (DtConstVector loc)
 void	printData () const

Protected Fields

 [DtEntityIdentifier](#) **theld**

 [DtVector](#) **loc**

Protected Methods

 [DtPduKind](#) **internalGetPduKind** () const

Inherited from [DtPdu](#):

Public Methods

 [DtPdu](#)& **operator=**(const [DtPdu](#) &pdu)

 virtual void* **packet**() const

 virtual int **status**() const

 virtual void **setVersion**(int vers)

 virtual int **protocolVersion**() const

 virtual void **setExerciseld**(int id)

 virtual int **exerciseld**() const

 virtual [DtPduKind](#) **kind**() const

 virtual [DtProtocolFamily](#) **protocolFamily**() const

 virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

 virtual DtTime **timeStamp**() const

 virtual [DtTimeStampType](#) **timeStampType**() const

 virtual DtTime **guessTimeValid**(DtTime referenceTime) const

 virtual void **print**() const

 virtual void **printHeader**() const

 inline int **exerciseld**() const

 inline [DtPduKind](#) **kind**() const

 inline DtNetPduHeader* **netHeader**() const

 inline void* **packet**() const

- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtLaserPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtLaserPdu(const DtNetLaserPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void addCallback([DtExerciseConn](#) *conn, DtLaserPduCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtLaserPduCb cb, void *usr)
- const [DtEntityIdentifier](#)& lasingEntityId() const
- DtLaserCodeName codeName() const
- const [DtEntityIdentifier](#)& lasedEntityId() const
- DtLaserCode laserCode() const
- float power() const
- float wavelength() const

- `DtConstVector lasedSpotRelative() const`
- `DtConstVector lasedSpotWorld() const`
- `void setLasingEntityId(const DtEntityIdentifier &id)`
- `void setCodeName(DtLaserCodeName name)`
- `void setLasedEntityId(const DtEntityIdentifier &id)`
- `void setLaserCode(DtLaserCode code)`
- `void setPower(float power)`
- `void setWavelength(float wavelength)`
- `void setLasedSpotRelative(DtConstVector loc)`
- `void setLasedSpotWorld(DtConstVector loc)`
- `void printData() const`
- `DtPduKind internalGetPduKind() const`
- `DtEntityIdentifier theId`
- `DtVector loc`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file ispartofpdu.h:

class DtlsPartOfPdu: public DtPdu

Inheritance:

DtlsPartOfPdu - DtPdu - DtBaseMsg

Public Methods

	DtlsPartOfPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtlsPartOfPdu (const DtNetIsPartOfPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtlsPartOfPdu (const <u>DtlsPartOfPdu</u> & orig)
 virtual	~DtlsPartOfPdu () <i>destructor</i>
 <u>DtlsPartOfPdu</u> &	operator= (const <u>DtlsPartOfPdu</u> & orig) <i>Assignment operator</i>
 virtual void	setOriginatingEntityID (const <u>DtEntityIdentifier</u> & entity) <i>Set/Get originating entity identifier</i>
 virtual const <u>DtEntityIdentifier</u> &	originatingEntityID () const
 virtual void	setReceivingEntityID (const <u>DtEntityIdentifier</u> & entity) <i>Set/Get receiving entity identifier</i>
 virtual const <u>DtEntityIdentifier</u> &	receivingEntityID () const
 virtual void	setNature (<u>DtNature</u> & value) <i>Set/Get nature</i>
 virtual <u>DtNature</u>	nature () const
 virtual void	setPosition (<u>DtPosition</u> & value) <i>Set/Get position</i>
 virtual <u>DtPosition</u>	position () const
 virtual void	setLocation (DtConstVector coords) <i>Set/Get part location</i>
 virtual DtConstVector	location () const
 virtual void	setStationName (<u>DtStationName</u> value) <i>Set/Get station name</i>
 virtual <u>DtStationName</u>	stationName () const
 virtual void	setStationNumber (DtU16 value) <i>Set/Get Station number</i>

virtual DtU16	stationNumber () const
virtual void	setEntityType (const DtEntityType & type) <i>Set/Get Part Entity Type</i>
virtual const DtEntityType &	entityType () const
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtIsPartOfPdu</i>
static void	addCallback (DtExerciseConn * conn, DtIsPartOfPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtIsPartOfPduCb cb, void* usr)

Protected Fields

static int	theStaticSize <i>Size of PDU with zero data bytes</i>
DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>
DtVector	myVector
DtEntityType	myEntityType

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
virtual DtNetIsPartOfPdu*	netPduStruct () const <i>Casts the netPdu element to our network struct Non-virtual since you can't override based on return type</i>
DtIsPartOfPdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type</i>

Inherited from [DtPdu](#):

Public Methods

virtual void* packet () const
virtual int status () const
virtual void setVersion (int vers)
virtual int protocolVersion () const
virtual void setExerciseld (int id)
virtual int exerciseld () const

- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const

- virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtIsPartOfPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors

- `DtIsPartOfPdu(const DtNetIsPartOfPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtIsPartOfPdu(const DtIsPartOfPdu& orig)`
- `virtual ~DtIsPartOfPdu()`
destructor
- `DtIsPartOfPdu& operator=(const DtIsPartOfPdu& orig)`
Assignment operator
- `virtual void setOriginatingEntityID(const DtEntityIdentifier& entity)`
Set/Get originating entity identifier
- `virtual const DtEntityIdentifier& originatingEntityID() const`
- `virtual void setReceivingEntityID(const DtEntityIdentifier& entity)`
Set/Get receiving entity identifier
- `virtual const DtEntityIdentifier& receivingEntityID() const`
- `virtual void setNature(DtNature& value)`
Set/Get nature
- `virtual DtNature nature() const`
- `virtual void setPosition(DtPosition& value)`
Set/Get position
- `virtual DtPosition position() const`
- `virtual void setLocation(DtConstVector coords)`
Set/Get part location
- `virtual DtConstVector location() const`
- `virtual void setStationName(DtStationName value)`
Set/Get station name
- `virtual DtStationName stationName() const`
- `virtual void setStationNumber(DtU16 value)`
Set/Get Station number
- `virtual DtU16 stationNumber() const`
- `virtual void setEntityType(const DtEntityType& type)`
Set/Get Part Entity Type
- `virtual const DtEntityType& entityType() const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer =`

`DtUSE_INTERNAL_BUFFER)`

Create a `DtIsPartOfPdu`. Caller is responsible for deletion.

• `static void addCallback(DtExerciseConn* conn, DtIsPartOfPduCb cb, void* usr)`

Add (register)/Remove (deregister) function to be called when pdu is received

• `static void removeCallback(DtExerciseConn* conn, DtIsPartOfPduCb cb, void* usr)`

• `virtual DtPduKind internalGetPduKind() const`

Returns the internal PDU kind

• `virtual DtNetIsPartOfPdu* netPduStruct() const`

Casts the netPdu element to our network struct Non-virtual since you can't override based on return type

• `DtIsPartOfPdu* self() const`

Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type

• `static int theStaticSize`

Size of PDU with zero data bytes

• `DtEntityIdentifier myEntityID`

Temporary objects for reference returns

• `DtVector myVector`

• `DtEntityType myEntityType`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtNature

IsPartOf PDU enums

-  DtOtherNature
 -  DtHostFireableMunition
 -  DtMunitionCarriedAsCargo
 -  DtFuelCarriedAsCarge
 -  DtGunmountAttachedToHost
 -  DtComputerGeneratedForcesCarriedAsCargo
 -  DtVehicleCarriedAsCargo
 -  DtEmitterMountedOnHost
 -  DtMobileCommandAndControlEntityCarriedAboardHost
 -  DtEntityStationedAt
 -  DtTeamMemberInFormationWith
-

Documentation

IsPartOf PDU enums

-  DtOtherNature
-  DtHostFireableMunition
-  DtMunitionCarriedAsCargo
-  DtFuelCarriedAsCarge
-  DtGunmountAttachedToHost
-  DtComputerGeneratedForcesCarriedAsCargo
-  DtVehicleCarriedAsCargo
-  DtEmitterMountedOnHost
-  DtMobileCommandAndControlEntityCarriedAboardHost
-  DtEntityStationedAt
-  DtTeamMemberInFormationWith

In file disenums.h:

enum DtPosition

 DtOtherPosition

 DtOnTopOf

 DtInsideOf

Documentation

 DtOtherPosition

 DtOnTopOf

 DtInsideOf

In file disenums.h:

enum DtStationName

-  DtOtherStationName
-  DtAircraftWingstation
-  DtShipsForwardGunmountStarboard
-  DtShipsForwardGunmountPort
-  DtShipsForwardGunmountCenterline
-  DtShipsAftGunmountStarboard
-  DtShipsAftGunmountPort
-  DtShipsAftGunmountCenterline
-  DtForwardTorpedoTube
-  DtAftTorpedoTube
-  DtBombBay
-  DtCargoBay
-  DtTruckBed
-  DtTrailerBed
-  DtWellDeck
-  DtOnStation_RNG_BRG
-  DtOnStation_XYZ

Documentation

-  DtOtherStationName
-  DtAircraftWingstation
-  DtShipsForwardGunmountStarboard
-  DtShipsForwardGunmountPort
-  DtShipsForwardGunmountCenterline
-  DtShipsAftGunmountStarboard

- DtShipsAftGunmountPort
- DtShipsAftGunmountCenterline
- DtForwardTorpedoTube
- DtAftTorpedoTube
- DtBombBay
- DtCargoBay
- DtTruckBed
- DtTrailerBed
- DtWellDeck
- DtOnStation_RNG_BRG
- DtOnStation_XYZ

[Alphabetic index](#) [Hierarchy of classes](#)

In file isgroupofpdu.h:

class DtlsGroupOfPdu: public DtPdu

Inheritance:

DtlsGroupOfPdu - DtPdu - DtBaseMsg

Public Methods

	DtlsGroupOfPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtlsGroupOfPdu (const DtNetIsGroupOfPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtlsGroupOfPdu (const DtlsGroupOfPdu & orig)
virtual	~DtlsGroupOfPdu () <i>destructor</i>
DtlsGroupOfPdu &	operator= (const DtlsGroupOfPdu & orig) <i>Assignment operator</i>
virtual void	setGroupEntityID (const DtEntityIdentifier & id) <i>Set/Get group entity ID</i>
virtual const DtEntityIdentifier &	groupEntityID () const
virtual void	setCategory (DtGroupedEntityCategory value) <i>Set/Get the grouped entity category (setCategory will also set the number of entities to zero)</i>
virtual DtGroupedEntityCategory	category () const
virtual DtBoolean	setNumEntities (DtU8 num) <i>Set/Get the number of entities</i>
virtual DtU8	numEntities () const
virtual void	setLatitude (DtFloat64 value) <i>Set/Get the reference latitude</i>
virtual DtFloat64	latitude () const
virtual void	setLongitude (DtFloat64 value) <i>Set/Get the reference longitude</i>
virtual DtFloat64	longitude () const
virtual DtBoolean	setBasicFixedWingAircraftRecord (int index, const DtBasicFixedWingAircraftRecord & init) <i>Set/Get a BasicFixedWingAircraftRecord</i>
virtual DtBoolean	getBasicFixedWingAircraftRecord (int index, DtBasicFixedWingAircraftRecord & record) const

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual [DtBoolean](#)

virtual void

static [DtPdu](#)*

static void

static void

setEnhancedFixedWingAircraftRecord (int index, const [DtEnhancedFixedWingAircraftRecord](#)& init)

Set/Get an EnhancedFixedWingAircraftRecord

getEnhancedFixedWingAircraftRecord (int index, [DtEnhancedFixedWingAircraftRecord](#)& record) const

setBasicRotorWingAircraftRecord (int index, const [DtBasicRotorWingAircraftRecord](#)& init)

Set/Get a BasicRotorWingAircraftRecord

getBasicRotorWingAircraftRecord (int index, [DtBasicRotorWingAircraftRecord](#)& record) const

setEnhancedRotorWingAircraftRecord (int index, const [DtEnhancedRotorWingAircraftRecord](#)& init)

Set/Get an EnhancedRotorWingAircraftRecord

getEnhancedRotorWingAircraftRecord (int index, [DtEnhancedRotorWingAircraftRecord](#)& record) const

setBasicGroundCombatSoldierRecord (int index, const [DtBasicGroundCombatSoldierRecord](#)& init)

Set/Get a BasicGroundCombatSoldierRecord

getBasicGroundCombatSoldierRecord (int index, [DtBasicGroundCombatSoldierRecord](#)& record) const

setEnhancedGroundCombatSoldierRecord (int index, const [DtEnhancedGroundCombatSoldierRecord](#)& init)

Set/Get an EnhancedGroundCombatSoldierRecord

getEnhancedGroundCombatSoldierRecord (int index, [DtEnhancedGroundCombatSoldierRecord](#)& record) const

setBasicGroundCombatVehicleRecord (int index, const [DtBasicGroundCombatVehicleRecord](#)& init)

Set/Get a BasicGroundCombatVehicleRecord

getBasicGroundCombatVehicleRecord (int index, [DtBasicGroundCombatVehicleRecord](#)& record) const

setEnhancedGroundCombatVehicleRecord (int index, const [DtEnhancedGroundCombatVehicleRecord](#)& init)

Set/Get an EnhancedGroundCombatVehicleRecord

getEnhancedGroundCombatVehicleRecord (int index, [DtEnhancedGroundCombatVehicleRecord](#)& record) const

setGroundLogisticsVehicleRecord (int index, const [DtGroundLogisticsVehicleRecord](#)& init)

Set/Get a GroundLogisticsVehicleRecord

getGroundLogisticsVehicleRecord (int index, [DtGroundLogisticsVehicleRecord](#)& record) const

printData () const

Print the data contained by this PDU

create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtIsGroupOfPdu

addCallback ([DtExerciseConn](#)* conn, DtIsGroupOfPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

removeCallback ([DtExerciseConn](#)* conn, DtIsGroupOfPduCb cb, void* usr)

Protected Fields

- static int **theStaticSize**
Size of PDU with zero parameter records
- [DtEntityIdentifier](#) **myEntityID**
Temporary objects for reference returns

Protected Methods

- virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind
 - virtual DtNetIsGroupOfPdu* **netPduStruct** () const
Casts the netPdu element to our network struct
 - [DtIsGroupOfPdu](#)* **self** () const
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
 - virtual char* **var** () const
Returns a pointer to the first byte of variable length data
 - virtual int **recordSize** () const
Returns the record size for the grouped entity category
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const

- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const

- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**

- void* **myInternalBuff**

- int **myIntBuffSize**

- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)

- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)

- virtual void **changeSize**(int size)

- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)

- virtual void **deleteBytes**(int offset, int numBytes)

- virtual void **deleteBytes**(void *address, int numBytes)

- virtual void **clearAll**()

- virtual void* **newInternalBuffer**(int size)

Documentation

- DtIsGroupOfPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Constructors

- DtIsGroupOfPdu(const DtNetIsGroupOfPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

- DtIsGroupOfPdu(const [DtIsGroupOfPdu](#)& orig)

- virtual ~DtIsGroupOfPdu()

destructor

- [DtIsGroupOfPdu](#)& operator=(const [DtIsGroupOfPdu](#)& orig)

Assignment operator

- virtual void setGroupEntityID(const [DtEntityIdentifier](#)& id)

Set/Get group entity ID

- virtual const [DtEntityIdentifier](#)& groupEntityID() const

virtual void setCategory([DtGroupedEntityCategory](#) value)

Set/Get the grouped entity category (setCategory will also set the number of entities to zero)

virtual [DtGroupedEntityCategory](#) category() const

virtual [DtBoolean](#) setNumEntities(DtU8 num)

Set/Get the number of entities

virtual DtU8 numEntities() const

virtual void setLatitude(DtFloat64 value)

Set/Get the reference latitude

virtual DtFloat64 latitude() const

virtual void setLongitude(DtFloat64 value)

Set/Get the reference longitude

virtual DtFloat64 longitude() const

virtual [DtBoolean](#) setBasicFixedWingAircraftRecord(int index, const [DtBasicFixedWingAircraftRecord](#)& init)

Set/Get a BasicFixedWingAircraftRecord. Returns DtFalse if the index is out of range, or if the entity category does not match.

virtual [DtBoolean](#) getBasicFixedWingAircraftRecord(int index, [DtBasicFixedWingAircraftRecord](#)& record) const

virtual [DtBoolean](#) setEnhancedFixedWingAircraftRecord(int index, const [DtEnhancedFixedWingAircraftRecord](#)& init)

Set/Get an EnhancedFixedWingAircraftRecord. Returns DtFalse if the index is out of range, or if the entity category does not match.

virtual [DtBoolean](#) getEnhancedFixedWingAircraftRecord(int index, [DtEnhancedFixedWingAircraftRecord](#)& record) const

virtual [DtBoolean](#) setBasicRotorWingAircraftRecord(int index, const [DtBasicRotorWingAircraftRecord](#)& init)

Set/Get a BasicRotorWingAircraftRecord. Returns DtFalse if the index is out of range, or if the entity category does not match.

virtual [DtBoolean](#) getBasicRotorWingAircraftRecord(int index, [DtBasicRotorWingAircraftRecord](#)& record) const

virtual [DtBoolean](#) setEnhancedRotorWingAircraftRecord(int index, const [DtEnhancedRotorWingAircraftRecord](#)& init)

Set/Get an EnhancedRotorWingAircraftRecord. Returns DtFalse if the index is out of range, or if the entity category does not match.

virtual [DtBoolean](#) getEnhancedRotorWingAircraftRecord(int index, [DtEnhancedRotorWingAircraftRecord](#)& record) const

virtual [DtBoolean](#) setBasicGroundCombatSoldierRecord(int index, const [DtBasicGroundCombatSoldierRecord](#)& init)

Set/Get a BasicGroundCombatSoldierRecord. Returns DtFalse if the index is out of range, or if the entity category

does not match.

virtual [DtBoolean](#) getBasicGroundCombatSoldierRecord(int index, [DtBasicGroundCombatSoldierRecord](#)& record) const

virtual [DtBoolean](#) setEnhancedGroundCombatSoldierRecord(int index, const [DtEnhancedGroundCombatSoldierRecord](#)& init)

Set/Get an EnhancedGroundCombatSoldierRecord. Returns DtFalse if the index is out of range, or if the entity category does not match.

virtual [DtBoolean](#) getEnhancedGroundCombatSoldierRecord(int index, [DtEnhancedGroundCombatSoldierRecord](#)& record) const

virtual [DtBoolean](#) setBasicGroundCombatVehicleRecord(int index, const [DtBasicGroundCombatVehicleRecord](#)& init)

Set/Get a BasicGroundCombatVehicleRecord. Returns DtFalse if the index is out of range, or if the entity category does not match.

virtual [DtBoolean](#) getBasicGroundCombatVehicleRecord(int index, [DtBasicGroundCombatVehicleRecord](#)& record) const

virtual [DtBoolean](#) setEnhancedGroundCombatVehicleRecord(int index, const [DtEnhancedGroundCombatVehicleRecord](#)& init)

Set/Get an EnhancedGroundCombatVehicleRecord. Returns DtFalse if the index is out of range, or if the entity category does not match.

virtual [DtBoolean](#) getEnhancedGroundCombatVehicleRecord(int index, [DtEnhancedGroundCombatVehicleRecord](#)& record) const

virtual [DtBoolean](#) setGroundLogisticsVehicleRecord(int index, const [DtGroundLogisticsVehicleRecord](#)& init)

Set/Get a GroundLogisticsVehicleRecord. Returns DtFalse if the index is out of range, or if the entity category does not match.

virtual [DtBoolean](#) getGroundLogisticsVehicleRecord(int index, [DtGroundLogisticsVehicleRecord](#)& record) const

virtual void printData() const

Print the data contained by this PDU

static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtIsGroupOfPdu. Caller is responsible for deletion.

static void addCallback([DtExerciseConn](#)* conn, DtIsGroupOfPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

static void removeCallback([DtExerciseConn](#)* conn, DtIsGroupOfPduCb cb, void* usr)

virtual [DtPduKind](#) internalGetPduKind() const

Returns the internal PDU kind

virtual DtNetIsGroupOfPdu* netPduStruct() const

Casts the netPdu element to our network struct

● [DtIsGroupOfPdu](#)* self() const

Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type

● virtual char* var() const

Returns a pointer to the first byte of variable length data

● virtual int recordSize() const

Returns the record size for the grouped entity category

● static int theStaticSize

Size of PDU with zero parameter records

● [DtEntityIdentifier](#) myEntityID

Temporary objects for reference returns

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtGroupedEntityCategory

IsGroupOf PDU enums

-  DtUndefinedGroupEntityCategory
-  DtBasicGroundCombatVehicle
-  DtEnhancedGroundCombatVehicle
-  DtBasicGroundCombatSoldier
-  DtEnhancedGroundCombatSoldier
-  DtBasicRotorWingAircraft
-  DtEnhancedRotorWingAircraft
-  DtBasicFixedWingAircraft
-  DtEnhancedFixedWingAircraft
-  DtGroundLogisticsVehicle

Documentation

IsGroupOf PDU enums

-  DtUndefinedGroupEntityCategory
-  DtBasicGroundCombatVehicle
-  DtEnhancedGroundCombatVehicle
-  DtBasicGroundCombatSoldier
-  DtEnhancedGroundCombatSoldier
-  DtBasicRotorWingAircraft
-  DtEnhancedRotorWingAircraft
-  DtBasicFixedWingAircraft
-  DtEnhancedFixedWingAircraft
-  DtGroundLogisticsVehicle

In file gedbasfixwing.h:

class DtBasicFixedWingAircraftRecord

class DtBasicFixedWingAircraftRecord: Instances of DtBasicFixedWingAircraftRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

Inheritance:

DtBasicFixedWingAircraftRecord

Public Methods

	DtBasicFixedWingAircraftRecord () <i>Constructors</i>
	DtBasicFixedWingAircraftRecord (const DtBasicFixedWingAircraftRecord & orig)
virtual	~DtBasicFixedWingAircraftRecord () <i>Destructor</i>
DtBasicFixedWingAircraftRecord &	operator= (const DtBasicFixedWingAircraftRecord & orig) <i>Assignment operators</i>
DtBasicFixedWingAircraftRecord &	operator= (const DtNetBasicFixedWingAircraftRecord& orig)
virtual void	setEntityID (DtU16 value) <i>Set/Get the entity ID</i>
virtual DtU16	entityID () const
virtual void	setX_Offset (DtInt16 value) <i>Set/Get the entity location X offset</i>
virtual DtInt16	X_Offset () const
virtual void	setY_Offset (DtInt16 value) <i>Set/Get the entity location Y offset</i>
virtual DtInt16	Y_Offset () const
virtual void	setZ_Offset (DtInt16 value) <i>Set/Get the entity location Z offset</i>
virtual DtInt16	Z_Offset () const
virtual void	setAppearance (DtU32 app) <i>Set/Get appearance (as a 32 bit integer)</i>
virtual DtU32	appearance () const
virtual void	setAppearanceAttribute (const DtAppearAttribute attr, const DtAppearAttrVal val) <i>Set/Get appearance by attribute</i>

virtual DtAppearAttrVal	appearanceAttribute (const DtAppearAttribute attr) const
virtual void	setPsi (DtInt8 value) <i>Set/Get the entity orientation psi</i>
virtual DtInt8	psi () const
virtual void	setTheta (DtInt8 value) <i>Set/Get the entity orientation theta</i>
virtual DtInt8	theta () const
virtual void	setPhi (DtInt8 value) <i>Set/Get the entity orientation phi</i>
virtual DtInt8	phi () const
virtual void	setFuelStatus (DtInt8 value) <i>Set/Get the fuel status</i>
virtual DtInt8	fuelStatus () const
virtual void	setHorizontalDeviation (DtInt8 value) <i>Set/Get the horizontal deviation</i>
virtual DtInt8	horizontalDeviation () const
virtual void	setVerticalDeviation (DtInt8 value) <i>Set/Get the vertical deviation</i>
virtual DtInt8	verticalDeviation () const
virtual void	setMovementSpeed (DtInt16 value) <i>Set/Get the movement speed</i>
virtual DtInt16	movementSpeed () const
virtual void	print () const <i>Print this record</i>
virtual void	unloadBaseNetRecord (const DtNetBasicFixedWingAircraftRecord& record) <i>Load/Unload data from a packet record</i>
virtual void	loadBaseNetRecord (DtNetBasicFixedWingAircraftRecord& record) const

Protected Fields

DtU16	myEntityID
DtInt16	myX_Offset
DtInt16	myY_Offset
DtInt16	myZ_Offset
DtU32	myAppearance
DtInt8	myPsi
DtInt8	myTheta
DtInt8	myPhi
DtU8	myFuelStatus
DtU8	myHorizontalDeviation
DtU8	myVerticalDeviation
DtInt16	myMovementSpeed

Documentation

class DtBasicFixedWingAircraftRecord: Instances of DtBasicFixedWingAircraftRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

`DtBasicFixedWingAircraftRecord()`

Constructors

`DtBasicFixedWingAircraftRecord( const DtBasicFixedWingAircraftRecord& orig)`

`virtual ~DtBasicFixedWingAircraftRecord()`

Destructor

`DtBasicFixedWingAircraftRecord& operator=( const DtBasicFixedWingAircraftRecord& orig)`

Assignment operators

`DtBasicFixedWingAircraftRecord& operator=( const DtNetBasicFixedWingAircraftRecord& orig)`

`virtual void setEntityID(DtU16 value)`

Set/Get the entity ID

`virtual DtU16 entityID() const`

`virtual void setX_Offset(DtInt16 value)`

Set/Get the entity location X offset

`virtual DtInt16 X_Offset() const`

`virtual void setY_Offset(DtInt16 value)`

Set/Get the entity location Y offset

`virtual DtInt16 Y_Offset() const`

`virtual void setZ_Offset(DtInt16 value)`

Set/Get the entity location Z offset

`virtual DtInt16 Z_Offset() const`

`virtual void setAppearance(DtU32 app)`

Set/Get appearance (as a 32 bit integer)

`virtual DtU32 appearance() const`

`virtual void setAppearanceAttribute( const DtAppearAttribute attr, const DtAppearAttrVal val)`

Set/Get appearance by attribute

`virtual DtAppearAttrVal appearanceAttribute( const DtAppearAttribute attr) const`

`virtual void setPsi(DtInt8 value)`

Set/Get the entity orientation psi

- virtual DtInt8 psi() const
- virtual void setTheta(DtInt8 value)
Set/Get the entity orientation theta
- virtual DtInt8 theta() const
- virtual void setPhi(DtInt8 value)
Set/Get the entity orientation phi
- virtual DtInt8 phi() const
- virtual void setFuelStatus(DtInt8 value)
Set/Get the fuel status
- virtual DtInt8 fuelStatus() const
- virtual void setHorizontalDeviation(DtInt8 value)
Set/Get the horizontal deviation
- virtual DtInt8 horizontalDeviation() const
- virtual void setVerticalDeviation(DtInt8 value)
Set/Get the vertical deviation
- virtual DtInt8 verticalDeviation() const
- virtual void setMovementSpeed(DtInt16 value)
Set/Get the movement speed
- virtual DtInt16 movementSpeed() const
- virtual void print() const
Print this record
- virtual void unloadBaseNetRecord(const DtNetBasicFixedWingAircraftRecord& record)
Load/Unload data from a packet record
- virtual void loadBaseNetRecord(DtNetBasicFixedWingAircraftRecord& record) const
- DtU16 myEntityID
- DtInt16 myX_Offset
- DtInt16 myY_Offset
- DtInt16 myZ_Offset
- DtU32 myAppearance
- DtInt8 myPsi
- DtInt8 myTheta
- DtInt8 myPhi
- DtU8 myFuelStatus
- DtU8 myHorizontalDeviation

● DtU8 myVerticalDeviation

● DtInt16 myMovementSpeed

Direct child classes:

[DtEnhancedFixedWingAircraftRecord](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file gedenhfixwing.h:

class DtEnhancedFixedWingAircraftRecord: public DtBasicFixedWingAircraftRecord

class DtEnhancedFixedWingAircraftRecord: Instances of DtEnhancedFixedWingAircraftRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

Inheritance:

DtEnhancedFixedWingAircraftRecord - DtBasicFixedWingAircraftRecord

Public Methods

	DtEnhancedFixedWingAircraftRecord () <i>Constructors</i>
	DtEnhancedFixedWingAircraftRecord (const DtEnhancedFixedWingAircraftRecord & orig)
 virtual	~DtEnhancedFixedWingAircraftRecord () <i>Destructor</i>
 DtEnhancedFixedWingAircraftRecord &	operator= (const DtEnhancedFixedWingAircraftRecord & orig) <i>Assignment operators</i>
 DtEnhancedFixedWingAircraftRecord &	operator= (const DtNetEnhancedFixedWingAircraftRecord& orig)
 virtual void	setSupplementalFuelStatus (DtU8 value) <i>Set/Get the fuel status</i>
 virtual DtU8	supplementalFuelStatus () const
 virtual void	setAirMaintenanceStatus (DtU8 value) <i>Set/Get the air maintenance status</i>
 virtual DtU8	airMaintenanceStatus () const
 virtual void	setPrimaryAmmunition (DtU8 value) <i>Set/Get the primary ammunition</i>
 virtual DtU8	primaryAmmunition () const
 virtual void	setSecondaryAmmunition (DtU8 value) <i>Set/Get the secondary ammunition</i>
 virtual DtU8	secondaryAmmunition () const
 virtual void	print () const <i>Print this record</i>
 virtual void	unloadNetRecord (const DtNetEnhancedFixedWingAircraftRecord& record) <i>Load/Unload data from a packet record</i>

virtual void

loadNetRecord (DtNetEnhancedFixedWingAircraftRecord& record) const

Protected Fields

DtU8 **mySupplementalFuelStatus**

DtU8 **myAirMaintenanceStatus**

DtU8 **myPrimaryAmmunition**

DtU8 **mySecondaryAmmunition**

Inherited from [DtBasicFixedWingAircraftRecord](#):

Public Methods

virtual void **setEntityID**(DtU16 value)

virtual DtU16 **entityID**() const

virtual void **setX_Offset**(DtInt16 value)

virtual DtInt16 **X_Offset**() const

virtual void **setY_Offset**(DtInt16 value)

virtual DtInt16 **Y_Offset**() const

virtual void **setZ_Offset**(DtInt16 value)

virtual DtInt16 **Z_Offset**() const

virtual void **setAppearance**(DtU32 app)

virtual DtU32 **appearance**() const

virtual void **setAppearanceAttribute**(const [DtAppearAttribute](#) attr, const [DtAppearAttrVal](#) val)

virtual [DtAppearAttrVal](#) **appearanceAttribute**(const [DtAppearAttribute](#) attr) const

virtual void **setPsi**(DtInt8 value)

virtual DtInt8 **psi**() const

virtual void **setTheta**(DtInt8 value)

virtual DtInt8 **theta**() const

virtual void **setPhi**(DtInt8 value)

virtual DtInt8 **phi**() const

virtual void **setFuelStatus**(DtInt8 value)

virtual DtInt8 **fuelStatus**() const

virtual void **setHorizontalDeviation**(DtInt8 value)

- virtual DtInt8 **horizontalDeviation**() const
- virtual void **setVerticalDeviation**(DtInt8 value)
- virtual DtInt8 **verticalDeviation**() const
- virtual void **setMovementSpeed**(DtInt16 value)
- virtual DtInt16 **movementSpeed**() const
- virtual void **unloadBaseNetRecord**(const DtNetBasicFixedWingAircraftRecord& record)
- virtual void **loadBaseNetRecord**(DtNetBasicFixedWingAircraftRecord& record) const

Protected Fields

- DtU16 **myEntityID**
 - DtInt16 **myX_Offset**
 - DtInt16 **myY_Offset**
 - DtInt16 **myZ_Offset**
 - DtU32 **myAppearance**
 - DtInt8 **myPsi**
 - DtInt8 **myTheta**
 - DtInt8 **myPhi**
 - DtU8 **myFuelStatus**
 - DtU8 **myHorizontalDeviation**
 - DtU8 **myVerticalDeviation**
 - DtInt16 **myMovementSpeed**
-

Documentation

class DtEnhancedFixedWingAircraftRecord: Instances of DtEnhancedFixedWingAircraftRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

- DtEnhancedFixedWingAircraftRecord()**
Constructors
- DtEnhancedFixedWingAircraftRecord(const [DtEnhancedFixedWingAircraftRecord](#)& orig)**
- virtual ~DtEnhancedFixedWingAircraftRecord()**
Destructor
- [DtEnhancedFixedWingAircraftRecord](#)& operator=(const [DtEnhancedFixedWingAircraftRecord](#)& orig)**

Assignment operators

- [DtEnhancedFixedWingAircraftRecord](#)& operator=(const DtNetEnhancedFixedWingAircraftRecord& orig)
- virtual void setSupplementalFuelStatus(DtU8 value)
Set/Get the fuel status
- virtual DtU8 supplementalFuelStatus() const
- virtual void setAirMaintenanceStatus(DtU8 value)
Set/Get the air maintenance status
- virtual DtU8 airMaintenanceStatus() const
- virtual void setPrimaryAmmunition(DtU8 value)
Set/Get the primary ammunition
- virtual DtU8 primaryAmmunition() const
- virtual void setSecondaryAmmunition(DtU8 value)
Set/Get the secondary ammunition
- virtual DtU8 secondaryAmmunition() const
- virtual void print() const
Print this record
- virtual void unloadNetRecord(const DtNetEnhancedFixedWingAircraftRecord& record)
Load/Unload data from a packet record
- virtual void loadNetRecord(DtNetEnhancedFixedWingAircraftRecord& record) const
- DtU8 mySupplementalFuelStatus
- DtU8 myAirMaintenanceStatus
- DtU8 myPrimaryAmmunition
- DtU8 mySecondaryAmmunition

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file gedbasrotwing.h:

class DtBasicRotorWingAircraftRecord

class DtBasicRotorWingAircraftRecord: Instances of DtBasicRotorWingAircraftRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

Inheritance:

DtBasicRotorWingAircraftRecord

Public Methods

	DtBasicRotorWingAircraftRecord () <i>Constructors</i>
	DtBasicRotorWingAircraftRecord (const DtBasicRotorWingAircraftRecord & orig)
virtual	~DtBasicRotorWingAircraftRecord () <i>Destructor</i>
DtBasicRotorWingAircraftRecord &	operator= (const DtBasicRotorWingAircraftRecord & orig) <i>Assignment operators</i>
DtBasicRotorWingAircraftRecord &	operator= (const DtNetBasicRotorWingAircraftRecord& orig)
virtual void	setEntityID (DtU16 value) <i>Set/Get the entity ID</i>
virtual DtU16	entityID () const
virtual void	setX_Offset (DtInt16 value) <i>Set/Get the entity location X offset</i>
virtual DtInt16	X_Offset () const
virtual void	setY_Offset (DtInt16 value) <i>Set/Get the entity location Y offset</i>
virtual DtInt16	Y_Offset () const
virtual void	setZ_Offset (DtInt16 value) <i>Set/Get the entity location Z offset</i>
virtual DtInt16	Z_Offset () const
virtual void	setAppearance (DtU32 app) <i>Set/Get appearance (as a 32 bit integer)</i>
virtual DtU32	appearance () const
virtual void	setAppearanceAttribute (const DtAppearAttribute attr, const DtAppearAttrVal val) <i>Set/Get appearance by attribute</i>

virtual [DtAppearAttrVal](#)

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt16

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual void

virtual void

appearanceAttribute (const [DtAppearAttribute](#) attr) const

setPsi (DtInt8 value)
Set/Get the entity orientation psi

psi () const

setTheta (DtInt8 value)
Set/Get the entity orientation theta

theta () const

setPhi (DtInt8 value)
Set/Get the entity orientation phi

phi () const

setFuelStatus (DtInt8 value)
Set/Get the fuel status

fuelStatus () const

setHorizontalDeviation (DtInt8 value)
Set/Get the horizontal deviation

horizontalDeviation () const

setVerticalDeviation (DtInt8 value)
Set/Get the vertical deviation

verticalDeviation () const

setMovementSpeed (DtInt16 value)
Set/Get the movement speed

movementSpeed () const

setTurretAzimuth (DtInt8 value)
Set/Get the turret azimuth

turretAzimuth () const

setGunElevation (DtInt8 value)
Set/Get the gun elevation

gunElevation () const

setTurretSlewRate (DtInt8 value)
Set/Get the turret slew rate

turretSlewRate () const

setGunElevationRate (DtInt8 value)
Set/Get the gun elevation rate

gunElevationRate () const

print () const
Print this record

unloadBaseNetRecord (const DtNetBasicRotorWingAircraftRecord& record)
Load/Unload data from a packet record

loadBaseNetRecord (DtNetBasicRotorWingAircraftRecord& record) const

Protected Fields

DtU16 **myEntityID**

DtInt16 **myX_Offset**

- DtInt16 **myY_Offset**
 - DtInt16 **myZ_Offset**
 - DtU32 **myAppearance**
 - DtInt8 **myPsi**
 - DtInt8 **myTheta**
 - DtInt8 **myPhi**
 - DtU8 **myFuelStatus**
 - DtU8 **myHorizontalDeviation**
 - DtU8 **myVerticalDeviation**
 - DtInt16 **myMovementSpeed**
 - DtInt8 **myTurretAzimuth**
 - DtInt8 **myGunElevation**
 - DtInt8 **myTurretSlewRate**
 - DtInt8 **myGunElevationRate**
-

Documentation

class DtBasicRotorWingAircraftRecord: Instances of DtBasicRotorWingAircraftRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

- `DtBasicRotorWingAircraftRecord()`
Constructors

- `DtBasicRotorWingAircraftRecord( const DtBasicRotorWingAircraftRecord& orig)`

- `virtual ~DtBasicRotorWingAircraftRecord()`
Destructor

- `DtBasicRotorWingAircraftRecord& operator=( const DtBasicRotorWingAircraftRecord& orig)`
Assignment operators

- `DtBasicRotorWingAircraftRecord& operator=( const DtNetBasicRotorWingAircraftRecord& orig)`

- `virtual void setEntityID(DtU16 value)`
Set/Get the entity ID

- `virtual DtU16 entityID() const`

- `virtual void setX_Offset(DtInt16 value)`
Set/Get the entity location X offset

- `virtual DtInt16 X_Offset() const`

- `virtual void setY_Offset(DtInt16 value)`

Set/Get the entity location Y offset

virtual DtInt16 Y_Offset() const

virtual void setZ_Offset(DtInt16 value)

Set/Get the entity location Z offset

virtual DtInt16 Z_Offset() const

virtual void setAppearance(DtU32 app)

Set/Get appearance (as a 32 bit integer)

virtual DtU32 appearance() const

virtual void setAppearanceAttribute(const [DtAppearAttribute](#) attr, const [DtAppearAttrVal](#) val)

Set/Get appearance by attribute

virtual [DtAppearAttrVal](#) appearanceAttribute(const [DtAppearAttribute](#) attr) const

virtual void setPsi(DtInt8 value)

Set/Get the entity orientation psi

virtual DtInt8 psi() const

virtual void setTheta(DtInt8 value)

Set/Get the entity orientation theta

virtual DtInt8 theta() const

virtual void setPhi(DtInt8 value)

Set/Get the entity orientation phi

virtual DtInt8 phi() const

virtual void setFuelStatus(DtInt8 value)

Set/Get the fuel status

virtual DtInt8 fuelStatus() const

virtual void setHorizontalDeviation(DtInt8 value)

Set/Get the horizontal deviation

virtual DtInt8 horizontalDeviation() const

virtual void setVerticalDeviation(DtInt8 value)

Set/Get the vertical deviation

virtual DtInt8 verticalDeviation() const

virtual void setMovementSpeed(DtInt16 value)

Set/Get the movement speed

virtual DtInt16 movementSpeed() const

virtual void setTurretAzimuth(DtInt8 value)

Set/Get the turret azimuth

- `virtual DtInt8 turretAzimuth() const`
- `virtual void setGunElevation(DtInt8 value)`
Set/Get the gun elevation
- `virtual DtInt8 gunElevation() const`
- `virtual void setTurretSlewRate(DtInt8 value)`
Set/Get the turret slew rate
- `virtual DtInt8 turretSlewRate() const`
- `virtual void setGunElevationRate(DtInt8 value)`
Set/Get the gun elevation rate
- `virtual DtInt8 gunElevationRate() const`
- `virtual void print() const`
Print this record
- `virtual void unloadBaseNetRecord( const DtNetBasicRotorWingAircraftRecord& record)`
Load/Unload data from a packet record
- `virtual void loadBaseNetRecord( DtNetBasicRotorWingAircraftRecord& record) const`
- `DtU16 myEntityID`
- `DtInt16 myX_Offset`
- `DtInt16 myY_Offset`
- `DtInt16 myZ_Offset`
- `DtU32 myAppearance`
- `DtInt8 myPsi`
- `DtInt8 myTheta`
- `DtInt8 myPhi`
- `DtU8 myFuelStatus`
- `DtU8 myHorizontalDeviation`
- `DtU8 myVerticalDeviation`
- `DtInt16 myMovementSpeed`
- `DtInt8 myTurretAzimuth`
- `DtInt8 myGunElevation`
- `DtInt8 myTurretSlewRate`
- `DtInt8 myGunElevationRate`

Direct child classes:

[DtEnhancedRotorWingAircraftRecord](#)

In file gedenhrotwing.h:

class DtEnhancedRotorWingAircraftRecord: public DtBasicRotorWingAircraftRecord

class DtEnhancedRotorWingAircraftRecord: Instances of DtEnhancedRotorWingAircraftRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

Inheritance:

DtEnhancedRotorWingAircraftRecord - DtBasicRotorWingAircraftRecord

Public Methods

	DtEnhancedRotorWingAircraftRecord () <i>Constructors</i>
	DtEnhancedRotorWingAircraftRecord (const DtEnhancedRotorWingAircraftRecord & orig)
 virtual	~DtEnhancedRotorWingAircraftRecord () <i>Destructor</i>
 DtEnhancedRotorWingAircraftRecord &	operator= (const DtEnhancedRotorWingAircraftRecord & orig) <i>Assignment operators</i>
 DtEnhancedRotorWingAircraftRecord &	operator= (const DtNetEnhancedRotorWingAircraftRecord& orig)
 virtual void	setSupplementalFuelStatus (DtU8 value) <i>Set/Get the fuel status</i>
 virtual DtU8	supplementalFuelStatus () const
 virtual void	setAirMaintenanceStatus (DtU8 value) <i>Set/Get the air maintenance status</i>
 virtual DtU8	airMaintenanceStatus () const
 virtual void	setPrimaryAmmunition (DtU8 value) <i>Set/Get the primary ammunition</i>
 virtual DtU8	primaryAmmunition () const
 virtual void	setSecondaryAmmunition (DtU8 value) <i>Set/Get the secondary ammunition</i>
 virtual DtU8	secondaryAmmunition () const
 virtual void	print () const <i>Print this record</i>
 virtual void	unloadNetRecord (const DtNetEnhancedRotorWingAircraftRecord& record) <i>Load/Unload data from a packet record</i>

virtual void

loadNetRecord (DtNetEnhancedRotorWingAircraftRecord& record) const

Protected Fields

DtU8 **mySupplementalFuelStatus**

DtU8 **myAirMaintenanceStatus**

DtU8 **myPrimaryAmmunition**

DtU8 **mySecondaryAmmunition**

Inherited from [DtBasicRotorWingAircraftRecord](#):

Public Methods

virtual void **setEntityID**(DtU16 value)

virtual DtU16 **entityID**() const

virtual void **setX_Offset**(DtInt16 value)

virtual DtInt16 **X_Offset**() const

virtual void **setY_Offset**(DtInt16 value)

virtual DtInt16 **Y_Offset**() const

virtual void **setZ_Offset**(DtInt16 value)

virtual DtInt16 **Z_Offset**() const

virtual void **setAppearance**(DtU32 app)

virtual DtU32 **appearance**() const

virtual void **setAppearanceAttribute**(const [DtAppearAttribute](#) attr, const [DtAppearAttrVal](#) val)

virtual [DtAppearAttrVal](#) **appearanceAttribute**(const [DtAppearAttribute](#) attr) const

virtual void **setPsi**(DtInt8 value)

virtual DtInt8 **psi**() const

virtual void **setTheta**(DtInt8 value)

virtual DtInt8 **theta**() const

virtual void **setPhi**(DtInt8 value)

virtual DtInt8 **phi**() const

virtual void **setFuelStatus**(DtInt8 value)

virtual DtInt8 **fuelStatus**() const

virtual void **setHorizontalDeviation**(DtInt8 value)

- virtual DtInt8 **horizontalDeviation**() const
- virtual void **setVerticalDeviation**(DtInt8 value)
- virtual DtInt8 **verticalDeviation**() const
- virtual void **setMovementSpeed**(DtInt16 value)
- virtual DtInt16 **movementSpeed**() const
- virtual void **setTurretAzimuth**(DtInt8 value)
- virtual DtInt8 **turretAzimuth**() const
- virtual void **setGunElevation**(DtInt8 value)
- virtual DtInt8 **gunElevation**() const
- virtual void **setTurretSlewRate**(DtInt8 value)
- virtual DtInt8 **turretSlewRate**() const
- virtual void **setGunElevationRate**(DtInt8 value)
- virtual DtInt8 **gunElevationRate**() const
- virtual void **unloadBaseNetRecord**(const DtNetBasicRotorWingAircraftRecord& record)
- virtual void **loadBaseNetRecord**(DtNetBasicRotorWingAircraftRecord& record) const

Protected Fields

- DtU16 **myEntityID**
- DtInt16 **myX_Offset**
- DtInt16 **myY_Offset**
- DtInt16 **myZ_Offset**
- DtU32 **myAppearance**
- DtInt8 **myPsi**
- DtInt8 **myTheta**
- DtInt8 **myPhi**
- DtU8 **myFuelStatus**
- DtU8 **myHorizontalDeviation**
- DtU8 **myVerticalDeviation**
- DtInt16 **myMovementSpeed**
- DtInt8 **myTurretAzimuth**
- DtInt8 **myGunElevation**
- DtInt8 **myTurretSlewRate**

Documentation

class DtEnhancedRotorWingAircraftRecord: Instances of DtEnhancedRotorWingAircraftRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

`DtEnhancedRotorWingAircraftRecord()`

Constructors

`DtEnhancedRotorWingAircraftRecord( const DtEnhancedRotorWingAircraftRecord& orig)`

`virtual ~DtEnhancedRotorWingAircraftRecord()`

Destructor

`DtEnhancedRotorWingAircraftRecord& operator=( const DtEnhancedRotorWingAircraftRecord& orig)`

Assignment operators

`DtEnhancedRotorWingAircraftRecord& operator=( const DtNetEnhancedRotorWingAircraftRecord& orig)`

`virtual void setSupplementalFuelStatus(DtU8 value)`

Set/Get the fuel status

`virtual DtU8 supplementalFuelStatus() const`

`virtual void setAirMaintenanceStatus(DtU8 value)`

Set/Get the air maintenance status

`virtual DtU8 airMaintenanceStatus() const`

`virtual void setPrimaryAmmunition(DtU8 value)`

Set/Get the primary ammunition

`virtual DtU8 primaryAmmunition() const`

`virtual void setSecondaryAmmunition(DtU8 value)`

Set/Get the secondary ammunition

`virtual DtU8 secondaryAmmunition() const`

`virtual void print() const`

Print this record

`virtual void unloadNetRecord( const DtNetEnhancedRotorWingAircraftRecord& record)`

Load/Unload data from a packet record

`virtual void loadNetRecord( DtNetEnhancedRotorWingAircraftRecord& record) const`

`DtU8 mySupplementalFuelStatus`

`DtU8 myAirMaintenanceStatus`

- DtU8 myPrimaryAmmunition
- DtU8 mySecondaryAmmunition

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file gedbassoldier.h:

class DtBasicGroundCombatSoldierRecord

class DtBasicGroundCombatSoldierRecord: Instances of DtBasicGroundCombatSoldierRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

Inheritance:

DtBasicGroundCombatSoldierRecord

Public Methods

	DtBasicGroundCombatSoldierRecord () <i>Constructors</i>
	DtBasicGroundCombatSoldierRecord (const DtBasicGroundCombatSoldierRecord& orig)
 virtual	~DtBasicGroundCombatSoldierRecord () <i>Destructor</i>
 DtBasicGroundCombatSoldierRecord &	operator= (const DtBasicGroundCombatSoldierRecord& orig) <i>Assignment operators</i>
 DtBasicGroundCombatSoldierRecord &	operator= (const DtNetBasicGroundCombatSoldierRecord& orig)
 virtual void	setEntityID (DtU16 value) <i>Set/Get the entity ID</i>
 virtual DtU16	entityID () const
 virtual void	setX_Offset (DtInt8 value) <i>Set/Get the entity location X offset</i>
 virtual DtInt8	X_Offset () const
 virtual void	setY_Offset (DtInt8 value) <i>Set/Get the entity location Y offset</i>
 virtual DtInt8	Y_Offset () const
 virtual void	setZ_Offset (DtInt8 value) <i>Set/Get the entity location Z offset</i>
 virtual DtInt8	Z_Offset () const
 virtual void	setAppearance (DtU32 app) <i>Set/Get appearance (as a 32 bit integer)</i>
 virtual DtU32	appearance () const
 virtual void	setAppearanceAttribute (const DtAppearAttribute attr, const DtAppearAttrVal val) <i>Set/Get appearance by attribute</i>

virtual [DtAppearAttrVal](#)

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual void

virtual void

appearanceAttribute (const [DtAppearAttribute](#) attr) const

setPsi (DtInt8 value)

Set/Get the entity orientation psi

psi () const

setTheta (DtInt8 value)

Set/Get the entity orientation theta

theta () const

setPhi (DtInt8 value)

Set/Get the entity orientation phi

phi () const

setEntitySpeed (DtInt8 value)

Set/Get the entity speed

entitySpeed () const

setHeadAzimuth (DtInt8 value)

Set/Get the head azimuth

headAzimuth () const

setHeadElevation (DtInt8 value)

Set/Get the head elevation

headElevation () const

setHeadScanRate (DtInt8 value)

Set/Get the head scan rate

headScanRate () const

setHeadElevationRate (DtInt8 value)

Set/Get the head elevation rate

headElevationRate () const

print () const

Print this record

unloadBaseNetRecord (const DtNetBasicGroundCombatSoldierRecord& record)

Load/Unload data from a packet record

loadBaseNetRecord (DtNetBasicGroundCombatSoldierRecord& record) const

Protected Fields

DtU16 **myEntityID**

DtInt8 **myX_Offset**

DtInt8 **myY_Offset**

DtInt8 **myZ_Offset**

DtU32 **myAppearance**

DtInt8 **myPsi**

DtInt8 **myTheta**

DtInt8 **myPhi**

- DtInt8 **myEntitySpeed**
 - DtInt8 **myHeadAzimuth**
 - DtInt8 **myHeadElevation**
 - DtInt8 **myHeadScanRate**
 - DtInt8 **myHeadElevationRate**
-

Documentation

class DtBasicGroundCombatSoldierRecord: Instances of DtBasicGroundCombatSoldierRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

- `DtBasicGroundCombatSoldierRecord()`
Constructors

- `DtBasicGroundCombatSoldierRecord( const DtBasicGroundCombatSoldierRecord& orig)`

- `virtual ~DtBasicGroundCombatSoldierRecord()`
Destructor

- `DtBasicGroundCombatSoldierRecord& operator=( const DtBasicGroundCombatSoldierRecord& orig)`
Assignment operators

- `DtBasicGroundCombatSoldierRecord& operator=( const DtNetBasicGroundCombatSoldierRecord& orig)`

- `virtual void setEntityID(DtU16 value)`
Set/Get the entity ID

- `virtual DtU16 entityID() const`

- `virtual void setX_Offset(DtInt8 value)`
Set/Get the entity location X offset

- `virtual DtInt8 X_Offset() const`

- `virtual void setY_Offset(DtInt8 value)`
Set/Get the entity location Y offset

- `virtual DtInt8 Y_Offset() const`

- `virtual void setZ_Offset(DtInt8 value)`
Set/Get the entity location Z offset

- `virtual DtInt8 Z_Offset() const`

- `virtual void setAppearance(DtU32 app)`
Set/Get appearance (as a 32 bit integer)

- `virtual DtU32 appearance() const`

- `virtual void setAppearanceAttribute( const DtAppearAttribute attr, const DtAppearAttrVal val)`
Set/Get appearance by attribute
- `virtual DtAppearAttrVal appearanceAttribute( const DtAppearAttribute attr) const`
- `virtual void setPsi(DtInt8 value)`
Set/Get the entity orientation psi
- `virtual DtInt8 psi() const`
- `virtual void setTheta(DtInt8 value)`
Set/Get the entity orientation theta
- `virtual DtInt8 theta() const`
- `virtual void setPhi(DtInt8 value)`
Set/Get the entity orientation phi
- `virtual DtInt8 phi() const`
- `virtual void setEntitySpeed(DtInt8 value)`
Set/Get the entity speed
- `virtual DtInt8 entitySpeed() const`
- `virtual void setHeadAzimuth(DtInt8 value)`
Set/Get the head azimuth
- `virtual DtInt8 headAzimuth() const`
- `virtual void setHeadElevation(DtInt8 value)`
Set/Get the head elevation
- `virtual DtInt8 headElevation() const`
- `virtual void setHeadScanRate(DtInt8 value)`
Set/Get the head scan rate
- `virtual DtInt8 headScanRate() const`
- `virtual void setHeadElevationRate(DtInt8 value)`
Set/Get the head elevation rate
- `virtual DtInt8 headElevationRate() const`
- `virtual void print() const`
Print this record
- `virtual void unloadBaseNetRecord( const DtNetBasicGroundCombatSoldierRecord& record)`
Load/Unload data from a packet record
- `virtual void loadBaseNetRecord( DtNetBasicGroundCombatSoldierRecord& record) const`
- `DtU16 myEntityID`
- `DtInt8 myX_Offset`

- DtInt8 myY_Offset
 - DtInt8 myZ_Offset
 - DtU32 myAppearance
 - DtInt8 myPsi
 - DtInt8 myTheta
 - DtInt8 myPhi
 - DtInt8 myEntitySpeed
 - DtInt8 myHeadAzimuth
 - DtInt8 myHeadElevation
 - DtInt8 myHeadScanRate
 - DtInt8 myHeadElevationRate
-

Direct child classes:

[DtEnhancedGroundCombatSoldierRecord](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file gedenhsoldier.h:

class [DtEnhancedGroundCombatSoldierRecord](#): public [DtBasicGroundCombatSoldierRecord](#)

class DtEnhancedGroundCombatSoldierRecord: Instances of DtEnhancedGroundCombatSoldierRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

Inheritance:

DtEnhancedGroundCombatSoldierRecord - [DtBasicGroundCombatSoldierRecord](#)

Public Methods

	DtEnhancedGroundCombatSoldierRecord () <i>Constructors</i>
	DtEnhancedGroundCombatSoldierRecord (const <u>DtEnhancedGroundCombatSoldierRecord</u>& orig)
 virtual	~DtEnhancedGroundCombatSoldierRecord () <i>Destructor</i>
 <u>DtEnhancedGroundCombatSoldierRecord</u> &	operator= (const <u>DtEnhancedGroundCombatSoldierRecord</u>& orig) <i>Assignment operators</i>
 <u>DtEnhancedGroundCombatSoldierRecord</u> &	operator= (const DtNetEnhancedGroundCombatSoldierRecord& orig)
 virtual void	setWaterStatus (DtU8 value) <i>Set/Get the water status</i>
 virtual DtU8	waterStatus () const
 virtual void	setRestStatus (<u>DtRestStatus</u> value) <i>Set/Get the rest status</i>
 virtual <u>DtRestStatus</u>	restStatus () const
 virtual void	setPrimaryAmmunition (DtU8 value) <i>Set/Get the primary ammunition</i>
 virtual DtU8	primaryAmmunition () const
 virtual void	setSecondaryAmmunition (DtU8 value) <i>Set/Get the secondary ammunition</i>
 virtual DtU8	secondaryAmmunition () const
 virtual void	print () const <i>Print this record</i>

virtual void

virtual void

unloadNetRecord (const
DtNetEnhancedGroundCombatSoldierRecord& record)
Load/Unload data from a packet record
loadNetRecord (DtNetEnhancedGroundCombatSoldierRecord& record)
const

Protected Fields

- DtU8 **myWaterStatus**
 - DtU8 **myRestStatus**
 - DtU8 **myPrimaryAmmunition**
 - DtU8 **mySecondaryAmmunition**
-

Inherited from [DtBasicGroundCombatSoldierRecord](#):

Public Methods

- virtual void **setEntityID**(DtU16 value)
- virtual DtU16 **entityID**() const
- virtual void **setX_Offset**(DtInt8 value)
- virtual DtInt8 **X_Offset**() const
- virtual void **setY_Offset**(DtInt8 value)
- virtual DtInt8 **Y_Offset**() const
- virtual void **setZ_Offset**(DtInt8 value)
- virtual DtInt8 **Z_Offset**() const
- virtual void **setAppearance**(DtU32 app)
- virtual DtU32 **appearance**() const
- virtual void **setAppearanceAttribute**(const [DtAppearAttribute](#) attr, const [DtAppearAttrVal](#) val)
- virtual [DtAppearAttrVal](#) **appearanceAttribute**(const [DtAppearAttribute](#) attr) const
- virtual void **setPsi**(DtInt8 value)
- virtual DtInt8 **psi**() const
- virtual void **setTheta**(DtInt8 value)
- virtual DtInt8 **theta**() const
- virtual void **setPhi**(DtInt8 value)
- virtual DtInt8 **phi**() const
- virtual void **setEntitySpeed**(DtInt8 value)

- virtual DtInt8 **entitySpeed**() const
- virtual void **setHeadAzimuth**(DtInt8 value)
- virtual DtInt8 **headAzimuth**() const
- virtual void **setHeadElevation**(DtInt8 value)
- virtual DtInt8 **headElevation**() const
- virtual void **setHeadScanRate**(DtInt8 value)
- virtual DtInt8 **headScanRate**() const
- virtual void **setHeadElevationRate**(DtInt8 value)
- virtual DtInt8 **headElevationRate**() const
- virtual void **unloadBaseNetRecord**(const DtNetBasicGroundCombatSoldierRecord& record)
- virtual void **loadBaseNetRecord**(DtNetBasicGroundCombatSoldierRecord& record) const

Protected Fields

- DtU16 **myEntityID**
- DtInt8 **myX_Offset**
- DtInt8 **myY_Offset**
- DtInt8 **myZ_Offset**
- DtU32 **myAppearance**
- DtInt8 **myPsi**
- DtInt8 **myTheta**
- DtInt8 **myPhi**
- DtInt8 **myEntitySpeed**
- DtInt8 **myHeadAzimuth**
- DtInt8 **myHeadElevation**
- DtInt8 **myHeadScanRate**
- DtInt8 **myHeadElevationRate**

Documentation

class DtEnhancedGroundCombatSoldierRecord: Instances of DtEnhancedGroundCombatSoldierRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

- DtEnhancedGroundCombatSoldierRecord()**

Constructors

• `DtEnhancedGroundCombatSoldierRecord( const DtEnhancedGroundCombatSoldierRecord& orig)`

• `virtual ~DtEnhancedGroundCombatSoldierRecord()`

Destructor

• `DtEnhancedGroundCombatSoldierRecord& operator=( const DtEnhancedGroundCombatSoldierRecord& orig)`

Assignment operators

• `DtEnhancedGroundCombatSoldierRecord& operator=( const DtNetEnhancedGroundCombatSoldierRecord& orig)`

• `virtual void setWaterStatus(DtU8 value)`

Set/Get the water status

• `virtual DtU8 waterStatus() const`

• `virtual void setRestStatus(DtRestStatus value)`

Set/Get the rest status

• `virtual DtRestStatus restStatus() const`

• `virtual void setPrimaryAmmunition(DtU8 value)`

Set/Get the primary ammunition

• `virtual DtU8 primaryAmmunition() const`

• `virtual void setSecondaryAmmunition(DtU8 value)`

Set/Get the secondary ammunition

• `virtual DtU8 secondaryAmmunition() const`

• `virtual void print() const`

Print this record

• `virtual void unloadNetRecord( const DtNetEnhancedGroundCombatSoldierRecord& record)`

Load/Unload data from a packet record

• `virtual void loadNetRecord( DtNetEnhancedGroundCombatSoldierRecord& record) const`

• `DtU8 myWaterStatus`

• `DtU8 myRestStatus`

• `DtU8 myPrimaryAmmunition`

• `DtU8 mySecondaryAmmunition`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtRestStatus

-  DtSoldierNotRested
 -  DtSoldier1HourSleep
 -  DtSoldier2HourSleep
 -  DtSoldier3HourSleep
 -  DtSoldier4HourSleep
 -  DtSoldier5HourSleep
 -  DtSoldier6HourSleep
 -  DtSoldier7HourSleep
 -  DtSoldierFullyRested
-

Documentation

-  DtSoldierNotRested
-  DtSoldier1HourSleep
-  DtSoldier2HourSleep
-  DtSoldier3HourSleep
-  DtSoldier4HourSleep
-  DtSoldier5HourSleep
-  DtSoldier6HourSleep
-  DtSoldier7HourSleep
-  DtSoldierFullyRested

In file gedbasvehicle.h:

class DtBasicGroundCombatVehicleRecord

class DtBasicGroundCombatVehicleRecord: Instances of DtBasicGroundCombatVehicleRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

Inheritance:

DtBasicGroundCombatVehicleRecord

Public Methods

	DtBasicGroundCombatVehicleRecord () <i>Constructors</i>
	DtBasicGroundCombatVehicleRecord (const DtBasicGroundCombatVehicleRecord& orig)
 virtual	~DtBasicGroundCombatVehicleRecord () <i>Destructor</i>
 DtBasicGroundCombatVehicleRecord &	operator= (const DtBasicGroundCombatVehicleRecord& orig) <i>Assignment operators</i>
 DtBasicGroundCombatVehicleRecord &	operator= (const DtNetBasicGroundCombatVehicleRecord& orig)
 virtual void	setEntityID (DtU16 value) <i>Set/Get the entity ID</i>
 virtual DtU16	entityID () const
 virtual void	setX_Offset (DtInt8 value) <i>Set/Get the entity location X offset</i>
 virtual DtInt8	X_Offset () const
 virtual void	setY_Offset (DtInt8 value) <i>Set/Get the entity location Y offset</i>
 virtual DtInt8	Y_Offset () const
 virtual void	setZ_Offset (DtInt8 value) <i>Set/Get the entity location Z offset</i>
 virtual DtInt8	Z_Offset () const
 virtual void	setAppearance (DtU32 app) <i>Set/Get appearance (as a 32 bit integer)</i>
 virtual DtU32	appearance () const
 virtual void	setAppearanceAttribute (const DtAppearAttribute attr, const DtAppearAttrVal val) <i>Set/Get appearance by attribute</i>

virtual [DtAppearAttrVal](#)

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual DtInt8

virtual void

virtual void

virtual void

appearanceAttribute (const [DtAppearAttribute](#) attr) const

setPsi (DtInt8 value)
Set/Get the entity orientation psi

psi () const

setTheta (DtInt8 value)
Set/Get the entity orientation theta

theta () const

setPhi (DtInt8 value)
Set/Get the entity orientation phi

phi () const

setEntitySpeed (DtInt8 value)
Set/Get the entity speed

entitySpeed () const

setTurretAzimuth (DtInt8 value)
Set/Get the turret azimuth

turretAzimuth () const

setGunElevation (DtInt8 value)
Set/Get the gun elevation

gunElevation () const

setTurretSlewRate (DtInt8 value)
Set/Get the turret slew rate

turretSlewRate () const

setGunElevationRate (DtInt8 value)
Set/Get the gun elevation rate

gunElevationRate () const

print () const
Print this record

unloadBaseNetRecord (const DtNetBasicGroundCombatVehicleRecord& record)
Load/Unload data from a packet record

loadBaseNetRecord (DtNetBasicGroundCombatVehicleRecord& record)
const

Protected Fields

DtU16 **myEntityID**

DtInt8 **myX_Offset**

DtInt8 **myY_Offset**

DtInt8 **myZ_Offset**

DtU32 **myAppearance**

DtInt8 **myPsi**

DtInt8 **myTheta**

DtInt8 **myPhi**

- DtInt8 **myEntitySpeed**
 - DtInt8 **myTurretAzimuth**
 - DtInt8 **myGunElevation**
 - DtInt8 **myTurretSlewRate**
 - DtInt8 **myGunElevationRate**
-

Documentation

class DtBasicGroundCombatVehicleRecord: Instances of DtBasicGroundCombatVehicleRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

- `DtBasicGroundCombatVehicleRecord()`
Constructors

- `DtBasicGroundCombatVehicleRecord( const DtBasicGroundCombatVehicleRecord& orig)`

- `virtual ~DtBasicGroundCombatVehicleRecord()`
Destructor

- `DtBasicGroundCombatVehicleRecord& operator=( const DtBasicGroundCombatVehicleRecord& orig)`
Assignment operators

- `DtBasicGroundCombatVehicleRecord& operator=( const DtNetBasicGroundCombatVehicleRecord& orig)`

- `virtual void setEntityID(DtU16 value)`
Set/Get the entity ID

- `virtual DtU16 entityID() const`

- `virtual void setX_Offset(DtInt8 value)`
Set/Get the entity location X offset

- `virtual DtInt8 X_Offset() const`

- `virtual void setY_Offset(DtInt8 value)`
Set/Get the entity location Y offset

- `virtual DtInt8 Y_Offset() const`

- `virtual void setZ_Offset(DtInt8 value)`
Set/Get the entity location Z offset

- `virtual DtInt8 Z_Offset() const`

- `virtual void setAppearance(DtU32 app)`
Set/Get appearance (as a 32 bit integer)

- `virtual DtU32 appearance() const`

- virtual void setAppearanceAttribute(const [DtAppearAttribute](#) attr, const [DtAppearAttrVal](#) val)
Set/Get appearance by attribute
- virtual [DtAppearAttrVal](#) appearanceAttribute(const [DtAppearAttribute](#) attr) const
- virtual void setPsi(DtInt8 value)
Set/Get the entity orientation psi
- virtual DtInt8 psi() const
- virtual void setTheta(DtInt8 value)
Set/Get the entity orientation theta
- virtual DtInt8 theta() const
- virtual void setPhi(DtInt8 value)
Set/Get the entity orientation phi
- virtual DtInt8 phi() const
- virtual void setEntitySpeed(DtInt8 value)
Set/Get the entity speed
- virtual DtInt8 entitySpeed() const
- virtual void setTurretAzimuth(DtInt8 value)
Set/Get the turret azimuth
- virtual DtInt8 turretAzimuth() const
- virtual void setGunElevation(DtInt8 value)
Set/Get the gun elevation
- virtual DtInt8 gunElevation() const
- virtual void setTurretSlewRate(DtInt8 value)
Set/Get the turret slew rate
- virtual DtInt8 turretSlewRate() const
- virtual void setGunElevationRate(DtInt8 value)
Set/Get the gun elevation rate
- virtual DtInt8 gunElevationRate() const
- virtual void print() const
Print this record
- virtual void unloadBaseNetRecord(const DtNetBasicGroundCombatVehicleRecord& record)
Load/Unload data from a packet record
- virtual void loadBaseNetRecord(DtNetBasicGroundCombatVehicleRecord& record) const
- DtU16 myEntityID
- DtInt8 myX_Offset

- DtInt8 myY_Offset
- DtInt8 myZ_Offset
- DtU32 myAppearance
- DtInt8 myPsi
- DtInt8 myTheta
- DtInt8 myPhi
- DtInt8 myEntitySpeed
- DtInt8 myTurretAzimuth
- DtInt8 myGunElevation
- DtInt8 myTurretSlewRate
- DtInt8 myGunElevationRate

Direct child classes:

[DtEnhancedGroundCombatVehicleRecord](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file gedenhvehicle.h:

class DtEnhancedGroundCombatVehicleRecord: public DtBasicGroundCombatVehicleRecord

class DtEnhancedGroundCombatVehicleRecord: Instances of DtEnhancedGroundCombatVehicleRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

Inheritance:

DtEnhancedGroundCombatVehicleRecord - DtBasicGroundCombatVehicleRecord

Public Methods

	DtEnhancedGroundCombatVehicleRecord () <i>Constructors</i>
	DtEnhancedGroundCombatVehicleRecord (const DtEnhancedGroundCombatVehicleRecord & orig)
 virtual	~DtEnhancedGroundCombatVehicleRecord () <i>Destructor</i>
 DtEnhancedGroundCombatVehicleRecord &	operator= (const DtEnhancedGroundCombatVehicleRecord & orig) <i>Assignment operators</i>
 DtEnhancedGroundCombatVehicleRecord &	operator= (const DtNetEnhancedGroundCombatVehicleRecord& orig)
 virtual void	setFuelStatus (DtU8 value) <i>Set/Get the fuel status</i>
 virtual DtU8	fuelStatus () const
 virtual void	setGroundMaintenanceStatus (DtU8 value) <i>Set/Get the ground maintenance status</i>
 virtual DtU8	groundMaintenanceStatus () const
 virtual void	setPrimaryAmmunition (DtU8 value) <i>Set/Get the primary ammunition</i>
 virtual DtU8	primaryAmmunition () const
 virtual void	setSecondaryAmmunition (DtU8 value) <i>Set/Get the secondary ammunition</i>
 virtual DtU8	secondaryAmmunition () const
 virtual void	print () const <i>Print this record</i>

virtual void

virtual void

unloadNetRecord (const
DtNetEnhancedGroundCombatVehicleRecord& record)
Load/Unload data from a packet record

loadNetRecord (DtNetEnhancedGroundCombatVehicleRecord&
record) const

Protected Fields

DtU8 **myFuelStatus**

DtU8 **myGroundMaintenanceStatus**

DtU8 **myPrimaryAmmunition**

DtU8 **mySecondaryAmmunition**

Inherited from [DtBasicGroundCombatVehicleRecord](#):

Public Methods

virtual void **setEntityID**(DtU16 value)

virtual DtU16 **entityID**() const

virtual void **setX_Offset**(DtInt8 value)

virtual DtInt8 **X_Offset**() const

virtual void **setY_Offset**(DtInt8 value)

virtual DtInt8 **Y_Offset**() const

virtual void **setZ_Offset**(DtInt8 value)

virtual DtInt8 **Z_Offset**() const

virtual void **setAppearance**(DtU32 app)

virtual DtU32 **appearance**() const

virtual void **setAppearanceAttribute**(const [DtAppearAttribute](#) attr, const [DtAppearAttrVal](#) val)

virtual [DtAppearAttrVal](#) **appearanceAttribute**(const [DtAppearAttribute](#) attr) const

virtual void **setPsi**(DtInt8 value)

virtual DtInt8 **psi**() const

virtual void **setTheta**(DtInt8 value)

virtual DtInt8 **theta**() const

virtual void **setPhi**(DtInt8 value)

virtual DtInt8 **phi**() const

virtual void **setEntitySpeed**(DtInt8 value)

- virtual DtInt8 **entitySpeed**() const
- virtual void **setTurretAzimuth**(DtInt8 value)
- virtual DtInt8 **turretAzimuth**() const
- virtual void **setGunElevation**(DtInt8 value)
- virtual DtInt8 **gunElevation**() const
- virtual void **setTurretSlewRate**(DtInt8 value)
- virtual DtInt8 **turretSlewRate**() const
- virtual void **setGunElevationRate**(DtInt8 value)
- virtual DtInt8 **gunElevationRate**() const
- virtual void **unloadBaseNetRecord**(const DtNetBasicGroundCombatVehicleRecord& record)
- virtual void **loadBaseNetRecord**(DtNetBasicGroundCombatVehicleRecord& record) const

Protected Fields

- DtU16 **myEntityID**
- DtInt8 **myX_Offset**
- DtInt8 **myY_Offset**
- DtInt8 **myZ_Offset**
- DtU32 **myAppearance**
- DtInt8 **myPsi**
- DtInt8 **myTheta**
- DtInt8 **myPhi**
- DtInt8 **myEntitySpeed**
- DtInt8 **myTurretAzimuth**
- DtInt8 **myGunElevation**
- DtInt8 **myTurretSlewRate**
- DtInt8 **myGunElevationRate**

Documentation

class DtEnhancedGroundCombatVehicleRecord: Instances of DtEnhancedGroundCombatVehicleRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

- DtEnhancedGroundCombatVehicleRecord()**

Constructors

• `DtEnhancedGroundCombatVehicleRecord( const DtEnhancedGroundCombatVehicleRecord& orig)`

• `virtual ~DtEnhancedGroundCombatVehicleRecord()`

Destructor

• `DtEnhancedGroundCombatVehicleRecord& operator=( const DtEnhancedGroundCombatVehicleRecord& orig)`

Assignment operators

• `DtEnhancedGroundCombatVehicleRecord& operator=( const DtNetEnhancedGroundCombatVehicleRecord& orig)`

• `virtual void setFuelStatus(DtU8 value)`

Set/Get the fuel status

• `virtual DtU8 fuelStatus() const`

• `virtual void setGroundMaintenanceStatus(DtU8 value)`

Set/Get the ground maintenance status

• `virtual DtU8 groundMaintenanceStatus() const`

• `virtual void setPrimaryAmmunition(DtU8 value)`

Set/Get the primary ammunition

• `virtual DtU8 primaryAmmunition() const`

• `virtual void setSecondaryAmmunition(DtU8 value)`

Set/Get the secondary ammunition

• `virtual DtU8 secondaryAmmunition() const`

• `virtual void print() const`

Print this record

• `virtual void unloadNetRecord( const DtNetEnhancedGroundCombatVehicleRecord& record)`

Load/Unload data from a packet record

• `virtual void loadNetRecord( DtNetEnhancedGroundCombatVehicleRecord& record) const`

• `DtU8 myFuelStatus`

• `DtU8 myGroundMaintenanceStatus`

• `DtU8 myPrimaryAmmunition`

• `DtU8 mySecondaryAmmunition`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file gedlogvehicle.h:

class DtGroundLogisticsVehicleRecord

class DtGroundLogisticsVehicleRecord: Instances of DtGroundLogisticsVehicleRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

Public Methods

	DtGroundLogisticsVehicleRecord () <i>Constructors</i>
	DtGroundLogisticsVehicleRecord (const DtGroundLogisticsVehicleRecord& orig)
virtual	~DtGroundLogisticsVehicleRecord () <i>Destructor</i>
DtGroundLogisticsVehicleRecord &	operator= (const DtGroundLogisticsVehicleRecord& orig) <i>Assignment operators</i>
DtGroundLogisticsVehicleRecord &	operator= (const DtNetGroundLogisticsVehicleRecord& orig)
virtual void	setEntityID (DtU16 value) <i>Set/Get the entity ID</i>
virtual DtU16	entityID () const
virtual void	setX_Offset (DtInt8 value) <i>Set/Get the entity location X offset</i>
virtual DtInt8	X_Offset () const
virtual void	setY_Offset (DtInt8 value) <i>Set/Get the entity location Y offset</i>
virtual DtInt8	Y_Offset () const
virtual void	setZ_Offset (DtInt8 value) <i>Set/Get the entity location Z offset</i>
virtual DtInt8	Z_Offset () const
virtual void	setAppearance (DtU32 app) <i>Set/Get appearance (as a 32 bit integer)</i>
virtual DtU32	appearance () const
virtual void	setAppearanceAttribute (const DtAppearAttribute attr, const DtAppearAttrVal val) <i>Set/Get appearance by attribute</i>
virtual DtAppearAttrVal	appearanceAttribute (const DtAppearAttribute attr) const
virtual void	setPsi (DtInt8 value) <i>Set/Get the entity orientation psi</i>
virtual DtInt8	psi () const
virtual void	setTheta (DtInt8 value) <i>Set/Get the entity orientation theta</i>
virtual DtInt8	theta () const

virtual void	setPhi (DtInt8 value) <i>Set/Get the entity orientation phi</i>
virtual DtInt8	phi () const
virtual void	setEntitySpeed (DtInt8 value) <i>Set/Get the entity speed</i>
virtual DtInt8	entitySpeed () const
virtual void	print () const <i>Print this record</i>
virtual void	unloadNetRecord (const DtNetGroundLogisticsVehicleRecord& record) <i>Load/Unload data from a packet record</i>
virtual void	loadNetRecord (DtNetGroundLogisticsVehicleRecord& record) const

Protected Fields

- DtU16 **myEntityID**
- DtInt8 **myX_Offset**
- DtInt8 **myY_Offset**
- DtInt8 **myZ_Offset**
- DtU32 **myAppearance**
- DtInt8 **myPsi**
- DtInt8 **myTheta**
- DtInt8 **myPhi**
- DtInt8 **myEntitySpeed**

Documentation

class DtGroundLogisticsVehicleRecord: Instances of DtGroundLogisticsVehicleRecord are used to contain the entity state and logistics information about an individual entity within a group of entities

- DtGroundLogisticsVehicleRecord()**
Constructors
- DtGroundLogisticsVehicleRecord(const [DtGroundLogisticsVehicleRecord](#)& orig)**
- virtual ~DtGroundLogisticsVehicleRecord()**
Destructor
- [DtGroundLogisticsVehicleRecord](#)& operator=(const [DtGroundLogisticsVehicleRecord](#)& orig)**
Assignment operators
- [DtGroundLogisticsVehicleRecord](#)& operator=(const DtNetGroundLogisticsVehicleRecord& orig)**
- virtual void setEntityID(DtU16 value)**
Set/Get the entity ID

- `virtual DtU16 entityID() const`
- `virtual void setX_Offset(DtInt8 value)`
Set/Get the entity location X offset
- `virtual DtInt8 X_Offset() const`
- `virtual void setY_Offset(DtInt8 value)`
Set/Get the entity location Y offset
- `virtual DtInt8 Y_Offset() const`
- `virtual void setZ_Offset(DtInt8 value)`
Set/Get the entity location Z offset
- `virtual DtInt8 Z_Offset() const`
- `virtual void setAppearance(DtU32 app)`
Set/Get appearance (as a 32 bit integer)
- `virtual DtU32 appearance() const`
- `virtual void setAppearanceAttribute( const DtAppearAttribute attr, const DtAppearAttrVal val)`
Set/Get appearance by attribute
- `virtual DtAppearAttrVal appearanceAttribute( const DtAppearAttribute attr) const`
- `virtual void setPsi(DtInt8 value)`
Set/Get the entity orientation psi
- `virtual DtInt8 psi() const`
- `virtual void setTheta(DtInt8 value)`
Set/Get the entity orientation theta
- `virtual DtInt8 theta() const`
- `virtual void setPhi(DtInt8 value)`
Set/Get the entity orientation phi
- `virtual DtInt8 phi() const`
- `virtual void setEntitySpeed(DtInt8 value)`
Set/Get the entity speed
- `virtual DtInt8 entitySpeed() const`
- `virtual void print() const`
Print this record
- `virtual void unloadNetRecord( const DtNetGroundLogisticsVehicleRecord& record)`
Load/Unload data from a packet record
- `virtual void loadNetRecord( DtNetGroundLogisticsVehicleRecord& record) const`
- `DtU16 myEntityID`

- DtInt8 myX_Offset
- DtInt8 myY_Offset
- DtInt8 myZ_Offset
- DtU32 myAppearance
- DtInt8 myPsi
- DtInt8 myTheta
- DtInt8 myPhi
- DtInt8 myEntitySpeed

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file intercmsigpdu.h:

class DtIntercomSignalPdu: public DtPdu

Inheritance:

DtIntercomSignalPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtIntercomSignalPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtIntercomSignalPdu (const DtNetIntercomSignalPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtIntercomSignalPdu (const DtIntercomSignalPdu & orig)
virtual	~DtIntercomSignalPdu () <i>destructor</i>
DtIntercomSignalPdu &	operator= (const DtIntercomSignalPdu & orig) <i>Assignment operator</i>
virtual void	setTransEntityID (const DtEntityIdentifier & entity) <i>Set/get transmitting entity identifier</i>
virtual const DtEntityIdentifier &	transEntityID () const
virtual void	setDeviceID (DtU16 intercomID) <i>Set/get intercom ID</i>
virtual DtU16	deviceID () const
virtual void	setEncodingClass (DtEncodingClass encClass) <i>Set/get intercom encoding class</i>
virtual DtEncodingClass	encodingClass () const
virtual void	setEncodingType (DtU16 encType) <i>Set/get intercom encoding type</i>
virtual DtU16	encodingType () const
virtual void	setTDL_Type (DtU16 TDL_Type) <i>Set/get intercom TDL type</i>
virtual DtU16	TDL_Type () const
virtual void	setSampleRate (DtU32 rate) <i>Set/get sample rate</i>
virtual DtU32	sampleRate () const
virtual void	setNumDataBits (DtU16 lengthInBits) <i>Set/get data length in bits</i>

virtual DtU16	numDataBits () const
virtual void	setNumDataBytes (DtU16 lengthInBytes) <i>Set/get data length in bytes</i>
virtual DtU16	numDataBytes () const
virtual void	setNumSamples (DtU16 samples) <i>Set/get number of samples</i>
virtual DtU16	numSamples () const
virtual void	setDataByte (int byteNum, DtU8 data) <i>Set/get a data byte</i>
virtual DtU8	dataByte (int byteNum) const
virtual void	setData (const DtU8* data, int numBytes) <i>Set/get multiple data bytes</i>
virtual void	getData (DtU8* buff) const
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtIntercomSignalPdu</i>
static void	addCallback (DtExerciseConn * conn, DtIntercomSignalPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtIntercomSignalPduCb cb, void* usr)

Protected Fields

static int	theStaticSize <i>Size of PDU with zero data bytes</i>
DtEntityIdentifier	myEntityID <i>Temporary object for reference returns</i>

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
virtual DtNetIntercomSignalPdu*	netPduStruct () const <i>Casts the netPdu element to our network struct</i>
DtIntercomSignalPdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type</i>
virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtIntercomSignalPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Constructors
- `DtIntercomSignalPdu(const DtNetIntercomSignalPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtIntercomSignalPdu(const DtIntercomSignalPdu& orig)`
- `virtual ~DtIntercomSignalPdu()`
destructor
- `DtIntercomSignalPdu& operator=(const DtIntercomSignalPdu& orig)`
Assignment operator
- `virtual void setTransEntityID(const DtEntityIdentifier& entity)`
Set/get transmitting entity identifier
- `virtual const DtEntityIdentifier& transEntityID() const`
- `virtual void setDeviceID(DtU16 intercomID)`
Set/get intercom ID
- `virtual DtU16 deviceID() const`
- `virtual void setEncodingClass(DtEncodingClass encClass)`
Set/get intercom encoding class
- `virtual DtEncodingClass encodingClass() const`
- `virtual void setEncodingType(DtU16 encType)`
Set/get intercom encoding type
- `virtual DtU16 encodingType() const`
- `virtual void setTDL_Type(DtU16 TDL_Type)`
Set/get intercom TDL type
- `virtual DtU16 TDL_Type() const`
- `virtual void setSampleRate(DtU32 rate)`
Set/get sample rate

- `virtual DtU32 sampleRate() const`
- `virtual void setNumDataBits(DtU16 lengthInBits)`
Set/get data length in bits
- `virtual DtU16 numDataBits() const`
- `virtual void setNumDataBytes(DtU16 lengthInBytes)`
Set/get data length in bytes
- `virtual DtU16 numDataBytes() const`
- `virtual void setNumSamples(DtU16 samples)`
Set/get number of samples
- `virtual DtU16 numSamples() const`
- `virtual void setDataByte(int byteNum, DtU8 data)`
Set/get a data byte
- `virtual DtU8 dataByte(int byteNum) const`
- `virtual void setData(const DtU8* data, int numBytes)`
Set/get multiple data bytes. Provide a pointer and data is copied to and from that memory. Check numDataBytes() to make sure that you provide enough room in the buffer.
- `virtual void getData(DtU8* buff) const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtIntercomSignalPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtIntercomSignalPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtIntercomSignalPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetIntercomSignalPdu* netPduStruct() const`
Casts the netPdu element to our network struct
- `DtIntercomSignalPdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- `virtual char* var() const`
Returns a pointer to the first byte of variable length data
- `static int theStaticSize`
Size of PDU with zero data bytes

 [DtEntityIdentifier](#) myEntityID

Temporary object for reference returns

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file intercmctlpdu.h:

class DtIntercomControlPdu: public DtPdu

Inheritance:

DtIntercomControlPdu - DtPdu - DtBaseMsg

Public Methods

	DtIntercomControlPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtIntercomControlPdu (const DtNetIntercomControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtIntercomControlPdu (const DtIntercomControlPdu & orig)
 virtual	~DtIntercomControlPdu () <i>destructor</i>
 DtIntercomControlPdu &	operator= (const DtIntercomControlPdu & orig) <i>Assignment operator</i>
 virtual void	setControlType (DtIntercomCtlType controlType) <i>Set/Get intercom control type</i>
 virtual DtIntercomCtlType	controlType () const
 virtual void	setCommClass (DtIntercomCommClass commClass) <i>Set/Get intercom communications channel class</i>
 virtual DtIntercomCommClass	commClass () const
 virtual void	setCommType (DtIntercomCommType commType) <i>Set/Get intercom communications channel type</i>
 virtual DtIntercomCommType	commType () const
 virtual void	setSourceEntityID (const DtEntityIdentifier & entity) <i>Set/Get source entity identifier</i>
 virtual const DtEntityIdentifier &	sourceEntityID () const
 virtual void	setSourceDeviceID (DtU16 deviceID) <i>Set/Get source communications device ID</i>
 virtual DtU16	sourceDeviceID () const
 virtual void	setSourceLineID (DtU8 lineID) <i>Set/Get source line ID</i>
 virtual DtU8	sourceLineID () const
 virtual void	setTransmitPriority (DtU8 transmitPriority) <i>Set/Get transmit priority</i>

virtual DtU8	transmitPriority () const
virtual void	setTransmitLineState (DtIntercomTransLineState lineState) <i>Set/Get transmit line state</i>
virtual DtIntercomTransLineState	transmitLineState () const
virtual void	setCmdType (DtIntercomCmdType cmdType) <i>Set/Get intercom command type</i>
virtual DtIntercomCmdType	cmdType () const
virtual void	setMasterEntityID (const DtEntityIdentifier & entity) <i>Set/Get master entity identifier</i>
virtual const DtEntityIdentifier &	masterEntityID () const
virtual void	setMasterDeviceID (DtU16 deviceID) <i>Set/Get master communications device ID</i>
virtual DtU16	masterDeviceID () const
virtual void	setMasterChannelID (DtU16 channelID) <i>Set/Get master channel ID</i>
virtual DtU16	masterChannelID () const
virtual DtU32	parmRecordsLength () const <i>Get the parameters length</i>
virtual DtBoolean	setNumParmRecords (int numRecords) <i>Set/Get number of parameter records (this is not a PDU field)</i>
virtual int	numParmRecords () const
virtual DtBoolean	getRecordType (int parmIndex, DtIntercomCtlRecType & recordType) const <i>Get the record type for parameter record parmIndex</i>
virtual DtBoolean	setParmRecord (int parmIndex, const DtIntercomEntityDestRec & init) <i>Set/Get the parameter record type and values of record parmIndex to values specified by the initializer in the second argument</i>
virtual DtBoolean	getParmRecord (int parmIndex, DtIntercomEntityDestRec & record) const
virtual DtBoolean	setParmRecord (int parmIndex, const DtIntercomGroupDestRec & init)
virtual DtBoolean	getParmRecord (int parmIndex, DtIntercomGroupDestRec & record) const
virtual DtBoolean	setParmRecord (int parmIndex, const DtIntercomGroupAsgmtRec & init)
virtual DtBoolean	getParmRecord (int parmIndex, DtIntercomGroupAsgmtRec & record) const
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtIntercomControlPdu</i>
static void	addCallback (DtExerciseConn * conn, DtIntercomControlPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtIntercomControlPduCb cb, void* usr)

Protected Fields

- static int **theStaticSize**
Size of PDU with zero parameter records
- int **myNumParmRecords**
Number of parameter records in this PDU
- [DtEntityIdentifier](#) **myEntityID**
Temporary object for reference returns

Protected Methods

- virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind
 - virtual DtNetIntercomControlPdu* **netPduStruct** () const
Casts the netPdu element to our network struct
 - [DtIntercomControlPdu](#)* **self** () const
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
 - virtual char* **var** () const
Returns a pointer to the first byte of variable length data
 - virtual int **recordOffset** (int recordIndex) const
Returns the offset (in bytes) of the given parameter record
 - virtual void **resizeRecord** (DtNetIntercomParmRec* rec, int newSize)
Resize a given parameter record to the size indicated
-

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const

- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtIntercomControlPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtIntercomControlPdu**(const DtNetIntercomControlPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtIntercomControlPdu**(const [DtIntercomControlPdu](#)& orig)
- virtual ~DtIntercomControlPdu**()
destructor

`DtIntercomControlPdu& operator=(const DtIntercomControlPdu& orig)`
Assignment operator

`virtual void setControlType(DtIntercomCtlType controlType)`
Set/Get intercom control type

`virtual DtIntercomCtlType controlType() const`

`virtual void setCommClass(DtIntercomCommClass commClass)`
Set/Get intercom communications channel class

`virtual DtIntercomCommClass commClass() const`

`virtual void setCommType(DtIntercomCommType commType)`
Set/Get intercom communications channel type

`virtual DtIntercomCommType commType() const`

`virtual void setSourceEntityID(const DtEntityIdentifier& entity)`
Set/Get source entity identifier

`virtual const DtEntityIdentifier& sourceEntityID() const`

`virtual void setSourceDeviceID(DtU16 deviceID)`
Set/Get source communications device ID

`virtual DtU16 sourceDeviceID() const`

`virtual void setSourceLineID(DtU8 lineID)`
Set/Get source line ID

`virtual DtU8 sourceLineID() const`

`virtual void setTransmitPriority(DtU8 transmitPriority)`
Set/Get transmit priority

`virtual DtU8 transmitPriority() const`

`virtual void setTransmitLineState(DtIntercomTransLineState lineState)`
Set/Get transmit line state

`virtual DtIntercomTransLineState transmitLineState() const`

`virtual void setCmdType(DtIntercomCmdType cmdType)`
Set/Get intercom command type

`virtual DtIntercomCmdType cmdType() const`

`virtual void setMasterEntityID(const DtEntityIdentifier& entity)`
Set/Get master entity identifier

`virtual const DtEntityIdentifier& masterEntityID() const`

`virtual void setMasterDeviceID(DtU16 deviceID)`
Set/Get master communications device ID

- `virtual DtU16 masterDeviceID() const`
- `virtual void setMasterChannelID(DtU16 channelID)`
Set/Get master channel ID
- `virtual DtU16 masterChannelID() const`
- `virtual DtU32 parmRecordsLength() const`
Get the parameters length. This field is automatically changed by the functions: `setNumParmRecords()`, and `setParmRecord()`
- `virtual DtBoolean setNumParmRecords(int numRecords)`
Set/Get number of parameter records (this is not a PDU field). If the required space would exceed the maximum space permitted for a PDU, then `DtFalse` may be returned.
- `virtual int numParmRecords() const`
- `virtual DtBoolean getRecordType(int parmIndex, DtIntercomCtlRecType& recordType) const`
Get the record type for parameter record `parmIndex`. If `parmIndex` is out of range or the parameter is invalid, then `DtFalse` is returned.
- `virtual DtBoolean setParmRecord(int parmIndex, const DtIntercomEntityDestRec& init)`
Set/Get the parameter record type and values of record `parmIndex` to values specified by the initializer in the second argument. If `parmIndex` is out of bounds, or if the record type does not match the function (for a get), then `DtFalse` is returned. If the space would exceed the maximum space permitted for a PDU (for a set) then `DtFalse` may be returned.
- `virtual DtBoolean getParmRecord(int parmIndex, DtIntercomEntityDestRec& record) const`
- `virtual DtBoolean setParmRecord(int parmIndex, const DtIntercomGroupDestRec& init)`
- `virtual DtBoolean getParmRecord(int parmIndex, DtIntercomGroupDestRec& record) const`
- `virtual DtBoolean setParmRecord(int parmIndex, const DtIntercomGroupAsgmtRec& init)`
- `virtual DtBoolean getParmRecord(int parmIndex, DtIntercomGroupAsgmtRec& record) const`
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a `DtIntercomControlPdu`. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtIntercomControlPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtIntercomControlPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetIntercomControlPdu* netPduStruct() const`
Casts the `netPdu` element to our network struct

• [DtIntercomControlPdu](#)* self() const

Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type

• virtual char* var() const

Returns a pointer to the first byte of variable length data

• virtual int recordOffset(int recordIndex) const

Returns the offset (in bytes) of the given parameter record

• virtual void resizeRecord(DtNetIntercomParmRec* rec, int newSize)

Resize a given parameter record to the size indicated

• static int theStaticSize

Size of PDU with zero parameter records

• int myNumParmRecords

Number of parameter records in this PDU

• [DtEntityIdentifier](#) myEntityID

Temporary object for reference returns

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtIntercomCtlType

Intercom Control PDU enums

-  **DtIntercomCtlStatus**
 -  **DtIntercomReqAck**
 -  **DtIntercomReqNoAck**
 -  **DtIntercomAck**
 -  **DtIntercomNack**
-

Documentation

Intercom Control PDU enums

-  **DtIntercomCtlStatus**
-  **DtIntercomReqAck**
-  **DtIntercomReqNoAck**
-  **DtIntercomAck**
-  **DtIntercomNack**

In file disenums.h:

enum DtIntercomCommClass

 DtIntercomSimCommChnl

 DtIntercomSimCommSupChnl

Documentation

 DtIntercomSimCommChnl

 DtIntercomSimCommSupChnl

In file disenums.h:

enum DtIntercomCommType

-
-  DtIntercomFDX
 -  DtIntercomHDX_RecOnly
 -  DtIntercomHDX_TransOnly
 -  DtIntercomHDX
-

Documentation

-  DtIntercomFDX
-  DtIntercomHDX_RecOnly
-  DtIntercomHDX_TransOnly
-  DtIntercomHDX

In file disenums.h:

enum DtIntercomTransLineState

-
-  DtIntercomNotApplicable
 -  DtIntercomNotTransmitting
 -  DtIntercomTransmitting
-

Documentation

-  DtIntercomNotApplicable
-  DtIntercomNotTransmitting
-  DtIntercomTransmitting

In file disenums.h:

enum DtIntercomCmdType

-
-  DtIntercomNoCmd
 -  DtIntercomCmdStatus
 -  DtIntercomConnect
 -  DtIntercomDisconnect
 -  DtIntercomReset
 -  DtIntercomOn
 -  DtIntercomOff
-

Documentation

-  DtIntercomNoCmd
-  DtIntercomCmdStatus
-  DtIntercomConnect
-  DtIntercomDisconnect
-  DtIntercomReset
-  DtIntercomOn
-  DtIntercomOff

In file disenums.h:

enum DtIntercomCtlRecType

-
-  DtIntercomEntDestRec
 -  DtIntercomGrpDestRec
 -  DtIntercomGrpAsgmtRec
-

Documentation

-  DtIntercomEntDestRec
-  DtIntercomGrpDestRec
-  DtIntercomGrpAsgmtRec

In file intercomentdst.h:

class DtIntercomEntityDestRec: public DtIntercomParmRec

class DtIntercomEntityDestRec: Instances of DtIntercomEntityDestRec are used to contain the information in the entity destination parameter records used by the intercom control PDU class

Inheritance:

DtIntercomEntityDestRec - DtIntercomParmRec

Public Methods

	DtIntercomEntityDestRec () <i>Constructors</i>
	DtIntercomEntityDestRec (const DtIntercomEntityDestRec & orig)
 virtual	~DtIntercomEntityDestRec () <i>Destructor</i>
 DtIntercomEntityDestRec &	operator= (const DtIntercomEntityDestRec & orig) <i>Assignment operators</i>
 DtIntercomEntityDestRec &	operator= (const DtNetIntercomEntityDestRec& orig)
 virtual void	setDestEntityID (const DtEntityIdentifier & entityID) <i>Set/Get the destination entity identifier</i>
 virtual const DtEntityIdentifier &	destEntityID () const
 virtual void	setDestDeviceID (DtU16 deviceID) <i>Set/Get the destination device ID</i>
 virtual DtU16	destDeviceID () const
 virtual void	setDestLineID (DtU8 lineID) <i>Set/Get the destination line ID</i>
 virtual DtU8	destLineID () const
 virtual void	setDestPriority (DtU8 priority) <i>Set/Get the destination priority</i>
 virtual DtU8	destPriority () const
 virtual void	setDestLineStateCmd (DtIntercomDestLineStateCmd cmd) <i>Set/Get the destination line state command</i>
 virtual DtIntercomDestLineStateCmd	destLineStateCmd () const
 virtual void	print () const <i>Print this record</i>

virtual void

unloadNetRecord (const DtNetIntercomEntityDestRec& record)
Load/Unload data from a packet record

virtual void

loadNetRecord (DtNetIntercomEntityDestRec& record) const

Protected Fields

[DtEntityIdentifier](#)

myDestEntityID

DtU16

myDestDeviceID

DtU8

myDestLineID

DtU8

myDestPriority

[DtIntercomDestLineStateCmd](#)

myDestLineStateCmd

Inherited from [DtIntercomParmRec](#):

Public Methods

virtual [DtIntercomCtlRecType](#) **recordType**() const

virtual DtU16 **recordLength**() const

Protected Fields

[DtIntercomCtlRecType](#) myRecordType

DtU16 myRecordLength

Documentation

class DtIntercomEntityDestRec: Instances of DtIntercomEntityDestRec are used to contain the information in the entity destination parameter records used by the intercom control PDU class

DtIntercomEntityDestRec()

Constructors

DtIntercomEntityDestRec(const [DtIntercomEntityDestRec](#)& orig)

virtual ~DtIntercomEntityDestRec()

Destructor

[DtIntercomEntityDestRec](#)& **operator=**(const [DtIntercomEntityDestRec](#)& orig)

Assignment operators

[DtIntercomEntityDestRec](#)& **operator=**(const DtNetIntercomEntityDestRec& orig)

virtual void setDestEntityID(const [DtEntityIdentifier](#)& entityID)

Set/Get the destination entity identifier

virtual const [DtEntityIdentifier](#)& destEntityID() const

virtual void setDestDeviceID(DtU16 deviceID)

Set/Get the destination device ID

virtual DtU16 destDeviceID() const

virtual void setDestLineID(DtU8 lineID)

Set/Get the destination line ID

virtual DtU8 destLineID() const

virtual void setDestPriority(DtU8 priority)

Set/Get the destination priority

virtual DtU8 destPriority() const

virtual void setDestLineStateCmd([DtIntercomDestLineStateCmd](#) cmd)

Set/Get the destination line state command

virtual [DtIntercomDestLineStateCmd](#) destLineStateCmd() const

virtual void print() const

Print this record

virtual void unloadNetRecord(const DtNetIntercomEntityDestRec& record)

Load/Unload data from a packet record

virtual void loadNetRecord(DtNetIntercomEntityDestRec& record) const

[DtEntityIdentifier](#) myDestEntityID

DtU16 myDestDeviceID

DtU8 myDestLineID

DtU8 myDestPriority

[DtIntercomDestLineStateCmd](#) myDestLineStateCmd

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file intercmparrec.h:

class DtIntercomParmRec

class DtIntercomParmRec: DtIntercomParmRec is an abstract base class for the parameter record classes which are used by the intercom control PDU (DtIntercomControlPdu)

Inheritance:

DtIntercomParmRec

Public Methods

- virtual ~DtIntercomParmRec ()
Destructor
- virtual [DtIntercomCtlRecType](#) **recordType** () const
Get the record type
- virtual DtU16 **recordLength** () const
Get the record length
- virtual void **print** () const
Print the record type and length

Protected Fields

- [DtIntercomCtlRecType](#) **myRecordType**
Data members:
- DtU16 **myRecordLength**

Protected Methods

- DtIntercomParmRec** ()
Constructors
 - DtIntercomParmRec** (const [DtIntercomParmRec](#)& orig)
 - DtIntercomParmRec** ([DtIntercomCtlRecType](#) recordType, int recordLength)
 - [DtIntercomParmRec](#)& **operator=** (const [DtIntercomParmRec](#)& orig)
Assignment operator
-

Documentation

class DtIntercomParmRec: DtIntercomParmRec is an abstract base class for the parameter record classes which are used by the intercom control PDU (DtIntercomControlPdu)

- virtual ~DtIntercomParmRec()
Destructor
 - virtual [DtIntercomCtlRecType](#) recordType() const
Get the record type
 - virtual DtU16 recordLength() const
Get the record length
 - virtual void print() const
Print the record type and length
 - DtIntercomParmRec()
Constructors
 - DtIntercomParmRec(const [DtIntercomParmRec](#)& orig)
 - DtIntercomParmRec([DtIntercomCtlRecType](#) recordType, int recordLength)
 - [DtIntercomParmRec](#)& operator=(const [DtIntercomParmRec](#)& orig)
Assignment operator
 - [DtIntercomCtlRecType](#) myRecordType
Data members:
 - DtU16 myRecordLength
-

Direct child classes:

[DtIntercomGroupDestRec](#)
[DtIntercomGroupAsgmtRec](#)
[DtIntercomEntityDestRec](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file intercmgrpdst.h:

class DtIntercomGroupDestRec: public DtIntercomParmRec

class DtIntercomGroupDestRec: Instances of DtIntercomGroupDestRec are used to contain the information in the group destination parameter record used by the intercom control PDU class

Inheritance:

DtIntercomGroupDestRec - DtIntercomParmRec

Public Methods

	DtIntercomGroupDestRec () <i>Constructors</i>
	DtIntercomGroupDestRec (const DtIntercomGroupDestRec & orig)
virtual	~DtIntercomGroupDestRec () <i>Destructor</i>
DtIntercomGroupDestRec &	operator= (const DtIntercomGroupDestRec & orig) <i>Assignment operators</i>
DtIntercomGroupDestRec &	operator= (const DtNetIntercomGroupDestRec& orig)
virtual void	setDestGroupBitField (DtU32 bitField) <i>Set/Get the group bit field</i>
virtual DtU32	destGroupBitField () const
virtual void	setDestPriority (DtU8 priority) <i>Set/Get the destination priority</i>
virtual DtU8	destPriority () const
virtual void	setDestLineStateCmd (DtIntercomDestLineStateCmd cmd) <i>Set/Get the destination line state command</i>
virtual DtIntercomDestLineStateCmd	destLineStateCmd () const
virtual void	print () const <i>Print this record</i>
virtual void	unloadNetRecord (const DtNetIntercomGroupDestRec& record) <i>Load/Unload data from a packet record</i>
virtual void	loadNetRecord (DtNetIntercomGroupDestRec& record) const

Protected Fields

- DtU32 myDestGroupBitField
 - DtU8 myDestPriority
 - [DtIntercomDestLineStateCmd](#) myDestLineStateCmd
-

Inherited from [DtIntercomParmRec](#):

Public Methods

- virtual [DtIntercomCtlRecType](#) recordType() const
- virtual DtU16 recordLength() const

Protected Fields

- [DtIntercomCtlRecType](#) myRecordType
 - DtU16 myRecordLength
-

Documentation

class DtIntercomGroupDestRec: Instances of DtIntercomGroupDestRec are used to contain the information in the group destination parameter record used by the intercom control PDU class

- DtIntercomGroupDestRec ()
Constructors
- DtIntercomGroupDestRec (const [DtIntercomGroupDestRec](#)& orig)
- virtual ~DtIntercomGroupDestRec ()
Destructor
- [DtIntercomGroupDestRec](#)& operator=(const [DtIntercomGroupDestRec](#)& orig)
Assignment operators
- [DtIntercomGroupDestRec](#)& operator=(const DtNetIntercomGroupDestRec& orig)
- virtual void setDestGroupBitField(DtU32 bitField)
Set/Get the group bit field
- virtual DtU32 destGroupBitField() const
- virtual void setDestPriority(DtU8 priority)
Set/Get the destination priority
- virtual DtU8 destPriority() const

- virtual void setDestLineStateCmd([DtIntercomDestLineStateCmd](#) cmd)
Set/Get the destination line state command
 - virtual [DtIntercomDestLineStateCmd](#) destLineStateCmd() const
 - virtual void print() const
Print this record
 - virtual void unloadNetRecord(const DtNetIntercomGroupDestRec& record)
Load/Unload data from a packet record
 - virtual void loadNetRecord(DtNetIntercomGroupDestRec& record) const
 - DtU32 myDestGroupBitField
 - DtU8 myDestPriority
 - [DtIntercomDestLineStateCmd](#) myDestLineStateCmd
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtIntercomDestLineStateCmd

-
- DtIntercomLS_NoCmd
 - DtIntercomSetLS_Trans
 - DtIntercomSetLS_NotTrans
 - DtIntercomRetLocLS_Ctl
-

Documentation

- DtIntercomLS_NoCmd
- DtIntercomSetLS_Trans
- DtIntercomSetLS_NotTrans
- DtIntercomRetLocLS_Ctl

In file intercmgrpasg.h:

class DtIntercomGroupAsgmtRec: public DtIntercomParmRec

class DtIntercomGroupAsgmtRec: Instances of DtIntercomGroupAsgmtRec are used to contain the information in the group assignment parameter record used by the intercom control PDU class

Inheritance:

DtIntercomGroupAsgmtRec - DtIntercomParmRec

Public Methods

	DtIntercomGroupAsgmtRec () <i>Constructors</i>
	DtIntercomGroupAsgmtRec (const DtIntercomGroupAsgmtRec & orig)
 virtual	~DtIntercomGroupAsgmtRec () <i>Destructor</i>
 DtIntercomGroupAsgmtRec &	operator= (const DtIntercomGroupAsgmtRec & orig) <i>Assignment operators</i>
 DtIntercomGroupAsgmtRec &	operator= (const DtNetIntercomGroupAsgmtRec& orig)
 virtual void	setGroupBitField (DtU32 bitField) <i>Set/Get the group bit field</i>
 virtual DtU32	groupBitField () const
 virtual void	setDestEntityID (const DtEntityIdentifier & entityID) <i>Set/Get the destination entity identifier</i>
 virtual const DtEntityIdentifier &	destEntityID () const
 virtual void	setDestEquipID (DtU16 equipID) <i>Set/Get the destination equipment ID</i>
 virtual DtU16	destEquipID () const
 virtual void	setDestLineID (DtU8 lineID) <i>Set/Get the destination line ID</i>
 virtual DtU8	destLineID () const
 virtual void	print () const <i>Print this record</i>
 virtual void	unloadNetRecord (const DtNetIntercomGroupAsgmtRec& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetIntercomGroupAsgmtRec& record) const

Protected Fields

- `DtU32 myGroupBitField`
 - `DtEntityIdentifier myDestEntityID`
 - `DtU16 myDestEquipID`
 - `DtU8 myDestLineID`
-

Inherited from [DtIntercomParmRec](#):

Public Methods

- virtual `DtIntercomCtlRecType recordType()` const
- virtual `DtU16 recordLength()` const

Protected Fields

- `DtIntercomCtlRecType myRecordType`
 - `DtU16 myRecordLength`
-

Documentation

class DtIntercomGroupAsgmtRec: Instances of DtIntercomGroupAsgmtRec are used to contain the information in the group assignment parameter record used by the intercom control PDU class

- `DtIntercomGroupAsgmtRec()`
Constructors
- `DtIntercomGroupAsgmtRec(const DtIntercomGroupAsgmtRec& orig)`
- `virtual ~DtIntercomGroupAsgmtRec()`
Destructor
- `DtIntercomGroupAsgmtRec& operator=(const DtIntercomGroupAsgmtRec& orig)`
Assignment operators
- `DtIntercomGroupAsgmtRec& operator=(const DtNetIntercomGroupAsgmtRec& orig)`
- `virtual void setGroupBitField(DtU32 bitField)`
Set/Get the group bit field
- `virtual DtU32 groupBitField() const`
- `virtual void setDestEntityID(const DtEntityIdentifier& entityID)`
Set/Get the destination entity identifier

- virtual const [DtEntityIdentifier](#)& destEntityID() const
- virtual void setDestEquipID(DtU16 equipID)
Set/Get the destination equipment ID
- virtual DtU16 destEquipID() const
- virtual void setDestLineID(DtU8 lineID)
Set/Get the destination line ID
- virtual DtU8 destLineID() const
- virtual void print() const
Print this record
- virtual void unloadNetRecord(const DtNetIntercomGroupAsgmtRec& record)
Load/Unload data from a packet record
- virtual void loadNetRecord(DtNetIntercomGroupAsgmtRec& record) const
- DtU32 myGroupBitFields
- [DtEntityIdentifier](#) myDestEntityID
- DtU16 myDestEquipID
- DtU8 myDestLineID

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file iffatcnavpdu.h:

class DtlffAtcNavaisPdu: public DtPdu

Inheritance:

DtlffAtcNavaisPdu - DtPdu - DtBaseMsg

Public Methods

	DtlffAtcNavaisPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtlffAtcNavaisPdu (const DtNetlffAtcNavaisPduLayer1* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtlffAtcNavaisPdu (const <u>DtlffAtcNavaisPdu</u> & orig)
 virtual	~DtlffAtcNavaisPdu () <i>destructor</i>
 static <u>DtPdu</u> *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtlffAtcNavaisPdu</i>
 static void	addCallback (<u>DtExerciseConn</u> * conn, DtlffAtcNavaisPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
 static void	removeCallback (<u>DtExerciseConn</u> * conn, DtlffAtcNavaisPduCb cb, void* usr)
 <u>DtlffAtcNavaisPdu</u> &	operator= (const <u>DtlffAtcNavaisPdu</u> & orig) <i>Assignment operator</i>
 virtual void	setEmittingEntityID (const <u>DtEntityIdentifier</u> & entity) <i>Set/Get emitting entity identifier</i>
 virtual const <u>DtEntityIdentifier</u> &	emittingEntityID () const
 virtual void	setEventID (const DtEventID& event) <i>Set/Get event identifier</i>
 virtual const DtEventID&	eventID () const
 virtual void	setLocation (DtConstVector coords) <i>Set/Get location</i>
 virtual DtConstVector	location () const
 virtual void	setSystemID (const <u>DtlffAtcNavaisSystemID</u> & id) <i>Set/Get system ID</i>
 virtual const <u>DtlffAtcNavaisSystemID</u> &	systemID () const

- virtual void
- virtual const [DtIffAtcNav aidsOperational](#)&
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual const [DtIffAtcNav aidsLayerHeader](#)&
- virtual void
- virtual const [DtIffAtcNav aidsBeamData](#)&
- virtual void
- virtual const [DtIffAtcNav aidsSecondOperational](#)&
- virtual DtU16
- virtual [DtBoolean](#)
- virtual [DtBoolean](#)
- virtual void

setOperational (const [DtIffAtcNav aidsOperational](#)& oper)
Set/Get fundamental operational data

operational () const

layerPresent (int layer) const
Query the layers field of the fundamental operational data record

enableLayer2 (DtU8 layerSpecificInfo, int numRecords)
Enable/Disable layer 2

disableLayer2 ()

layer2Header () const
Get layer header for layer 2 This record is set automatically by the PDU class

setBeamData (const [DtIffAtcNav aidsBeamData](#)& beamData)
Set/Get beam data

beamData () const

setSecondaryOperational (const [DtIffAtcNav aidsSecondOperational](#)& oper)
Set/Get secondary operational parameters

secondaryOperational () const

numParameterSets () const
Get the number of fundamental parameter data records from the secondary operational parameters record

setParmData (int parmIndex, const [DtIffAtcNav aidsParameter](#)& init)
Set/Get the fundamental parameter data records for the second layer

getParmData (int parmIndex, [DtIffAtcNav aidsParameter](#)& record)
 const

printData () const
Print the data contained by this PDU

Public

- Note: the following accessor functions are for records in layer 2

Protected Fields

- static int
- [DtEntityIdentifier](#)
- DtEventID
- [DtVector](#)
- [DtIffAtcNav aidsSystemID](#)
- [DtIffAtcNav aidsOperational](#)
- [DtIffAtcNav aidsLayerHeader](#)
- theStaticSize**
Size of PDU with zero parameter records
- myEntityID**
Temporary objects for reference returns
- myEventID**
- myEntCoords**
- mySystemID**
- myOperational**
- myLayerHeader**

- [DtIffAtcNav aidsBeamData](#) **myBeamData**
- [DtIffAtcNav aidsSecondOperational](#) **mySecondOperational**
- [DtIffAtcNav aidsParameter](#) **myParameter**

Protected Methods

- virtual [DtPduKind](#) **internalGetPduKind** () const
Returns the internal PDU kind
- virtual DtNetIffAtcNav aidsPduLayer1* **netPduStruct** () const
Casts the netPdu element to our network struct
- virtual DtNetIffAtcNav aidsPduLayer2* **netPduStructL2** () const
Casts the netPdu element to our network struct
- [DtIffAtcNav aidsPdu](#)* **self** () const
Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- virtual DtNetIffAtcNav aidsParameter* **parameter** () const
Returns a pointer to the first parameter record (after static data)

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const

- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const

- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**

- void* **myInternalBuff**

- int **myIntBuffSize**

- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)

- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)

- virtual void **changeSize**(int size)

- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)

- virtual void **deleteBytes**(int offset, int numBytes)

- virtual void **deleteBytes**(void *address, int numBytes)

- virtual void **clearAll**()

- virtual void* **newInternalBuffer**(int size)

Documentation

- DtIffAtcNav aidsPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Constructors

- DtIffAtcNav aidsPdu(const DtNetIffAtcNav aidsPduLayer1* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

- DtIffAtcNav aidsPdu(const [DtIffAtcNav aidsPdu](#)& orig)

- virtual ~DtIffAtcNav aidsPdu()

destructor

- static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtIffAtcNav aidsPdu. Caller is responsible for deletion.

- static void addCallback([DtExerciseConn](#)* conn, DtIffAtcNav aidsPduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

- static void removeCallback([DtExerciseConn](#)* conn, DtIffAtcNav aidsPduCb cb, void* usr)

-  `DtIffAtcNav aidsPdu& operator=(const DtIffAtcNav aidsPdu& orig)`
 Assignment operator
-  `virtual void setEmittingEntityID(const DtEntityIdentifier& entity)`
 Set/Get emitting entity identifier
-  `virtual const DtEntityIdentifier& emittingEntityID() const`
-  `virtual void setEventID(const DtEventID& event)`
 Set/Get event identifier
-  `virtual const DtEventID& eventID() const`
-  `virtual void setLocation(DtConstVector coords)`
 Set/Get location
-  `virtual DtConstVector location() const`
-  `virtual void setSystemID(const DtIffAtcNav aidsSystemID& id)`
 Set/Get system ID
-  `virtual const DtIffAtcNav aidsSystemID& systemID() const`
-  `virtual void setOperational(const DtIffAtcNav aidsOperational& oper)`
 Set/Get fundamental operational data
-  `virtual const DtIffAtcNav aidsOperational& operational() const`
-  `virtual DtBoolean layerPresent(int layer) const`
 Query the layers field of the fundamental operational data record
-  `virtual DtBoolean enableLayer2(DtU8 layerSpecificInfo, int numRecords)`
 Enable/Disable layer 2. This method is also used to select the number of fundamental parameter data records for the second layer.
-  `virtual DtBoolean disableLayer2()`
- 

Note: the following accessor functions are for records in layer 2. If layer 2 is not enabled, the sets will do nothing and the gets will return undefined results.
-  `virtual const DtIffAtcNav aidsLayerHeader& layer2Header() const`
 Get layer header for layer 2 This record is set automatically by the PDU class
-  `virtual void setBeamData(const DtIffAtcNav aidsBeamData& beamData)`
 Set/Get beam data
-  `virtual const DtIffAtcNav aidsBeamData& beamData() const`
-  `virtual void setSecondaryOperational( const DtIffAtcNav aidsSecondOperational& oper)`
 Set/Get secondary operational parameters
-  `virtual const DtIffAtcNav aidsSecondOperational& secondaryOperational() const`
-  `virtual DtU16 numParameterSets() const`

Get the number of fundamental parameter data records from the secondary operational parameters record

virtual [DtBoolean](#) setParmData(int parmIndex, const [DtIffAtcNav aidsParameter](#)& init)

Set/Get the fundamental parameter data records for the second layer

virtual [DtBoolean](#) getParmData(int parmIndex, [DtIffAtcNav aidsParameter](#)& record) const

virtual void printData() const

Print the data contained by this PDU

virtual [DtPduKind](#) internalGetPduKind() const

Returns the internal PDU kind

virtual DtNetIffAtcNav aidsPduLayer1* netPduStruct() const

Casts the netPdu element to our network struct

virtual DtNetIffAtcNav aidsPduLayer2* netPduStructL2() const

Casts the netPdu element to our network struct

[DtIffAtcNav aidsPdu](#)* self() const

Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type

virtual DtNetIffAtcNav aidsParameter* parameter() const

Returns a pointer to the first parameter record (after static data)

static int theStaticSize

Size of PDU with zero parameter records

[DtEntityIdentifier](#) myEntityID

Temporary objects for reference returns

DtEventID myEventID

[DtVector](#) myEntCoords

[DtIffAtcNav aidsSystemID](#) mySystemID

[DtIffAtcNav aidsOperational](#) myOperational

[DtIffAtcNav aidsLayerHeader](#) myLayerHeader

[DtIffAtcNav aidsBeamData](#) myBeamData

[DtIffAtcNav aidsSecondOperational](#) mySecondOperational

[DtIffAtcNav aidsParameter](#) myParameter

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file iffatcnasysid.h:

class DtlffAtcNavaisSystemID

class DtlffAtcNavaisSystemID: DtlffAtcNavaisSystemID is a class which contains information identifying the emitting system in the IFF/ATC/NAVAIDS PDU

Public Methods

	DtlffAtcNavaisSystemID () <i>Constructors</i>
	DtlffAtcNavaisSystemID (const DtlffAtcNavaisSystemID & orig)
DtlffAtcNavaisSystemID &	operator= (const DtlffAtcNavaisSystemID & orig) <i>Assignment operator</i>
DtlffAtcNavaisSystemID &	operator= (const DtNetlffAtcNavaisSystemID& orig)
virtual	~DtlffAtcNavaisSystemID () <i>Destructor</i>
virtual void	setSystemType (DtlffAtcNavaisSysType type) <i>Set/Get the system type</i>
virtual DtlffAtcNavaisSysType	systemType () const
virtual void	setSystemName (DtlffAtcNavaisSysName name) <i>Set/Get the system name</i>
virtual DtlffAtcNavaisSysName	systemName () const
virtual void	setSystemMode (DtlffAtcNavaisSysMode mode) <i>Set/Get the system mode</i>
virtual DtlffAtcNavaisSysMode	systemMode () const
virtual void	setChangeOptions (DtU8 opts) <i>Set/Get the system change/options</i>
virtual DtU8	changeOptions () const
virtual void	setChangeOptionsBit (DtlffAtcNavaisChangeOptionsBit bit, DtBoolean value) <i>Set/Get the system change/option by bit value</i>
virtual DtBoolean	changeOptionsBit (DtlffAtcNavaisChangeOptionsBit bit) const
virtual void	print () const <i>Print the information contained in the record</i>
virtual void	unloadNetRecord (const DtNetlffAtcNavaisSystemID& record) <i>Load/Unload data from a packet record</i>
virtual void	loadNetRecord (DtNetlffAtcNavaisSystemID& record) const

Protected Fields

- [DtIffAtcNav aidsSysType](#) myType
Data members:
 - [DtIffAtcNav aidsSysName](#) myName
 - [DtIffAtcNav aidsSysMode](#) myMode
 - DtU8 myChangeOptions
-

Documentation

class DtIffAtcNav aidsSystemID: DtIffAtcNav aidsSystemID is a class which contains information identifying the emitting system in the IFF/ATC/NAVAIDS PDU

- DtIffAtcNav aidsSystemID ()
Constructors
- DtIffAtcNav aidsSystemID (const [DtIffAtcNav aidsSystemID](#)& orig)
- [DtIffAtcNav aidsSystemID](#)& operator=(const [DtIffAtcNav aidsSystemID](#)& orig)
Assignment operator
- [DtIffAtcNav aidsSystemID](#)& operator=(const DtNetIffAtcNav aidsSystemID& orig)
- virtual ~DtIffAtcNav aidsSystemID ()
Destructor
- virtual void setSystemType ([DtIffAtcNav aidsSysType](#) type)
Set/Get the system type
- virtual [DtIffAtcNav aidsSysType](#) systemType () const
- virtual void setSystemName ([DtIffAtcNav aidsSysName](#) name)
Set/Get the system name
- virtual [DtIffAtcNav aidsSysName](#) systemName () const
- virtual void setSystemMode ([DtIffAtcNav aidsSysMode](#) mode)
Set/Get the system mode
- virtual [DtIffAtcNav aidsSysMode](#) systemMode () const
- virtual void setChangeOptions (DtU8 opts)
Set/Get the system change/options
- virtual DtU8 changeOptions () const
- virtual void setChangeOptionsBit ([DtIffAtcNav aidsChangeOptionsBit](#) bit, [DtBoolean](#) value)
Set/Get the system change/option by bit value
- virtual [DtBoolean](#) changeOptionsBit ([DtIffAtcNav aidsChangeOptionsBit](#) bit) const

- `virtual void print() const`
Print the information contained in the record
 - `virtual void unloadNetRecord(const DtNetIffAtcNav aidsSystemID& record)`
Load/Unload data from a packet record
 - `virtual void loadNetRecord(DtNetIffAtcNav aidsSystemID& record) const`
 - [`DtIffAtcNav aidsSysType`](#) `myType`
Data members:
 - [`DtIffAtcNav aidsSysName`](#) `myName`
 - [`DtIffAtcNav aidsSysMode`](#) `myMode`
 - `DtU8 myChangeOptions`
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtlffAtcNavaidSysType

-
- DtlffAtcNavaidOtherSysType
 - DtlffAtcNavaidTransponder
 - DtlffAtcNavaidInterrogator
 - DtlffAtcNavaidSovietTransponder
 - DtlffAtcNavaidSovietInterrogator
-

Documentation

- DtIffAtcNavaidOtherSysType
- DtIffAtcNavaidTransponder
- DtIffAtcNavaidInterrogator
- DtIffAtcNavaidSovietTransponder
- DtIffAtcNavaidSovietInterrogator

In file disenums.h:

enum DtIffAtcNavaidSysName

IFF/ATC/NAVAIDS PDU enums

-  DtIffAtcNavaidSysOtherSysName
 -  DtIffAtcNavaidSysMarkX
 -  DtIffAtcNavaidSysMarkXII
 -  DtIffAtcNavaidSysATCRBS
 -  DtIffAtcNavaidSysSoviet
 -  DtIffAtcNavaidSysModeS
 -  DtIffAtcNavaidSysMarkX_XII_ATCRBS
 -  DtIffAtcNavaidSysMarkX_XII_ATCRBS_ModeS
-

Documentation

IFF/ATC/NAVAIDS PDU enums

-  DtIffAtcNavaidSysOtherSysName
-  DtIffAtcNavaidSysMarkX
-  DtIffAtcNavaidSysMarkXII
-  DtIffAtcNavaidSysATCRBS
-  DtIffAtcNavaidSysSoviet
-  DtIffAtcNavaidSysModeS
-  DtIffAtcNavaidSysMarkX_XII_ATCRBS
-  DtIffAtcNavaidSysMarkX_XII_ATCRBS_ModeS

In file disenums.h:

enum DtIffAtcNavaidSysMode

 DtIffAtcNavaidSysModeInvalid

Documentation

 DtIffAtcNavaidSysModeInvalid

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtIffAtcNavaidChangeOptionsBit

-
-  DtIffAtcNavaidChangeIndicator
 -  DtIffAtcNavaidAlternateMode4
 -  DtIffAtcNavaidAlternateMode5C
-

Documentation

-  DtIffAtcNavaidChangeIndicator
-  DtIffAtcNavaidAlternateMode4
-  DtIffAtcNavaidAlternateMode5C

In file iffatcnavoper.h:

class DtlffAtcNav aidsOperational

class DtlffAtcNav aidsOperational: DtlffAtcNav aidsOperational is a class which contains the fundamental operational data for the IFF/ATC/NAVAIDS PDU

Public Methods

	DtlffAtcNav aidsOperational () <i>Constructors</i>
	DtlffAtcNav aidsOperational (const DtlffAtcNav aidsOperational & orig)
DtlffAtcNav aidsOperational &	operator= (const DtlffAtcNav aidsOperational & orig) <i>Assignment operator</i>
DtlffAtcNav aidsOperational &	operator= (const DtNetIffAtcNav aidsOperational& orig)
virtual	~DtlffAtcNav aidsOperational () <i>Destructor</i>
virtual void	setSystemStatus (DtU8 status) <i>Set/Get the system status</i>
virtual DtU8	systemStatus () const
virtual void	setSystemStatusBit (DtlffAtcNav aidsStatusBit bit, DtBoolean value) <i>Set/Get the system status by bit</i>
virtual DtBoolean	systemStatusBit (DtlffAtcNav aidsStatusBit bit) const
virtual void	setAltParameter4 (DtlffAtcNav aidsAltParm4 mod) <i>Set/Get alternate parameter 4</i>
virtual DtlffAtcNav aidsAltParm4	altParameter4 () const
virtual void	setInformationLayers (DtU8 layers) <i>Set/Get the information layers field (each layer has a corresponding bit in this field which indicates the presence of that layer)</i>
virtual DtU8	informationLayers () const
virtual void	setLayer (DtU8 layer, DtBoolean value) <i>Set/Query an information layer field for a particular layer</i>
virtual DtBoolean	layerPresent (int layer) const
virtual void	setModifier (DtlffAtcNav aidsModifier mod) <i>Set/Get modifier</i>
virtual DtlffAtcNav aidsModifier	modifier () const
virtual void	setParameter (int index, DtU16 value) <i>Set/Get a parameter (unsigned 16-bit values)</i>
virtual DtU16	parameter (int index) const
virtual void	setParameterCode (int index, int code, int value) <i>Set/Get parameter code elements</i>
virtual int	parameterCode (int index, int code) const

virtual void	setParameterBit (int index, DtIffAtcNav aidsParameterBit bit, DtBoolean value) <i>Set/Get parameter code bit</i>
virtual DtBoolean	parameterBit (int index, DtIffAtcNav aidsParameterBit bit) const
virtual void	print () const <i>Print the information contained in the record</i>
virtual void	unloadNetRecord (const DtNetIffAtcNav aidsOperational& record) <i>Load/Unload data from a packet record</i>
virtual void	loadNetRecord (DtNetIffAtcNav aidsOperational& record) const

Public

The following functions allow you to set/get the fundamental operational parameters

Protected Fields

DtU8	myStatus <i>Data members:</i>
DtU8	myAltParameter4
DtU8	myLayers
DtU8	myModifier
DtU16	myParameter [6]

Documentation

class DtIffAtcNav aidsOperational: DtIffAtcNav aidsOperational is a class which contains the fundamental operational data for the IFF/ATC/NAVAIDS PDU
DtIffAtcNav aidsOperational() Constructors
DtIffAtcNav aidsOperational(const DtIffAtcNav aidsOperational& orig)
DtIffAtcNav aidsOperational& operator=(const DtIffAtcNav aidsOperational& orig) Assignment operator
DtIffAtcNav aidsOperational& operator=(const DtNetIffAtcNav aidsOperational& orig)
virtual ~DtIffAtcNav aidsOperational() Destructor
virtual void setSystemStatus(DtU8 status) Set/Get the system status
virtual DtU8 systemStatus() const
virtual void setSystemStatusBit(DtIffAtcNav aidsStatusBit bit, DtBoolean value)

Set/Get the system status by bit

virtual [DtBoolean](#) systemStatusBit([DtIffAtcNavAidsStatusBit](#) bit) const

virtual void setAltParameter4([DtIffAtcNavAidsAltParm4](#) mod)

Set/Get alternate parameter 4

virtual [DtIffAtcNavAidsAltParm4](#) altParameter4() const

virtual void setInformationLayers(DtU8 layers)

Set/Get the information layers field (each layer has a corresponding bit in this field which indicates the presence of that layer)

virtual DtU8 informationLayers() const

virtual void setLayer(DtU8 layer, [DtBoolean](#) value)

Set/Query an information layer field for a particular layer

virtual [DtBoolean](#) layerPresent(int layer) const

virtual void setModifier([DtIffAtcNavAidsModifier](#) mod)

Set/Get modifier

virtual [DtIffAtcNavAidsModifier](#) modifier() const

The following functions allow you to set/get the fundamental operational parameters. These parameters may be stored in different formats depending on the system type field. These functions do not check to make sure that the appropriate format is being used.

virtual void setParameter(int index, DtU16 value)

Set/Get a parameter (unsigned 16-bit values)

virtual DtU16 parameter(int index) const

virtual void setParameterCode(int index, int code, int value)

Set/Get parameter code elements. These are stored in 3-bit integers which occupy bits 0-14 of the parameter.

virtual int parameterCode(int index, int code) const

virtual void setParameterBit(int index, [DtIffAtcNavAidsParameterBit](#) bit, [DtBoolean](#) value)

Set/Get parameter code bit

virtual [DtBoolean](#) parameterBit(int index, [DtIffAtcNavAidsParameterBit](#) bit) const

virtual void print() const

Print the information contained in the record

virtual void unloadNetRecord(const DtNetIffAtcNavAidsOperational& record)

Load/Unload data from a packet record

virtual void loadNetRecord(DtNetIffAtcNavAidsOperational& record) const

DtU8 myStatus

Data members:

- DtU8 myAltParameter4
 - DtU8 myLayers
 - DtU8 myModifier
 - DtU16 myParameter[6]
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtlffAtcNavaidStatusBit

-
-  DtlffAtcNavaidSystemOnOff
 -  DtlffAtcNavaidParameter1
 -  DtlffAtcNavaidParameter2
 -  DtlffAtcNavaidParameter3
 -  DtlffAtcNavaidParameter4
 -  DtlffAtcNavaidParameter5
 -  DtlffAtcNavaidParameter6
 -  DtlffAtcNavaidOperationalStatus
-

Documentation

-  DtIffAtcNavaidSystemOnOff
-  DtIffAtcNavaidParameter1
-  DtIffAtcNavaidParameter2
-  DtIffAtcNavaidParameter3
-  DtIffAtcNavaidParameter4
-  DtIffAtcNavaidParameter5
-  DtIffAtcNavaidParameter6
-  DtIffAtcNavaidOperationalStatus

In file disenums.h:

enum DtlffAtcNavaidAltParm4

-
-  **DtlffAtcNavaidAltParm4Other**
 -  **DtlffAtcNavaidAltParm4Valid**
 -  **DtlffAtcNavaidAltParm4Invalid**
 -  **DtlffAtcNavaidAltParm4NoResponse**
-

Documentation

-  **DtIffAtcNavaidAltParm4Other**
-  **DtIffAtcNavaidAltParm4Valid**
-  **DtIffAtcNavaidAltParm4Invalid**
-  **DtIffAtcNavaidAltParm4NoResponse**

In file disenums.h:

enum DtIffAtcNavaidModifier

 DtIffAtcNavaidModifierInvalid

Documentation

 DtIffAtcNavaidModifierInvalid

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtIffAtcNavaidParameterBit

-
-  DtIffAtcNavaidParameterTCAS
 -  DtIffAtcNavaidParameterStatus
 -  DtIffAtcNavaidParameterDamage
 -  DtIffAtcNavaidParameterMalfunction
-

Documentation

-  DtIffAtcNavaidParameterTCAS
-  DtIffAtcNavaidParameterStatus
-  DtIffAtcNavaidParameterDamage
-  DtIffAtcNavaidParameterMalfunction

In file iffatcnavl.h:

class DtlffAtcNavaidLayerHeader

class DtlffAtcNavaidLayerHeader: DtlffAtcNavaidLayerHeader is a class which contains the layer header information for a layer of the IFF/ATC/NAVAIDS PDU

Public Methods

	DtlffAtcNavaidLayerHeader () <i>Constructors</i>
	DtlffAtcNavaidLayerHeader (const DtlffAtcNavaidLayerHeader & orig)
 DtlffAtcNavaidLayerHeader &	operator= (const DtlffAtcNavaidLayerHeader & orig) <i>Assignment operator</i>
 DtlffAtcNavaidLayerHeader &	operator= (const DtNetlffAtcNavaidLayerHeader& orig)
 virtual	~DtlffAtcNavaidLayerHeader () <i>Destructor</i>
 virtual void	setLayerNumber (DtU8 num) <i>Set/Get the layer number</i>
 virtual DtU8	layerNumber () const
 virtual void	setLayerInfo (DtlffAtcNavaidLayerSpecificInfo info) <i>Set/Get the layer specific information</i>
 virtual DtlffAtcNavaidLayerSpecificInfo	layerInfo () const
 virtual void	setLayerLength (DtU16 length) <i>Set/Get the layer length</i>
 virtual DtU16	layerLength () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetlffAtcNavaidLayerHeader& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetlffAtcNavaidLayerHeader& record) const

Protected Fields

 DtU8	myLayerNumber <i>Data members:</i>
 DtU8	myLayerInfo
 DtU16	myLayerLength

Documentation

class DtIffAtcNav aidsLayerHeader:

DtIffAtcNav aidsLayerHeader is a class which contains the layer header information for a layer of the IFF/ATC/NAVAIDS PDU. These records are maintained automatically by the PDU class as layers are created and deleted from the PDU.

`DtIffAtcNav aidsLayerHeader()`

Constructors

`DtIffAtcNav aidsLayerHeader(const DtIffAtcNav aidsLayerHeader& orig)`

`DtIffAtcNav aidsLayerHeader& operator=(const DtIffAtcNav aidsLayerHeader& orig)`

Assignment operator

`DtIffAtcNav aidsLayerHeader& operator=(const DtNetIffAtcNav aidsLayerHeader& orig)`

`virtual ~DtIffAtcNav aidsLayerHeader()`

Destructor

`virtual void setLayerNumber(DtU8 num)`

Set/Get the layer number

`virtual DtU8 layerNumber() const`

`virtual void setLayerInfo(DtIffAtcNav aidsLayerSpecificInfo info)`

Set/Get the layer specific information

`virtual DtIffAtcNav aidsLayerSpecificInfo layerInfo() const`

`virtual void setLayerLength(DtU16 length)`

Set/Get the layer length

`virtual DtU16 layerLength() const`

`virtual void print() const`

Print the information contained in the record

`virtual void unloadNetRecord(const DtNetIffAtcNav aidsLayerHeader& record)`

Load/Unload data from a packet record

`virtual void loadNetRecord(DtNetIffAtcNav aidsLayerHeader& record) const`

`DtU8 myLayerNumber`

Data members:

`DtU8 myLayerInfo`

`DtU16 myLayerLength`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtlffAtcNavaidLayerSpecificInfo

 DtlffAtcNavaidLayerInfoInvalid

Documentation

 DtIffAtcNavaidLayerInfoInvalid

[Alphabetic index](#) [Hierarchy of classes](#)

In file iffatcnavbeam.h:

class DtIffAtcNavaisBeamData

class DtIffAtcNavaisBeamData: DtIffAtcNavaisBeamData is a class which contains the antenna data for the IFF/ATC/NAVAIDS PDU

Public Methods

	DtIffAtcNavaisBeamData () <i>Constructors</i>
	DtIffAtcNavaisBeamData (const DtIffAtcNavaisBeamData & orig)
 DtIffAtcNavaisBeamData &	operator= (const DtIffAtcNavaisBeamData & orig) <i>Assignment operator</i>
 DtIffAtcNavaisBeamData &	operator= (const DtNetIffAtcNavaisBeamData& orig)
 virtual	~DtIffAtcNavaisBeamData () <i>Destructor</i>
 virtual void	setAzimuthCenter (DtFloat32 value) <i>Set/Get the Azimuth Center</i>
 virtual DtFloat32	azimuthCenter () const
 virtual void	setAzimuthSweep (DtFloat32 value) <i>Set/Get the Azimuth Sweep</i>
 virtual DtFloat32	azimuthSweep () const
 virtual void	setElevationCenter (DtFloat32 value) <i>Set/Get the Elevation Center</i>
 virtual DtFloat32	elevationCenter () const
 virtual void	setElevationSweep (DtFloat32 value) <i>Set/Get the Elevation Sweep</i>
 virtual DtFloat32	elevationSweep () const
 virtual void	setSweepSync (DtFloat32 value) <i>Set/Get the Sweep Sync</i>
 virtual DtFloat32	sweepSync () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetIffAtcNavaisBeamData& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetIffAtcNavaisBeamData& record) const

Protected Fields

 DtFloat32	myAzimuthCenter <i>Data members:</i>
 DtFloat32	myAzimuthSweep

- DtFloat32 **myElevationCenter**
 - DtFloat32 **myElevationSweep**
 - DtFloat32 **mySweepSync**
-

Documentation

class DtIffAtcNav aidsBeamData: DtIffAtcNav aidsBeamData is a class which contains the antenna data for the IFF/ATC/NAVAIDS PDU

- `DtIffAtcNav aidsBeamData()`

Constructors

- `DtIffAtcNav aidsBeamData(const DtIffAtcNav aidsBeamData& orig)`

- `DtIffAtcNav aidsBeamData& operator=(const DtIffAtcNav aidsBeamData& orig)`

Assignment operator

- `DtIffAtcNav aidsBeamData& operator=(const DtNetIffAtcNav aidsBeamData& orig)`

- `virtual ~DtIffAtcNav aidsBeamData()`

Destructor

- `virtual void setAzimuthCenter(DtFloat32 value)`

Set/Get the Azimuth Center

- `virtual DtFloat32 azimuthCenter() const`

- `virtual void setAzimuthSweep(DtFloat32 value)`

Set/Get the Azimuth Sweep

- `virtual DtFloat32 azimuthSweep() const`

- `virtual void setElevationCenter(DtFloat32 value)`

Set/Get the Elevation Center

- `virtual DtFloat32 elevationCenter() const`

- `virtual void setElevationSweep(DtFloat32 value)`

Set/Get the Elevation Sweep

- `virtual DtFloat32 elevationSweep() const`

- `virtual void setSweepSync(DtFloat32 value)`

Set/Get the Sweep Sync

- `virtual DtFloat32 sweepSync() const`

- `virtual void print() const`

Print the information contained in the record

- `virtual void unloadNetRecord(const DtNetIffAtcNav aidsBeamData& record)`

Load/Unload data from a packet record

virtual void loadNetRecord(DtNetIffAtcNavAidsBeamData& record) const

DtFloat32 myAzimuthCenter

Data members:

DtFloat32 myAzimuthSweep

DtFloat32 myElevationCenter

DtFloat32 myElevationSweep

DtFloat32 mySweepSync

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file iffatcnavopr2.h:

class DtlffAtcNav aidsSecondOperational

class DtlffAtcNav aidsSecondOperational: DtlffAtcNav aidsSecondOperational is a class which contains the secondary operational data for a layer of the IFF/ATC/NAVAIDS PDU

Public Methods

	DtlffAtcNav aidsSecondOperational () <i>Constructors</i>
	DtlffAtcNav aidsSecondOperational (const DtlffAtcNav aidsSecondOperational & orig)
 DtlffAtcNav aidsSecondOperational &	operator= (const DtlffAtcNav aidsSecondOperational & orig) <i>Assignment operators</i>
 DtlffAtcNav aidsSecondOperational &	operator= (const DtNetIffAtcNav aidsSecondOperational& orig)
 virtual	~DtlffAtcNav aidsSecondOperational () <i>Destructor</i>
 virtual void	setParameter1 (DtlffAtcNav aidsParm1 value) <i>Set/Get parameter 1</i>
 virtual DtlffAtcNav aidsParm1	parameter1 () const
 virtual void	setParameter2 (DtlffAtcNav aidsParm2 value) <i>Set/Get parameter 2</i>
 virtual DtlffAtcNav aidsParm2	parameter2 () const
 virtual void	setNumParameterSets (DtU16) <i>Get the number of parameter sets</i>
 virtual DtU16	numParameterSets () const
 virtual void	print () const <i>Print the information contained in the record</i>
 virtual void	unloadNetRecord (const DtNetIffAtcNav aidsSecondOperational& record) <i>Load/Unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetIffAtcNav aidsSecondOperational& record) const

Protected Fields

 DtU8	myParameter1 <i>Data members:</i>
 DtU8	myParameter2
 DtU16	myNumParameterSets

Documentation

class DtIffAtcNav aidsSecondOperational: DtIffAtcNav aidsSecondOperational is a class which contains the secondary operational data for a layer of the IFF/ATC/NAVAIDS PDU

`DtIffAtcNav aidsSecondOperational()`

Constructors

`DtIffAtcNav aidsSecondOperational( const DtIffAtcNav aidsSecondOperational& orig)`

`DtIffAtcNav aidsSecondOperational& operator=( const DtIffAtcNav aidsSecondOperational& orig)`

Assignment operators

`DtIffAtcNav aidsSecondOperational& operator=( const DtNetIffAtcNav aidsSecondOperational& orig)`

`virtual ~DtIffAtcNav aidsSecondOperational()`

Destructor

`virtual void setParameter1(DtIffAtcNav aidsParm1 value)`

Set/Get parameter 1

`virtual DtIffAtcNav aidsParm1 parameter1() const`

`virtual void setParameter2(DtIffAtcNav aidsParm2 value)`

Set/Get parameter 2

`virtual DtIffAtcNav aidsParm2 parameter2() const`

`virtual void setNumParameterSets(DtU16)`

Get the number of parameter sets

`virtual DtU16 numParameterSets() const`

`virtual void print() const`

Print the information contained in the record

`virtual void unloadNetRecord( const DtNetIffAtcNav aidsSecondOperational& record)`

Load/Unload data from a packet record

`virtual void loadNetRecord( DtNetIffAtcNav aidsSecondOperational& record) const`

`DtU8 myParameter1`

Data members:

`DtU8 myParameter2`

`DtU16 myNumParameterSets`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtlffAtcNavaidParm1

 DtlffAtcNavaidParm1Invalid

Documentation

 DtIffAtcNavaidParm1Invalid

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtlffAtcNavaidParm2

 DtlffAtcNavaidParm2Invalid

Documentation

 DtIffAtcNavaidParm2Invalid

[Alphabetic index](#) [Hierarchy of classes](#)

In file iffatcnavaiparm.h:

class DtlffAtcNavaisParameter

class DtlffAtcNavaisParameter: DtlffAtcNavaisParameter is a class which contains the parameter data for the IFF/ATC/NAVAIDS PDU

Public Methods

	DtlffAtcNavaisParameter () <i>Constructors</i>
	DtlffAtcNavaisParameter (const DtlffAtcNavaisParameter & orig)
DtlffAtcNavaisParameter &	operator= (const DtlffAtcNavaisParameter & orig) <i>Assignment operator</i>
DtlffAtcNavaisParameter &	operator= (const DtNetlffAtcNavaisParameter& orig)
virtual	~DtlffAtcNavaisParameter () <i>Destructor</i>
virtual void	setERP (DtFloat32 value) <i>Set/Get the ERP</i>
virtual DtFloat32	ERP () const
virtual void	setFrequency (DtFloat32 value) <i>Set/Get the frequency</i>
virtual DtFloat32	frequency () const
virtual void	setPgRF (DtFloat32 value) <i>Set/Get the PgRF</i>
virtual DtFloat32	PgRF () const
virtual void	setPulseWidth (DtFloat32 value) <i>Set/Get the pulse width</i>
virtual DtFloat32	pulseWidth () const
virtual void	setBurstLength (DtFloat32 value) <i>Set/Get the burst length</i>
virtual DtFloat32	burstLength () const
virtual void	setApplicableModes (DtlffAtcNavaisApplicableModes value) <i>Set/Get the applicable modes</i>
virtual DtlffAtcNavaisApplicableModes	applicableModes () const
virtual void	print () const <i>Print the information contained in the record</i>
virtual void	unloadNetRecord (const DtNetlffAtcNavaisParameter& record) <i>Load/Unload data from a packet record</i>
virtual void	loadNetRecord (DtNetlffAtcNavaisParameter& record) const

Protected Fields

 DtFloat32 **myERP**
Data members:

 DtFloat32 **myFrequency**

 DtFloat32 **myPgRF**

 DtFloat32 **myPulseWidth**

 DtFloat32 **myBurstLength**

 DtU8 **myApplicableModes**

Documentation

class DtIffAtcNav aidsParameter: DtIffAtcNav aidsParameter is a class which contains the parameter data for the IFF/ATC/NAVAIDS PDU

 **DtIffAtcNav aidsParameter()**

Constructors

 **DtIffAtcNav aidsParameter(const [DtIffAtcNav aidsParameter](#)& orig)**

 **[DtIffAtcNav aidsParameter](#)& operator=(const [DtIffAtcNav aidsParameter](#)& orig)**

Assignment operator

 **[DtIffAtcNav aidsParameter](#)& operator=(const DtNetIffAtcNav aidsParameter& orig)**

 **virtual ~DtIffAtcNav aidsParameter()**

Destructor

 **virtual void setERP(DtFloat32 value)**

Set/Get the ERP

 **virtual DtFloat32 ERP() const**

 **virtual void setFrequency(DtFloat32 value)**

Set/Get the frequency

 **virtual DtFloat32 frequency() const**

 **virtual void setPgRF(DtFloat32 value)**

Set/Get the PgRF

 **virtual DtFloat32 PgRF() const**

 **virtual void setPulseWidth(DtFloat32 value)**

Set/Get the pulse width

 **virtual DtFloat32 pulseWidth() const**

 **virtual void setBurstLength(DtFloat32 value)**

Set/Get the burst length

- `virtual DtFloat32 burstLength() const`
- `virtual void setApplicableModes(DtIffAtcNav aidsApplicableModes value)`
Set/Get the applicable modes
- `virtual DtIffAtcNav aidsApplicableModes applicableModes() const`
- `virtual void print() const`
Print the information contained in the record
- `virtual void unloadNetRecord(const DtNetIffAtcNav aidsParameter& record)`
Load/Unload data from a packet record
- `virtual void loadNetRecord(DtNetIffAtcNav aidsParameter& record) const`
- `DtFloat32 myERP`
Data members:
- `DtFloat32 myFrequency`
- `DtFloat32 myPgRF`
- `DtFloat32 myPulseWidth`
- `DtFloat32 myBurstLength`
- `DtU8 myApplicableModes`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtIffAtcNavaidApplicableModes

 DtIffAtcNavaidAppModelInvalid

Documentation

 DtIffAtcNavaidAppModeInvalid

[Alphabetic index](#) [Hierarchy of classes](#)

In file griddatapdu.h:

class DtGriddedDataPdu: public DtPdu

Inheritance:

DtGriddedDataPdu - DtPdu - DtBaseMsg

Public Methods

	DtGriddedDataPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtGriddedDataPdu (const DtNetGriddedDataPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtGriddedDataPdu (const <u>DtGriddedDataPdu</u> & orig)
 virtual	~DtGriddedDataPdu () <i>destructor</i>
 static <u>DtPdu</u> *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtGriddedDataPdu</i>
 static void	addCallback (<u>DtExerciseConn</u> * conn, DtGriddedDataPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
 static void	removeCallback (<u>DtExerciseConn</u> * conn, DtGriddedDataPduCb cb, void* usr)
 <u>DtGriddedDataPdu</u> &	operator= (const <u>DtGriddedDataPdu</u> & orig) <i>Assignment operator</i>
 virtual void	setEnvSimAppID (const <u>DtEntityIdentifier</u> & app) <i>Set/Get environmental simulation application identifier</i>
 virtual const <u>DtEntityIdentifier</u> &	envSimAppID () const
 virtual void	setFieldNumber (DtU16 field) <i>Set/Get field number</i>
 virtual DtU16	fieldNumber () const
 virtual void	setPduNumber (DtU16 num) <i>Set/Get the PDU index number</i>
 virtual DtU16	pduNumber () const
 virtual void	setPduTotal (DtU16 total) <i>Set/Get total number of PDUs used to transmit the environmental data</i>
 virtual DtU16	pduTotal () const
 virtual void	setCoordSystem (DtU16 sys) <i>Set/Get coordinate system</i>

virtual DtU16	coordSystem () const
virtual DtBoolean	setNumGridAxes (DtU8 axes) <i>Set/Get number of grid axes</i>
virtual DtU8	numGridAxes () const
virtual void	setConstantGrid (DtU8 constant) <i>Set/Get constant grid indicator</i>
virtual DtU8	constantGrid () const
virtual void	setEnvironmentType (const DtEnvironmentTypeRecord & type) <i>Set/Get environment type</i>
virtual const DtEnvironmentTypeRecord &	environmentType () const
virtual void	setOrientation (const DtTaitBryan & orientation) <i>Set/Get orientation</i>
virtual const DtTaitBryan &	orientation () const
virtual void	setSampleTime (int index, DtU32 time) <i>Set/Get sample time</i>
virtual DtU32	sampleTime (int index) const
virtual void	setTotalValues (DtU32 total) <i>Set/Get total values</i>
virtual DtU32	totalValues () const
virtual DtBoolean	setVectorDim (DtU8 dim) <i>Set/Get vector dimension</i>
virtual DtU8	vectorDim () const
virtual DtGridAxisType	gridAxisRecordType (int index) const <i>Get the grid axes record type for a particular record</i>
virtual DtBoolean	setGridAxisRecord (int index, const DtGridAxisRecordFixed & init) <i>Set/Get the grid axis record type and values of record index to values specified by the initializer in the second argument</i>
virtual DtBoolean	getGridAxisRecord (int index, DtGridAxisRecordFixed & record) const
virtual DtBoolean	setGridAxisRecord (int index, const DtGridAxisRecordVariable & init)
virtual DtBoolean	getGridAxisRecord (int index, DtGridAxisRecordVariable & record) const
virtual DtGridDataRecType	gridDataRecordType (int index) const <i>Get the grid data record type for a particular record</i>
virtual DtBoolean	setGridDataRecord (int index, const DtGridDataRecordType0 & init) <i>Set/Get the grid data record type and values of record index to values specified by the initializer in the second argument</i>
virtual DtBoolean	getGridDataRecord (int index, DtGridDataRecordType0 & record) const
virtual DtBoolean	setGridDataRecord (int index, const DtGridDataRecordType1 & init)
virtual DtBoolean	getGridDataRecord (int index, DtGridDataRecordType1 & record) const
virtual DtBoolean	setGridDataRecord (int index, const DtGridDataRecordType2 & init)
virtual DtBoolean	getGridDataRecord (int index, DtGridDataRecordType2 & record) const
virtual void	printData () const <i>Print the data contained by this PDU</i>

Protected Fields

 static int	theStaticSize <i>Size of PDU with zero parameter records</i>
 DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>
 DtEnvironmentTypeRecord	myEnvironmentType
 DtTaitBryan	myEulerAngles

Protected Methods

 virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
 virtual DtNetGriddedDataPdu*	netPduStruct () const <i>Casts the netPdu element to our network struct</i>
 virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>
 DtGriddedDataPdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type</i>
 virtual int	gridAxisRecordLength (DtNetGridAxisRecordFixed* rec) const <i>Returns the length of a given grid axis record</i>
 virtual int	gridDataRecordLength (DtNetGridDataRecord* rec) const <i>Returns the length of a given grid data record</i>
 virtual int	gridAxisRecordOffset (int recordIndex) const <i>Returns a buffer offset to the given grid axis record</i>
 virtual int	gridDataRecordOffset (int recordIndex) const <i>Returns a buffer offset to the given grid data record</i>
 virtual DtBoolean	resizeGridAxisRecord (DtNetGridAxisRecordFixed* rec, int newSize) <i>Resizes a grid axis record</i>
 virtual DtBoolean	resizeGridDataRecord (DtNetGridDataRecord* rec, int newSize) <i>Resizes a grid data record</i>

Inherited from [DtPdu](#):

Public Methods

 virtual void* packet () const
 virtual int status () const
 virtual void setVersion (int vers)
 virtual int protocolVersion () const
 virtual void setExerciseld (int id)
 virtual int exerciseld () const
 virtual DtPduKind kind () const

- virtual [DtProtocolFamily](#) **protocolFamily()** const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp()** const
- virtual [DtTimeStampType](#) **timeStampType()** const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print()** const
- virtual void **printHeader()** const
- inline int **exerciseld()** const
- inline [DtPduKind](#) **kind()** const
- inline DtNetPduHeader* **netHeader()** const
- inline void* **packet()** const
- inline [DtProtocolFamily](#) **protocolFamily()** const
- inline int **protocolVersion()** const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status()** const
- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader()** const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)

- virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtGriddedDataPdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors

- DtGriddedDataPdu**(const DtNetGriddedDataPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

- `DtGriddedDataPdu(const DtGriddedDataPdu& orig)`
- `virtual ~DtGriddedDataPdu()`
destructor
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtGriddedDataPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtGriddedDataPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtGriddedDataPduCb cb, void* usr)`
- `DtGriddedDataPdu& operator=(const DtGriddedDataPdu& orig)`
Assignment operator
- `virtual void setEnvSimAppID(const DtEntityIdentifier& app)`
Set/Get environmental simulation application identifier
- `virtual const DtEntityIdentifier& envSimAppID() const`
- `virtual void setFieldNumber(DtU16 field)`
Set/Get field number
- `virtual DtU16 fieldNumber() const`
- `virtual void setPduNumber(DtU16 num)`
Set/Get the PDU index number
- `virtual DtU16 pduNumber() const`
- `virtual void setPduTotal(DtU16 total)`
Set/Get total number of PDUs used to transmit the environmental data
- `virtual DtU16 pduTotal() const`
- `virtual void setCoordSystem(DtU16 sys)`
Set/Get coordinate system
- `virtual DtU16 coordSystem() const`
- `virtual DtBoolean setNumGridAxes(DtU8 axes)`
Set/Get number of grid axes
- `virtual DtU8 numGridAxes() const`
- `virtual void setConstantGrid(DtU8 constant)`
Set/Get constant grid indicator
- `virtual DtU8 constantGrid() const`
- `virtual void setEnvironmentType(const DtEnvironmentTypeRecord& type)`
Set/Get environment type

- virtual const [DtEnvironmentTypeRecord&](#) environmentType() const
- virtual void setOrientation(const [DtTaitBryan&](#) orientation)
Set/Get orientation
- virtual const [DtTaitBryan&](#) orientation() const
- virtual void setSampleTime(int index, DtU32 time)
Set/Get sample time. This is a 64-bit quantity. The argument index selects between the most significant 32-bit word (0) and least significant 32-bit word (1) of the complete 64-bit value.
- virtual DtU32 sampleTime(int index) const
- virtual void setTotalValues(DtU32 total)
Set/Get total values
- virtual DtU32 totalValues() const
- virtual [DtBoolean](#) setVectorDim(DtU8 dim)
Set/Get vector dimension. setVectorDim may return false if the PDU size exceeds the maximum allowable size for a PDU.
- virtual DtU8 vectorDim() const
- virtual [DtGridAxisType](#) gridAxisRecordType(int index) const
Get the grid axes record type for a particular record
- virtual [DtBoolean](#) setGridAxisRecord(int index, const [DtGridAxisRecordFixed&](#) init)
Set/Get the grid axis record type and values of record index to values specified by the initializer in the second argument. If index is out of bounds, or if the record type at the specified index does not match the parameter record type (for a get), then DtFalse is returned. If the space would exceed the maximum space permitted for a PDU (for a set) then DtFalse may be returned.
- virtual [DtBoolean](#) getGridAxisRecord(int index, [DtGridAxisRecordFixed&](#) record) const
- virtual [DtBoolean](#) setGridAxisRecord(int index, const [DtGridAxisRecordVariable&](#) init)
- virtual [DtBoolean](#) getGridAxisRecord(int index, [DtGridAxisRecordVariable&](#) record) const
- virtual [DtGridDataRecType](#) gridDataRecordType(int index) const
Get the grid data record type for a particular record
- virtual [DtBoolean](#) setGridDataRecord(int index, const [DtGridDataRecordType0&](#) init)
Set/Get the grid data record type and values of record index to values specified by the initializer in the second argument. If index is out of bounds, or if the record type at the specified index does not match the parameter record type (for a get), then DtFalse is returned. If the space would exceed the maximum space permitted for a PDU (for a set) then DtFalse may be returned.
- virtual [DtBoolean](#) getGridDataRecord(int index, [DtGridDataRecordType0&](#) record) const
- virtual [DtBoolean](#) setGridDataRecord(int index, const [DtGridDataRecordType1&](#) init)
- virtual [DtBoolean](#) getGridDataRecord(int index, [DtGridDataRecordType1&](#) record) const
- virtual [DtBoolean](#) setGridDataRecord(int index, const [DtGridDataRecordType2&](#) init)

- `virtual DtBoolean getGridDataRecord(int index, DtGridDataRecordType2& record) const`
- `virtual void printData() const`
Print the data contained by this PDU
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetGriddedDataPdu* netPduStruct() const`
Casts the netPdu element to our network struct
- `virtual char* var() const`
Returns a pointer to the first byte of variable length data
- `DtGriddedDataPdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
- `virtual int gridAxisRecordLength(DtNetGridAxisRecordFixed* rec) const`
Returns the length of a given grid axis record
- `virtual int gridDataRecordLength(DtNetGridDataRecord* rec) const`
Returns the length of a given grid data record
- `virtual int gridAxisRecordOffset(int recordIndex) const`
Returns a buffer offset to the given grid axis record
- `virtual int gridDataRecordOffset(int recordIndex) const`
Returns a buffer offset to the given grid data record
- `virtual DtBoolean resizeGridAxisRecord(DtNetGridAxisRecordFixed* rec, int newSize)`
Resizes a grid axis record
- `virtual DtBoolean resizeGridDataRecord(DtNetGridDataRecord* rec, int newSize)`
Resizes a grid data record
- `static int theStaticSize`
Size of PDU with zero parameter records
- `DtEntityIdentifier myEntityID`
Temporary objects for reference returns
- `DtEnvironmentTypeRecord myEnvironmentType`
- `DtTaitBryan myEulerAngles`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtGridAxisType

-
-  DtGridAxisFixed
 -  DtGridAxisVariable
-

Documentation

-  DtGridAxisFixed
-  DtGridAxisVariable

In file gridaxirecfix.h:

class DtGridAxisRecordFixed

class DtGridAxisRecordFixed: DtGridAxisRecordFixed is the fixed grid axis record used by the gridded data PDU class (DtGridDataPdu)

Inheritance:

DtGridAxisRecordFixed

Public Methods

	DtGridAxisRecordFixed () <i>Constructors</i>
	DtGridAxisRecordFixed (const DtGridAxisRecordFixed & orig)
virtual	~DtGridAxisRecordFixed () <i>Destructor</i>
DtGridAxisRecordFixed &	operator= (const DtGridAxisRecordFixed & orig) <i>Assignment operator</i>
DtGridAxisRecordFixed &	operator= (const DtNetGridAxisRecordFixed& orig)
virtual void	setDomainInitialXi (const DtFloat64& initial) <i>Set/Get the domain initial Xi</i>
virtual const DtFloat64&	domainInitialXi () const
virtual void	setDomainFinalXi (const DtFloat64& final) <i>Set/Get the domain final Xi</i>
virtual const DtFloat64&	domainFinalXi () const
virtual void	setDomainNumPointsXi (DtU16 num) <i>Set/Get the domain number of points</i>
virtual DtU16	domainNumPointsXi () const
virtual void	setInterleafFactor (DtU8 factor) <i>Set/Get the interleaf factor</i>
virtual DtU8	interleafFactor () const
virtual DtGridAxisType	axisType () const <i>Get the axis type</i>
virtual DtBoolean	setNumPointsXi (DtU16 num) <i>Set/Get the number of points</i>
virtual DtU16	numPointsXi () const
virtual void	setInitialIndexXi (DtU16 index) <i>Set/Get the initial index</i>

virtual DtU16	initialIndexXi () const
virtual int	recordLength () const <i>Set the total number of bytes required for the network structure</i>
virtual void	print () const <i>Print the sample type and data representation</i>
virtual void	baseUnloadNetRecord (const DtNetGridAxisRecordFixed& record) <i>Load/unload data from a packet record</i>
virtual void	baseLoadNetRecord (DtNetGridAxisRecordFixed& record) const
static DtGridAxisType	axisType (const DtNetGridAxisRecordFixed* rec) <i>Get the grid axis record type from a network structure</i>
static int	recordLength (const DtNetGridAxisRecordFixed* rec) <i>Get the grid axis record length from a network structure</i>

Protected Fields

DtFloat64	myDomainInitialXi <i>Data members:</i>
DtFloat64	myDomainFinalXi
DtU16	myDomainNumPointsXi
DtU8	myInterleafFactor
DtGridAxisType	myAxisType
DtU16	myNumPointsXi
DtU16	myInitialIndexXi

Protected Methods

- DtGridAxisRecordFixed** ([DtGridAxisType](#) axType)
Constructor for derived classes

Documentation

class DtGridAxisRecordFixed:

DtGridAxisRecordFixed is the fixed grid axis record used by the gridded data PDU class (DtGridDataPdu). It is also the base class for the variable grid axis record.

- DtGridAxisRecordFixed()**
Constructors
- DtGridAxisRecordFixed**(const [DtGridAxisRecordFixed](#)& orig)
- virtual ~DtGridAxisRecordFixed()**
Destructor
- [DtGridAxisRecordFixed](#)**& operator=(const [DtGridAxisRecordFixed](#)& orig)
Assignment operator

- [DtGridAxisRecordFixed](#)& operator=(const DtNetGridAxisRecordFixed& orig)
- virtual void setDomainInitialXi(const DtFloat64& initial)
Set/Get the domain initial Xi
- virtual const DtFloat64& domainInitialXi() const
- virtual void setDomainFinalXi(const DtFloat64& final)
Set/Get the domain final Xi
- virtual const DtFloat64& domainFinalXi() const
- virtual void setDomainNumPointsXi(DtU16 num)
Set/Get the domain number of points
- virtual DtU16 domainNumPointsXi() const
- virtual void setInterleafFactor(DtU8 factor)
Set/Get the interleaf factor
- virtual DtU8 interleafFactor() const
- virtual [DtGridAxisType](#) axisType() const
Get the axis type
- virtual [DtBoolean](#) setNumPointsXi(DtU16 num)
Set/Get the number of points. May return false if PDU size exceeds the maximum size for a PDU.
- virtual DtU16 numPointsXi() const
- virtual void setInitialIndexXi(DtU16 index)
Set/Get the initial index
- virtual DtU16 initialIndexXi() const
- virtual int recordLength() const
Set the total number of bytes required for the network structure
- virtual void print() const
Print the sample type and data representation
- virtual void baseUnloadNetRecord(const DtNetGridAxisRecordFixed& record)
Load/unload data from a packet record
- virtual void baseLoadNetRecord(DtNetGridAxisRecordFixed& record) const
- static [DtGridAxisType](#) axisType(const DtNetGridAxisRecordFixed* rec)
Get the grid axis record type from a network structure
- static int recordLength(const DtNetGridAxisRecordFixed* rec)
Get the grid axis record length from a network structure
- DtGridAxisRecordFixed([DtGridAxisType](#) axType)
Constructor for derived classes
- DtFloat64 myDomainInitialXi

Data members:

- `DtFloat64 myDomainFinalXi`
 - `DtU16 myDomainNumPointsXi`
 - `DtU8 myInterleafFactor`
 - [DtGridAxisType](#) `myAxisType`
 - `DtU16 myNumPointsXi`
 - `DtU16 myInitialIndexXi`
-

Direct child classes:

[DtGridAxisRecordVariable](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file gridaxirecvar.h:

class DtGridAxisRecordVariable: public DtGridAxisRecordFixed

class DtGridAxisRecordVariable: Instances of DtGridAxisRecordVariable are used to contain the information in the grid axis variable spacing records used by the gridded data PDU

Inheritance:

DtGridAxisRecordVariable - DtGridAxisRecordFixed

Public Methods

	DtGridAxisRecordVariable () <i>Constructors</i>
	DtGridAxisRecordVariable (const DtGridAxisRecordVariable & orig)
virtual	~DtGridAxisRecordVariable () <i>Destructor</i>
DtGridAxisRecordVariable &	operator= (const DtGridAxisRecordVariable & orig) <i>Assignment operators</i>
DtGridAxisRecordVariable &	operator= (const DtNetGridAxisRecordVariable& orig)
virtual int	recordLength () const <i>Get the total number of bytes required for the network structure</i>
virtual void	setCoordinateScaleXi (const DtFloat64& num) <i>Set/Get the coordinate scale</i>
virtual const DtFloat64&	coordinateScaleXi () const
virtual void	setCoordinateOffsetXi (const DtFloat64& num) <i>Set/Get the coordinate offset</i>
virtual const DtFloat64&	coordinateOffsetXi () const
virtual DtBoolean	setNumPointsXi (DtU16 num) <i>Set/Get the number of points</i>
virtual DtU16	numPointsXi () const
virtual void	setXi (int index, DtU16 x) <i>Set/Get an Xi value</i>
virtual DtU16	xi (int index) const
virtual void	print () const <i>Print this record</i>
virtual void	unloadNetRecord (const DtNetGridAxisRecordVariable& record) <i>Load/unload data from a packet record</i>

virtual void

static int

loadNetRecord (DtNetGridAxisRecordVariable& record) const

recordLength (DtNetGridAxisRecordVariable* rec)

Get the grid axis record length from a network structure

Protected Fields

DtFloat64 **myCoordinateScaleXi**

DtFloat64 **myCoordinateOffsetXi**

DtU16* **myXi**

Inherited from [DtGridAxisRecordFixed](#):

Public Methods

virtual void **setDomainInitialXi**(const DtFloat64& initial)

virtual const DtFloat64& **domainInitialXi**() const

virtual void **setDomainFinalXi**(const DtFloat64& final)

virtual const DtFloat64& **domainFinalXi**() const

virtual void **setDomainNumPointsXi**(DtU16 num)

virtual DtU16 **domainNumPointsXi**() const

virtual void **setInterleafFactor**(DtU8 factor)

virtual DtU8 **interleafFactor**() const

virtual [DtGridAxisType](#) **axisType**() const

virtual void **setInitialIndexXi**(DtU16 index)

virtual DtU16 **initialIndexXi**() const

virtual void **baseUnloadNetRecord**(const DtNetGridAxisRecordFixed& record)

virtual void **baseLoadNetRecord**(DtNetGridAxisRecordFixed& record) const

static [DtGridAxisType](#) **axisType**(const DtNetGridAxisRecordFixed* rec)

Protected Fields

DtFloat64 **myDomainInitialXi**

DtFloat64 **myDomainFinalXi**

DtU16 **myDomainNumPointsXi**

DtU8 **myInterleafFactor**

• [DtGridAxisType](#) myAxisType

• DtU16 myNumPointsXi

• DtU16 myInitialIndexXi

Documentation

class DtGridAxisRecordVariable: Instances of DtGridAxisRecordVariable are used to contain the information in the grid axis variable spacing records used by the gridded data PDU

• `DtGridAxisRecordVariable()`

Constructors

• `DtGridAxisRecordVariable(const DtGridAxisRecordVariable& orig)`

• `virtual ~DtGridAxisRecordVariable()`

Destructor

• `DtGridAxisRecordVariable& operator=( const DtGridAxisRecordVariable& orig)`

Assignment operators

• `DtGridAxisRecordVariable& operator=( const DtNetGridAxisRecordVariable& orig)`

• `virtual int recordLength() const`

Get the total number of bytes required for the network structure

• `virtual void setCoordinateScaleXi(const DtFloat64& num)`

Set/Get the coordinate scale

• `virtual const DtFloat64& coordinateScaleXi() const`

• `virtual void setCoordinateOffsetXi(const DtFloat64& num)`

Set/Get the coordinate offset

• `virtual const DtFloat64& coordinateOffsetXi() const`

• `virtual DtBoolean setNumPointsXi(DtU16 num)`

Set/Get the number of points. If the number of points is changed, previous points are not preserved.

• `virtual DtU16 numPointsXi() const`

• `virtual void setXi(int index, DtU16 x)`

Set/Get an Xi value

• `virtual DtU16 xi(int index) const`

• `virtual void print() const`

Print this record

• `virtual void unloadNetRecord(const DtNetGridAxisRecordVariable& record)`

Load/unload data from a packet record

• `virtual void loadNetRecord(DtNetGridAxisRecordVariable& record) const`

static int recordLength(DtNetGridAxisRecordVariable* rec)

Get the grid axis record length from a network structure

DtFloat64 myCoordinateScaleXi

DtFloat64 myCoordinateOffsetXi

DtU16* myXi

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtGridDataRecType

Gridded data PDU enums

-  DtGridDataRecType0
 -  DtGridDataRecType1
 -  DtGridDataRecType2
-

Documentation

Gridded data PDU enums

-  DtGridDataRecType0
-  DtGridDataRecType1
-  DtGridDataRecType2

In file griddatarect0.h:

class DtGridDataRecordType0: public DtGridDataRecord

class DtGridDataRecordType0: Instances of DtGridDataRecordType0 are used to contain the information in the grid data records (of representation type 0) used by the gridded data PDU class

Inheritance:

DtGridDataRecordType0 - [DtGridDataRecord](#)

Public Methods

	DtGridDataRecordType0 () <i>Constructors</i>
	DtGridDataRecordType0 (const DtGridDataRecordType0 & orig)
virtual	~DtGridDataRecordType0 () <i>Destructor</i>
DtGridDataRecordType0 &	operator= (const DtGridDataRecordType0 & orig) <i>Assignment operators</i>
DtGridDataRecordType0 &	operator= (const DtNetGridDataRecordType0& orig)
virtual int	recordLength () const <i>Get the total number of bytes required for the network structure</i>
virtual DtBoolean	setNumDataBytes (DtU16 bytes) <i>Set/Get the number of data bytes When number of data bytes is changed, old values are not preserved</i>
virtual DtU16	numDataBytes () const
virtual void	setDataByte (int byteNum, DtU8 data) <i>Set/Get a data byte</i>
virtual DtU8	dataByte (int byteNum) const
virtual DtBoolean	setData (const DtU8* data, int numBytes) <i>Set/Get multiple data bytes</i>
virtual void	getData (DtU8* buff) const
virtual void	print () const <i>Print this record</i>
virtual void	unloadNetRecord (const DtNetGridDataRecordType0& record) <i>Load/unload data from a packet record</i>
virtual void	loadNetRecord (DtNetGridDataRecordType0& record) const

static int

recordLength (DtNetGridDataRecordType0* rec)
Get the grid data record representation type from a network structure

Protected Fields

DtU16 **myNumDataBytes**

DtU8* **myData**

Inherited from [DtGridDataRecord](#):

Public Methods

virtual void **setSampleType**(DtU16 type)

virtual DtU16 **sampleType**() const

virtual [DtGridDataRecType](#) **recordType**() const

virtual void **baseUnloadNetRecord**(const DtNetGridDataRecord& record)

virtual void **baseLoadNetRecord**(DtNetGridDataRecord& record) const

static [DtGridDataRecType](#) **recordType**(DtNetGridDataRecord* rec)

Protected Fields

DtU16 **mySampleType**

[DtGridDataRecType](#) **myRecordType**

Documentation

class DtGridDataRecordType0: Instances of DtGridDataRecordType0 are used to contain the information in the grid data records (of representation type 0) used by the gridded data PDU class

DtGridDataRecordType0 ()

Constructors

DtGridDataRecordType0 (const [DtGridDataRecordType0](#)& orig)

virtual **~DtGridDataRecordType0** ()

Destructor

[DtGridDataRecordType0](#)& **operator=**(const [DtGridDataRecordType0](#)& orig)

Assignment operators

[DtGridDataRecordType0](#)& **operator=**(const DtNetGridDataRecordType0& orig)

● `virtual int recordLength() const`

Get the total number of bytes required for the network structure

● `virtual DtBoolean setNumDataBytes(DtU16 bytes)`

Set/Get the number of data bytes When number of data bytes is changed, old values are not preserved

● `virtual DtU16 numDataBytes() const`

● `virtual void setDataByte(int byteNum, DtU8 data)`

Set/Get a data byte

● `virtual DtU8 dataByte(int byteNum) const`

● `virtual DtBoolean setData(const DtU8* data, int numBytes)`

Set/Get multiple data bytes. Provide a pointer and data is copied to and from that memory. Will copy numDataBytes() on get.

● `virtual void getData(DtU8* buff) const`

● `virtual void print() const`

Print this record

● `virtual void unloadNetRecord(const DtNetGridDataRecordType0& record)`

Load/unload data from a packet record

● `virtual void loadNetRecord(DtNetGridDataRecordType0& record) const`

● `static int recordLength(DtNetGridDataRecordType0* rec)`

Get the grid data record representation type from a network structure

● `DtU16 myNumDataBytes`

● `DtU8* myData`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file griddatarec.h:

class DtGridDataRecord

class DtGridDataRecord: DtGridDataRecord is an abstract base class for the grid data record classes which are used by the gridded data PDU class (DtGridDataPdu)

Inheritance:

DtGridDataRecord

Public Methods

virtual	~DtGridDataRecord () <i>Destructor</i>
virtual void	setSampleType (DtU16 type) <i>Set/Get the grid data record sample type</i>
virtual DtU16	sampleType () const
virtual DtGridDataRecType	recordType () const <i>Get the grid data record representation type</i>
virtual int	recordLength () const <i>Get the total number of bytes required for the network structure</i>
virtual void	print () const <i>Print the sample type and data representation</i>
virtual void	baseUnloadNetRecord (const DtNetGridDataRecord& record) <i>Load/unload data from a packet record</i>
virtual void	baseLoadNetRecord (DtNetGridDataRecord& record) const
static DtGridDataRecType	recordType (DtNetGridDataRecord* rec) <i>Get the grid data record representation type from a network structure</i>

Protected Fields

DtU16	mySampleType <i>Data members:</i>
DtGridDataRecType	myRecordType

Protected Methods

	DtGridDataRecord () <i>Constructors (protected because this is an abstract base class)</i>
	DtGridDataRecord (const DtGridDataRecord & orig)

- **DtGridDataRecord** ([DtGridDataRecType](#) recType)
 - [DtGridDataRecord](#)& **operator=** (const [DtGridDataRecord](#)& orig)
Assignment operator
-

Documentation

class DtGridDataRecord: DtGridDataRecord is an abstract base class for the grid data record classes which are used by the gridded data PDU class (DtGridDataPdu)

- **virtual ~DtGridDataRecord()**
Destructor
 - **virtual void setSampleType(DtU16 type)**
Set/Get the grid data record sample type
 - **virtual DtU16 sampleType() const**
 - **virtual [DtGridDataRecType](#) recordType() const**
Get the grid data record representation type
 - **virtual int recordLength() const**
Get the total number of bytes required for the network structure
 - **virtual void print() const**
Print the sample type and data representation
 - **virtual void baseUnloadNetRecord(const DtNetGridDataRecord& record)**
Load/unload data from a packet record
 - **virtual void baseLoadNetRecord(DtNetGridDataRecord& record) const**
 - **static [DtGridDataRecType](#) recordType(DtNetGridDataRecord* rec)**
Get the grid data record representation type from a network structure
 - **DtGridDataRecord()**
Constructors (protected because this is an abstract base class)
 - **DtGridDataRecord(const [DtGridDataRecord](#)& orig)**
 - **DtGridDataRecord([DtGridDataRecType](#) recType)**
 - **[DtGridDataRecord](#)& operator= (const [DtGridDataRecord](#)& orig)**
Assignment operator
 - **DtU16 mySampleType**
Data members:
 - **[DtGridDataRecType](#) myRecordType**
-

Direct child classes:

[DtGridDataRecordType2](#)
[DtGridDataRecordType1](#)

[DtGridDataRecordType0](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file griddatarect2.h:

class DtGridDataRecordType2: public DtGridDataRecord

class DtGridDataRecordType2: Instances of DtGridDataRecordType2 are used to contain the information in the grid data records (of representation type 2) used by the gridded data PDU class

Inheritance:

DtGridDataRecordType2 - [DtGridDataRecord](#)

Public Methods

	DtGridDataRecordType2 () <i>Constructors</i>
	DtGridDataRecordType2 (const DtGridDataRecordType2 & orig)
virtual	~DtGridDataRecordType2 () <i>Destructor</i>
DtGridDataRecordType2 &	operator= (const DtGridDataRecordType2 & orig) <i>Assignment operators</i>
DtGridDataRecordType2 &	operator= (const DtNetGridDataRecordType2& orig)
virtual int	recordLength () const <i>Get the total number of bytes required for the network structure</i>
virtual DtBoolean	setNumDataValues (DtU16 values) <i>Set/Get the number of data values Old values are not preserved when the number of values is changed</i>
virtual DtU16	numDataValues () const
virtual void	setDataValue (int index, DtFloat32 data) <i>Set/Get a data value</i>
virtual DtFloat32	dataValue (int index) const
virtual void	print () const <i>Print this record</i>
virtual void	unloadNetRecord (const DtNetGridDataRecordType2& record) <i>Load/unload data from a packet record</i>
virtual void	loadNetRecord (DtNetGridDataRecordType2& record) const
static int	recordLength (DtNetGridDataRecordType2* rec) <i>Get the grid data record representation type from a network structure</i>

Protected Fields

- `DtU16 myNumDataValues`
 - `DtFloat32* myData`
-

Inherited from [DtGridDataRecord](#):

Public Methods

- virtual void **setSampleType**(DtU16 type)
- virtual DtU16 **sampleType**() const
- virtual [DtGridDataRecType](#) **recordType**() const
- virtual void **baseUnloadNetRecord**(const DtNetGridDataRecord& record)
- virtual void **baseLoadNetRecord**(DtNetGridDataRecord& record) const
- static [DtGridDataRecType](#) **recordType**(DtNetGridDataRecord* rec)

Protected Fields

- DtU16 **mySampleType**
 - [DtGridDataRecType](#) **myRecordType**
-

Documentation

class DtGridDataRecordType2: Instances of DtGridDataRecordType2 are used to contain the information in the grid data records (of representation type 2) used by the gridded data PDU class

- `DtGridDataRecordType2 ()`
Constructors
- `DtGridDataRecordType2 (const DtGridDataRecordType2& orig)`
- `virtual ~DtGridDataRecordType2 ()`
Destructor
- `DtGridDataRecordType2& operator=( const DtGridDataRecordType2& orig)`
Assignment operators
- `DtGridDataRecordType2& operator=( const DtNetGridDataRecordType2& orig)`
- `virtual int recordLength() const`
Get the total number of bytes required for the network structure

virtual [DtBoolean](#) setNumDataValues(DtU16 values)

Set/Get the number of data values Old values are not preserved when the number of values is changed

virtual DtU16 numDataValues() const

virtual void setDataValue(int index, DtFloat32 data)

Set/Get a data value

virtual DtFloat32 dataValue(int index) const

virtual void print() const

Print this record

virtual void unloadNetRecord(const DtNetGridDataRecordType2& record)

Load/unload data from a packet record

virtual void loadNetRecord(DtNetGridDataRecordType2& record) const

static int recordLength(DtNetGridDataRecordType2* rec)

Get the grid data record representation type from a network structure

DtU16 myNumDataValues

DtFloat32* myData

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file griddatarect1.h:

class DtGridDataRecordType1: public DtGridDataRecord

class DtGridDataRecordType1: Instances of DtGridDataRecordType1 are used to contain the information in the grid data records (of representation type 1) used by the gridded data PDU class

Inheritance:

DtGridDataRecordType1 - DtGridDataRecord

Public Methods

	DtGridDataRecordType1 () <i>Constructors</i>
	DtGridDataRecordType1 (const DtGridDataRecordType1 & orig)
virtual	~DtGridDataRecordType1 () <i>Destructor</i>
DtGridDataRecordType1 &	operator= (const DtGridDataRecordType1 & orig) <i>Assignment operators</i>
DtGridDataRecordType1 &	operator= (const DtNetGridDataRecordType1& orig)
virtual int	recordLength () const <i>Get the total number of bytes required for the network structure</i>
virtual void	setFieldScale (DtFloat32 scale) <i>Set/Get the field scale</i>
virtual DtFloat32	fieldScale () const
virtual void	setFieldOffset (DtFloat32 offset) <i>Set/Get the field offset</i>
virtual DtFloat32	fieldOffset () const
virtual DtBoolean	setNumDataValues (DtU16 values) <i>Set/Get the number of data values Old values are not preserved when number of values is changed</i>
virtual DtU16	numDataValues () const
virtual void	setDataValue (int index, DtInt16 data) <i>Set/Get a data value</i>
virtual DtInt16	dataValue (int index) const
virtual void	print () const <i>Print this record</i>

- virtual void **unloadNetRecord** (const DtNetGridDataRecordType1& record)
Load/unload data from a packet record
- virtual void **loadNetRecord** (DtNetGridDataRecordType1& record) const
- static int **recordLength** (DtNetGridDataRecordType1* rec)
Get the grid data record representation type from a network structure

Protected Fields

- DtFloat32 **myFieldScale**
- DtFloat32 **myFieldOffset**
- DtU16 **myNumDataValues**
- DtInt16* **myData**

Inherited from [DtGridDataRecord](#):

Public Methods

- virtual void **setSampleType**(DtU16 type)
- virtual DtU16 **sampleType**() const
- virtual [DtGridDataRecType](#) **recordType**() const
- virtual void **baseUnloadNetRecord**(const DtNetGridDataRecord& record)
- virtual void **baseLoadNetRecord**(DtNetGridDataRecord& record) const
- static [DtGridDataRecType](#) **recordType**(DtNetGridDataRecord* rec)

Protected Fields

- DtU16 **mySampleType**
- [DtGridDataRecType](#) **myRecordType**

Documentation

class DtGridDataRecordType1: Instances of DtGridDataRecordType1 are used to contain the information in the grid data records (of representation type 1) used by the gridded data PDU class

- DtGridDataRecordType1** ()
Constructors
- DtGridDataRecordType1** (const [DtGridDataRecordType1](#)& orig)
- virtual ~DtGridDataRecordType1** ()

Destructor

`DtGridDataRecordType1& operator=( const DtGridDataRecordType1& orig)`

Assignment operators

`DtGridDataRecordType1& operator=( const DtNetGridDataRecordType1& orig)`

`virtual int recordLength() const`

Get the total number of bytes required for the network structure

`virtual void setFieldScale(DtFloat32 scale)`

Set/Get the field scale

`virtual DtFloat32 fieldScale() const`

`virtual void setFieldOffset(DtFloat32 offset)`

Set/Get the field offset

`virtual DtFloat32 fieldOffset() const`

`virtual DtBoolean setNumDataValues(DtU16 values)`

Set/Get the number of data values Old values are not preserved when number of values is changed

`virtual DtU16 numDataValues() const`

`virtual void setDataValue(int index, DtInt16 data)`

Set/Get a data value

`virtual DtInt16 dataValue(int index) const`

`virtual void print() const`

Print this record

`virtual void unloadNetRecord(const DtNetGridDataRecordType1& record)`

Load/unload data from a packet record

`virtual void loadNetRecord(DtNetGridDataRecordType1& record) const`

`static int recordLength(DtNetGridDataRecordType1* rec)`

Get the grid data record representation type from a network structure

`DtFloat32 myFieldScale`

`DtFloat32 myFieldOffset`

`DtU16 myNumDataValues`

`DtInt16* myData`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file envprocpdu.h:

class DtEnvironmentalProcessPdu: public DtPdu

Inheritance:

DtEnvironmentalProcessPdu - DtPdu - DtBaseMsg

Public Methods

	DtEnvironmentalProcessPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtEnvironmentalProcessPdu (const DtNetEnvironmentalProcessPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtEnvironmentalProcessPdu (const DtEnvironmentalProcessPdu & orig)
 virtual	~DtEnvironmentalProcessPdu () <i>destructor</i>
 DtEnvironmentalProcessPdu &	operator= (const DtEnvironmentalProcessPdu & orig) <i>Assignment operator</i>
 virtual void	setEnvironmentalProcessID (const DtEntityIdentifier & entity) <i>Set/Get originating environmental process ID</i>
 virtual const DtEntityIdentifier &	environmentalProcessID () const
 virtual void	setEnvironmentType (const DtEntityType & type) <i>Set/Get environment type</i>
 virtual const DtEntityType &	environmentType () const
 virtual void	setModelType (DtEnvironmentalModelType value) <i>Set/Get the model type</i>
 virtual DtEnvironmentalModelType	modelType () const
 virtual void	setEnvironmentalStatus (DtU8 value) <i>Set/Get the environmental status (as an 8-bit value)</i>
 virtual DtU8	environmentalStatus () const
 virtual void	setEnvironmentalStatusBit (DtEnvironmentalStatusBit bit, DtBoolean value) <i>Set/Get an environmental status bit</i>
 virtual DtBoolean	environmentalStatusBit (DtEnvironmentalStatusBit bit) const
 virtual DtBoolean	setNumberOfEnvironmentalRecords (DtU16 newNumberOfRecords) <i>Set/Get number of environment records</i>
 virtual DtU16	numberOfEnvironmentalRecords () const

virtual void	setSequenceNumber (DtU16 value) <i>Set/Get sequence number</i>
virtual DtU16	sequenceNumber () const
virtual DtBoolean	setRecordBitSize (int recordIndex, int newRecordBitSize) <i>Set the size of an environmental record (in bits)</i>
virtual DtNetEnvironmentalRecord*	recordPointer (int recordIndex) <i>Get a pointer to an environmental record</i>
virtual void	printData () const <i>Print the data contained by this PDU</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtEnvironmentalProcessPdu</i>
static void	addCallback (DtExerciseConn * conn, DtEnvironmentalProcessPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtEnvironmentalProcessPduCb cb, void* usr)

Protected Fields

static int	theStaticSize <i>Size of PDU with zero data bytes</i>
DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>
DtEntityType	myEntityType

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
DtNetEnvironmentalProcessPdu*	netPduStruct () const <i>Casts the netPdu element to our network struct Non-virtual since you can't override based on return type</i>
DtEnvironmentalProcessPdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type</i>
virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>
virtual int	recordOffset (int recordIndex) const <i>Returns an offset in bytes from the beginning of the variable length data region to the beginning of a particular record</i>

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtEnvironmentalProcessPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Constructors
- `DtEnvironmentalProcessPdu(const DtNetEnvironmentalProcessPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtEnvironmentalProcessPdu(const DtEnvironmentalProcessPdu& orig)`
- `virtual ~DtEnvironmentalProcessPdu()`
destructor
- `DtEnvironmentalProcessPdu& operator=(const DtEnvironmentalProcessPdu& orig)`
Assignment operator
- `virtual void setEnvironmentalProcessID(const DtEntityIdentifier& entity)`
Set/Get originating environmental process ID
- `virtual const DtEntityIdentifier& environmentalProcessID() const`
- `virtual void setEnvironmentType(const DtEntityType& type)`
Set/Get environment type
- `virtual const DtEntityType& environmentType() const`
- `virtual void setModelType(DtEnvironmentalModelType value)`
Set/Get the model type
- `virtual DtEnvironmentalModelType modelType() const`
- `virtual void setEnvironmentalStatus(DtU8 value)`
Set/Get the environmental status (as an 8-bit value)
- `virtual DtU8 environmentalStatus() const`
- `virtual void setEnvironmentalStatusBit( DtEnvironmentalStatusBit bit, DtBoolean value)`
Set/Get an environmental status bit
- `virtual DtBoolean environmentalStatusBit( DtEnvironmentalStatusBit bit) const`
- `virtual DtBoolean setNumberOfEnvironmentalRecords(DtU16 newNumberOfRecords)`
Set/Get number of environment records

- `virtual DtU16 numberOfEnvironmentalRecords() const`
- `virtual void setSequenceNumber(DtU16 value)`
Set/Get sequence number
- `virtual DtU16 sequenceNumber() const`
- `virtual DtBoolean setRecordBitSize(int recordIndex, int newRecordBitSize)`
Set the size of an environmental record (in bits). Returns DtFalse if the size is invalid.
- `virtual DtNetEnvironmentalRecord* recordPointer(int recordIndex)`
Get a pointer to an environmental record. Returns DtFalse if the record index is out of range.
- `virtual void printData() const`
Print the data contained by this PDU
- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtEnvironmentalProcessPdu. Caller is responsible for deletion.
- `static void addCallback(DtExerciseConn* conn, DtEnvironmentalProcessPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received
- `static void removeCallback(DtExerciseConn* conn, DtEnvironmentalProcessPduCb cb, void* usr)`
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `DtNetEnvironmentalProcessPdu* netPduStruct() const`
Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- `DtEnvironmentalProcessPdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- `virtual char* var() const`
Returns a pointer to the first byte of variable length data
- `virtual int recordOffset(int recordIndex) const`
Returns an offset in bytes from the beginning of the variable length data region to the beginning of a particular record
- `static int theStaticSize`
Size of PDU with zero data bytes
- `DtEntityIdentifier myEntityID`
Temporary objects for reference returns
- `DtEntityType myEntityType`

This class has no child classes.

In file disenums.h:

enum DtEnvironmentalModelType

Environmental Process PDU enums

 DtEnvModelInvalid

Documentation

Environmental Process PDU enums

 DtEnvModelInvalid

In file disenums.h:

enum DtEnvironmentalStatusBit

 DtEnvironmentalStatusLast

 DtEnvironmentalStatusOn

Documentation

 DtEnvironmentalStatusLast

 DtEnvironmentalStatusOn

[Alphabetic index](#) [Hierarchy of classes](#)

In file esupdatepdu.h:

class DtEntityStateUpdatePdu: public DtPdu

Inheritance:

DtEntityStateUpdatePdu - DtPdu - DtBaseMsg

Public Methods

	DtEntityStateUpdatePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtEntityStateUpdatePdu (const DtNetEntityStateUpdatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtEntityStateUpdatePdu (const DtEntityStateUpdatePdu & orig)
virtual	~DtEntityStateUpdatePdu () <i>destructor</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtEntityStateUpdatePdu</i>
static void	addCallback (DtExerciseConn * conn, DtEntityStateUpdatePduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtEntityStateUpdatePduCb cb, void* usr)
DtEntityStateUpdatePdu &	operator= (const DtEntityStateUpdatePdu & orig) <i>Assignment operator</i>
virtual void	setEntityID (const DtEntityIdentifier & id) <i>Set/Get entity ID</i>
virtual const DtEntityIdentifier &	entityID () const
virtual DtU8	numRecords () const <i>Get the number of articulation parameter records (this field is automatically set by the PDU when the articulated part list or attached part list is set)</i>
virtual void	setVelocity (DtConstVector velocity) <i>Set/Get velocity</i>
virtual DtConstVector	velocity () const
virtual void	setLocation (DtConstVector coords) <i>Set/Get location</i>
virtual DtConstVector	location () const
virtual void	setOrientation (const DtTaitBryan & orientation) <i>Set/Get orientation</i>
virtual const DtTaitBryan &	orientation () const

virtual void	setAppearance (DtU32 app) <i>Set/Get appearance (as a 32 bit integer)</i>
virtual DtU32	appearance () const
virtual void	setAppearanceAttribute (const DtAppearAttribute attr, const DtAppearAttrVal val) <i>Set/Get appearance by attribute</i>
virtual DtAppearAttrVal	appearanceAttribute (const DtAppearAttribute attr) const
virtual DtBoolean	setArtPartParameters (const DtArtPartList & partList) <i>Set/Get articulated parts parameters</i>
virtual DtBoolean	getArtPartParameters (DtArtPartList & partList) const
virtual DtBoolean	setAttPartParameters (const DtAttachedPartList & partList) <i>Set/Get attached parts parameters</i>
virtual DtBoolean	getAttPartParameters (DtAttachedPartList & partList) const
virtual void	printData () const <i>Print the data contained by this PDU</i>

Public

- The articulated parts and attached parts information are both optional

Protected Fields

static int	theStaticSize <i>Size of PDU with zero parameter records</i>
int	myNumArtPartParams <i>Number of articulated part parameters for this PDU</i>
int	myNumAttPartParams <i>Number of attached part parameters for this PDU</i>
DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>
DtVector	myVector
DtTaitBryan	myEulerAngles

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
virtual DtNetEntityStateUpdatePdu*	netPduStruct () const <i>Casts the netPdu element to our network struct</i>
DtEntityStateUpdatePdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type</i>
virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>
virtual void	setNumRecords (DtU8 value) <i>Set number of records</i>

virtual int

artParmOffset () const
Returns the offset (in bytes) of a field in the variable data region

virtual int

attParmOffset () const

Inherited from [DtPdu](#):

Public Methods

virtual void* **packet**() const

virtual int **status**() const

virtual void **setVersion**(int vers)

virtual int **protocolVersion**() const

virtual void **setExerciseld**(int id)

virtual int **exerciseld**() const

virtual [DtPduKind](#) **kind**() const

virtual [DtProtocolFamily](#) **protocolFamily**() const

virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

virtual DtTime **timeStamp**() const

virtual [DtTimeStampType](#) **timeStampType**() const

virtual DtTime **guessTimeValid**(DtTime referenceTime) const

virtual void **print**() const

virtual void **printHeader**() const

inline int **exerciseld**() const

inline [DtPduKind](#) **kind**() const

inline DtNetPduHeader* **netHeader**() const

inline void* **packet**() const

inline [DtProtocolFamily](#) **protocolFamily**() const

inline int **protocolVersion**() const

inline void **setExerciseld**(int id)

inline void **setStatus**(int okStatus)

inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)

inline void **setVersion**(int vers)

inline int **status**() const

- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)

- virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtEntityStateUpdatePdu**(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors

DtEntityStateUpdatePdu(const DtNetEntityStateUpdatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

- DtEntityStateUpdatePdu**(const [DtEntityStateUpdatePdu](#)& orig)

- virtual ~DtEntityStateUpdatePdu**()
destructor

static [DtPdu](#)* **create**(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Create a DtEntityStateUpdatePdu. Caller is responsible for deletion.

- static void addCallback**([DtExerciseConn](#)* conn, DtEntityStateUpdatePduCb cb, void* usr)
Add (register)/Remove (deregister) function to be called when pdu is received

static void removeCallback([DtExerciseConn](#)* conn, DtEntityStateUpdatePduCb cb, void* usr)

- [DtEntityStateUpdatePdu](#)& **operator=**(const [DtEntityStateUpdatePdu](#)& orig)
Assignment operator

- virtual void setEntityID**(const [DtEntityIdentifier](#)& id)
Set/Get entity ID

- virtual const** [DtEntityIdentifier](#)& **entityID**() **const**

- virtual DtU8 numRecords**() **const**
Get the number of articulation parameter records (this field is automatically set by the PDU when the articulated part list or attached part list is set)

- virtual void setVelocity**(DtConstVector velocity)
Set/Get velocity

- virtual DtConstVector velocity**() **const**

- `virtual void setLocation(DtConstVector coords)`
Set/Get location
- `virtual DtConstVector location() const`
- `virtual void setOrientation(const DtTaitBryan& orientation)`
Set/Get orientation
- `virtual const DtTaitBryan& orientation() const`
- `virtual void setAppearance(DtU32 app)`
Set/Get appearance (as a 32 bit integer)
- `virtual DtU32 appearance() const`
- `virtual void setAppearanceAttribute( const DtAppearAttribute attr, const DtAppearAttrVal val)`
Set/Get appearance by attribute
- `virtual DtAppearAttrVal appearanceAttribute( const DtAppearAttribute attr) const`

The articulated parts and attached parts information are both optional. Either or both can be set.

- `virtual DtBoolean setArtPartParameters(const DtArtPartList& partList)`
Set/Get articulated parts parameters. Parameters are passed in the form of an articulated parts list. Returns `DtFalse` if the operation could not be performed.
- `virtual DtBoolean getArtPartParameters(DtArtPartList& partList) const`
- `virtual DtBoolean setAttPartParameters(const DtAttachedPartList& partList)`
Set/Get attached parts parameters. Parameters are passed in the form of an attached parts list. Returns `DtFalse` if the operation could not be performed.
- `virtual DtBoolean getAttPartParameters(DtAttachedPartList& partList) const`
- `virtual void printData() const`
Print the data contained by this PDU
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetEntityStateUpdatePdu* netPduStruct() const`
Casts the netPdu element to our network struct
- `DtEntityStateUpdatePdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
- `virtual char* var() const`
Returns a pointer to the first byte of variable length data
- `virtual void setNumRecords(DtU8 value)`
Set number of records

- `virtual int artParmOffset() const`
Returns the offset (in bytes) of a field in the variable data region
- `virtual int attParmOffset() const`
- `static int theStaticSize`
Size of PDU with zero parameter records
- `int myNumArtPartParams`
Number of articulated part parameters for this PDU
- `int myNumAttPartParams`
Number of attached part parameters for this PDU
- `DtEntityIdentifier myEntityID`
Temporary objects for reference returns
- `DtVector myVector`
- `DtTaitBryan myEulerAngles`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file `designatepdu.h`:

class DtDesignatorPdu: public DtPdu

Inheritance:

DtDesignatorPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtDesignatorPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtDesignatorPdu (const DtNetDesignatorPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static void	addCallback (DtExerciseConn *conn, DtDesignatorPduCb cb, void *usr)
static void	removeCallback (DtExerciseConn *conn, DtDesignatorPduCb cb, void *usr)
const DtEntityIdentifier &	designatingEntityId () const
DtDesignatorCodeName	codeName () const
const DtEntityIdentifier &	designatedEntityId () const
DtDesignatorCode	designatorCode () const
float	power () const
float	wavelength () const
DtConstVector	designatorSpotRelative () const
DtConstVector	designatorSpotWorld () const
DtDeadReckonTypes	deadReckoningAlgorithm () const
DtConstVector	linearAcceleration () const
void	setDesignatingEntityId (const DtEntityIdentifier &id)
void	setCodeName (DtDesignatorCodeName name)
void	setDesignatedEntityId (const DtEntityIdentifier &id)
void	setDesignatorCode (DtDesignatorCode code)
void	setPower (float power)
void	setWavelength (float wavelength)

void	setDesignatorSpotRelative (DtConstVector loc)
void	setDesignatorSpotWorld (DtConstVector loc)
void	setDeadReckoningAlgorithm (DtDeadReckonTypes alg)
void	setLinearAcceleration (DtConstVector acc)
void	printData () const

Protected Fields

DtEntityIdentifier	theId
DtVector	loc

Protected Methods

DtPduKind	internalGetPduKind () const
---------------------------	------------------------------------

Inherited from [DtPdu](#):

Public Methods

DtPdu	operator= (const DtPdu &pdu)
virtual void*	packet () const
virtual int	status () const
virtual void	setVersion (int vers)
virtual int	protocolVersion () const
virtual void	setExerciseld (int id)
virtual int	exerciseld () const
virtual DtPduKind	kind () const
virtual DtProtocolFamily	protocolFamily () const
virtual void	setTimeStamp (DtTime time, DtTimeStampType type = DtTimeStampRelative)
virtual DtTime	timeStamp () const
virtual DtTimeStampType	timeStampType () const
virtual DtTime	guessTimeValid (DtTime referenceTime) const
virtual void	print () const
virtual void	printHeader () const

- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
- virtual DtNetPduHeader* **netHeader**() const
- virtual void **initPdu**(int size, DtBufferPtr buffer)
- virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
- virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
- virtual void **sizeChanged**(int size)
- virtual void **clearData**()
- virtual void **initHeader**([DtPduKind](#) kind, int size)

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtDesignatorPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtDesignatorPdu(const DtNetDesignatorPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static [DtPdu](#)* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- static void addCallback([DtExerciseConn](#) *conn, DtDesignatorPduCb cb, void *usr)
- static void removeCallback([DtExerciseConn](#) *conn, DtDesignatorPduCb cb, void *usr)
- const [DtEntityIdentifier](#)& designatingEntityId() const

- [DtDesignatorCodeName](#) codeName() const
- const [DtEntityIdentifier](#)& designatedEntityId() const
- [DtDesignatorCode](#) designatorCode() const
- float power() const
- float wavelength() const
- DtConstVector designatorSpotRelative() const
- DtConstVector designatorSpotWorld() const
- [DtDeadReckonTypes](#) deadReckoningAlgorithm() const
- DtConstVector linearAcceleration() const
- void setDesignatingEntityId(const [DtEntityIdentifier](#) &id)
- void setCodeName([DtDesignatorCodeName](#) name)
- void setDesignatedEntityId(const [DtEntityIdentifier](#) &id)
- void setDesignatorCode([DtDesignatorCode](#) code)
- void setPower(float power)
- void setWavelength(float wavelength)
- void setDesignatorSpotRelative(DtConstVector loc)
- void setDesignatorSpotWorld(DtConstVector loc)
- void setDeadReckoningAlgorithm([DtDeadReckonTypes](#) alg)
- void setLinearAcceleration(DtConstVector acc)
- void printData() const
- [DtPduKind](#) internalGetPduKind() const
- [DtEntityIdentifier](#) theId
- [DtVector](#) loc

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtDesignatorCodeName

 DtDesignatorNameOther

 DtDesignatorNameTBD

Documentation

 DtDesignatorNameOther

 DtDesignatorNameTBD

In file disenums.h:

enum DtDesignatorCode

 DtDesignatorCodeOther

Documentation

 DtDesignatorCodeOther

[Alphabetic index](#) [Hierarchy of classes](#)

In file colliselpdu.h:

class DtCollisionElasticPdu: public DtPdu

Inheritance:

DtCollisionElasticPdu - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtCollisionElasticPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtCollisionElasticPdu (const DtNetCollisionElasticPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtCollisionElasticPdu (const DtCollisionElasticPdu & orig)
virtual	~DtCollisionElasticPdu () <i>destructor</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtCollisionElasticPdu</i>
static void	addCallback (DtExerciseConn * conn, DtCollisionElasticPduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtCollisionElasticPduCb cb, void* usr)
DtCollisionElasticPdu &	operator= (const DtCollisionElasticPdu & orig) <i>Assignment operator</i>
virtual void	setIssuingEntityID (const DtEntityIdentifier & entity) <i>Set/Get issuing entity identifier</i>
virtual const DtEntityIdentifier &	issuingEntityID () const
virtual void	setCollidingEntityID (const DtEntityIdentifier & entity) <i>Set/Get colliding entity identifier</i>
virtual const DtEntityIdentifier &	collidingEntityID () const
virtual void	setEventID (const DtEventID& event) <i>Set/Get event ID</i>
virtual const DtEventID&	eventID () const
virtual void	setContactVelocity (DtConstVector velocity) <i>Set/Get contact velocity</i>
virtual DtConstVector	contactVelocity () const
virtual void	setMass (DtFloat32 value) <i>Set/Get mass</i>
virtual DtFloat32	mass () const

virtual void	setLocationOfImpact (DtConstVector coords) <i>Set/Get location of impact</i>
virtual DtConstVector	locationOfImpact () const
virtual void	setCollisionInterResultXX (DtFloat32 result) <i>Set/Get collision intermediate result XX</i>
virtual DtFloat32	collisionInterResultXX () const
virtual void	setCollisionInterResultXY (DtFloat32 result) <i>Set/Get collision intermediate result XY</i>
virtual DtFloat32	collisionInterResultXY () const
virtual void	setCollisionInterResultXZ (DtFloat32 result) <i>Set/Get collision intermediate result XZ</i>
virtual DtFloat32	collisionInterResultXZ () const
virtual void	setCollisionInterResultYY (DtFloat32 result) <i>Set/Get collision intermediate result YY</i>
virtual DtFloat32	collisionInterResultYY () const
virtual void	setCollisionInterResultYZ (DtFloat32 result) <i>Set/Get collision intermediate result YZ</i>
virtual DtFloat32	collisionInterResultYZ () const
virtual void	setCollisionInterResultZZ (DtFloat32 result) <i>Set/Get collision intermediate result ZZ</i>
virtual DtFloat32	collisionInterResultZZ () const
virtual void	setUnitSurfaceNormal (DtConstVector coords) <i>Set/Get unit surface normal</i>
virtual DtConstVector	unitSurfaceNormal () const
virtual void	setCoefficientOfRestitution (DtFloat32 coeff) <i>Set/Get coefficient of restitution</i>
virtual DtFloat32	coefficientOfRestitution () const
virtual void	printData () const <i>Print the data contained by this PDU</i>

Protected Fields

static int	theStaticSize <i>Size of PDU with zero data bytes</i>
DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>
DtEventID	myEventID
DtVector	myVector

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
virtual DtNetCollisionElasticPdu*	netPduStruct () const <i>Casts the netPdu element to our network struct Non-virtual since you can't override based on return type</i>

Inherited from [DtPdu](#):

Public Methods

-  virtual void* **packet**() const
-  virtual int **status**() const
-  virtual void **setVersion**(int vers)
-  virtual int **protocolVersion**() const
-  virtual void **setExerciseld**(int id)
-  virtual int **exerciseld**() const
-  virtual [DtPduKind](#) **kind**() const
-  virtual [DtProtocolFamily](#) **protocolFamily**() const
-  virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
-  virtual DtTime **timeStamp**() const
-  virtual [DtTimeStampType](#) **timeStampType**() const
-  virtual DtTime **guessTimeValid**(DtTime referenceTime) const
-  virtual void **print**() const
-  virtual void **printHeader**() const
-  inline int **exerciseld**() const
-  inline [DtPduKind](#) **kind**() const
-  inline DtNetPduHeader* **netHeader**() const
-  inline void* **packet**() const
-  inline [DtProtocolFamily](#) **protocolFamily**() const
-  inline int **protocolVersion**() const
-  inline void **setExerciseld**(int id)
-  inline void **setStatus**(int okStatus)
-  inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
-  inline void **setVersion**(int vers)
-  inline int **status**() const

- inline DtTime **timeStamp()** const
- inline [DtTimeStampType](#) **timeStampType()** const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader()** const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep()** const
- virtual int **length()** const
- inline const char* **netRep()** const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)

- virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtCollisionElasticPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Constructors

- `DtCollisionElasticPdu(const DtNetCollisionElasticPdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`

- `DtCollisionElasticPdu(const DtCollisionElasticPdu& orig)`

- `virtual ~DtCollisionElasticPdu()`
destructor

- `static DtPdu* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
Create a DtCollisionElasticPdu. Caller is responsible for deletion.

- `static void addCallback(DtExerciseConn* conn, DtCollisionElasticPduCb cb, void* usr)`
Add (register)/Remove (deregister) function to be called when pdu is received

- `static void removeCallback(DtExerciseConn* conn, DtCollisionElasticPduCb cb, void* usr)`

- `DtCollisionElasticPdu& operator=(const DtCollisionElasticPdu& orig)`
Assignment operator

- `virtual void setIssuingEntityID(const DtEntityIdentifier& entity)`
Set/Get issuing entity identifier

- `virtual const DtEntityIdentifier& issuingEntityID() const`

- `virtual void setCollidingEntityID(const DtEntityIdentifier& entity)`
Set/Get colliding entity identifier

- `virtual const DtEntityIdentifier& collidingEntityID() const`

- `virtual void setEventID(const DtEventID& event)`
Set/Get event ID

- `virtual const DtEventID& eventID() const`

- `virtual void setContactVelocity(DtConstVector velocity)`
Set/Get contact velocity
- `virtual DtConstVector contactVelocity() const`
- `virtual void setMass(DtFloat32 value)`
Set/Get mass
- `virtual DtFloat32 mass() const`
- `virtual void setLocationOfImpact(DtConstVector coords)`
Set/Get location of impact
- `virtual DtConstVector locationOfImpact() const`
- `virtual void setCollisionInterResultXX(DtFloat32 result)`
Set/Get collision intermediate result XX
- `virtual DtFloat32 collisionInterResultXX() const`
- `virtual void setCollisionInterResultXY(DtFloat32 result)`
Set/Get collision intermediate result XY
- `virtual DtFloat32 collisionInterResultXY() const`
- `virtual void setCollisionInterResultXZ(DtFloat32 result)`
Set/Get collision intermediate result XZ
- `virtual DtFloat32 collisionInterResultXZ() const`
- `virtual void setCollisionInterResultYY(DtFloat32 result)`
Set/Get collision intermediate result YY
- `virtual DtFloat32 collisionInterResultYY() const`
- `virtual void setCollisionInterResultYZ(DtFloat32 result)`
Set/Get collision intermediate result YZ
- `virtual DtFloat32 collisionInterResultYZ() const`
- `virtual void setCollisionInterResultZZ(DtFloat32 result)`
Set/Get collision intermediate result ZZ
- `virtual DtFloat32 collisionInterResultZZ() const`
- `virtual void setUnitSurfaceNormal(DtConstVector coords)`
Set/Get unit surface normal
- `virtual DtConstVector unitSurfaceNormal() const`
- `virtual void setCoefficientOfRestitution(DtFloat32 coeff)`
Set/Get coefficient of restitution
- `virtual DtFloat32 coefficientOfRestitution() const`
- `virtual void printData() const`
Print the data contained by this PDU

• **virtual [DtPduKind](#) internalGetPduKind() const**

Returns the internal PDU kind

• **virtual [DtNetCollisionElasticPdu*](#) netPduStruct() const**

Casts the netPdu element to our network struct Non-virtual since you can't override based on return type

• **[DtCollisionElasticPdu*](#) self() const**

Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type

• **static int theStaticSize**

Size of PDU with zero data bytes

• **[DtEntityIdentifier](#) myEntityID**

Temporary objects for reference returns

• **DtEventID myEventID**

• **[DtVector](#) myVector**

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file arealobjstpdu.h:

class DtArealObjectStatePdu: public DtPdu

Inheritance:

DtArealObjectStatePdu - DtPdu - DtBaseMsg

Public Methods

	DtArealObjectStatePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtArealObjectStatePdu (const DtNetArealObjectStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtArealObjectStatePdu (const DtArealObjectStatePdu & orig)
 virtual	~DtArealObjectStatePdu () <i>destructor</i>
 static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtArealObjectStatePdu</i>
 static void	addCallback (DtExerciseConn * conn, DtArealObjectStatePduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
 static void	removeCallback (DtExerciseConn * conn, DtArealObjectStatePduCb cb, void* usr)
 DtArealObjectStatePdu &	operator= (const DtArealObjectStatePdu & orig) <i>Assignment operator</i>
 virtual void	setObjectID (const DtEntityIdentifier & id) <i>Set/Get object ID</i>
 virtual const DtEntityIdentifier &	objectID () const
 virtual void	setReferencedObjectID (const DtEntityIdentifier & id) <i>Set/Get referenced object ID</i>
 virtual const DtEntityIdentifier &	referencedObjectID () const
 virtual void	setUpdateNumber (DtU16 field) <i>Set/Get update number</i>
 virtual DtU16	updateNumber () const
 virtual void	setForceID (DtForceType force) <i>Set/Get the force ID</i>
 virtual DtForceType	forceID () const
 virtual void	setModifications (DtU8 value) <i>Set/Get the modifications (by 8-bit integer)</i>

virtual DtU8	modifications () const
virtual void	setModificationsField (DtModificationField field, DtBoolean value) <i>Set/Get the modifications (by field)</i>
virtual DtBoolean	modificationsField (DtModificationField field) const
virtual void	setObjectType (const DtObjectType & type) <i>Set/Get the object type</i>
virtual const DtObjectType &	objectType () const
virtual void	setAppearance (DtU16 app) <i>Set/Get appearance, the lower 16 bits (as an integer)</i>
virtual DtU16	appearance () const
virtual void	setAppearanceAttribute (DtAppearAttribute attr, DtAppearAttrVal val) <i>Set/Get appearance by attribute for the lower 16 bits</i>
virtual DtAppearAttrVal	appearanceAttribute (DtAppearAttribute attr) const
virtual void	setSpecificAppearance (DtU32 app) <i>Set/Get specific appearance, the upper 32 bits (as a 32 bit integer)</i>
virtual DtU32	specificAppearance () const
virtual void	setSpecificAppearanceAttribute (const DtAppearAttribute attr, const DtAppearAttrVal val) <i>Set/Get specific appearance by attribute, the upper 32 bits</i>
virtual DtAppearAttrVal	specificAppearanceAttribute (const DtAppearAttribute attr) const
virtual DtBoolean	setNumPoints (DtU16 num) <i>Set/Get the number of points</i>
virtual DtU16	numPoints () const
virtual void	setRequesterID (const DtSimulationAddress& type) <i>Set/Get the requester ID</i>
virtual const DtSimulationAddress&	requesterID () const
virtual void	setReceivingID (const DtSimulationAddress& type) <i>Set/Get the receiving ID</i>
virtual const DtSimulationAddress&	receivingID () const
virtual DtBoolean	setPointLocation (int index, DtConstVector init) <i>Set/Get a point location</i>
virtual DtBoolean	getPointLocation (int index, DtVectorRef record) const
virtual void	printData () const <i>Print the data contained by this PDU</i>

Protected Fields

static int	theStaticSize <i>Size of PDU with zero parameter records</i>
DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>
DtObjectType	myObjectType
DtSimulationAddress	mySimulationAddress

Protected Methods

virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
virtual DtNetArealObjectStatePdu*	netPduStruct () const <i>Casts the netPdu element to our network struct</i>
virtual DtArealObjectStatePdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type</i>
virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const

- inline void **setExerciseId**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**

int myMsgSize

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

DtArealObjectStatePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Constructors

DtArealObjectStatePdu(const DtNetArealObjectStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

DtArealObjectStatePdu(const [DtArealObjectStatePdu](#)& orig)

virtual ~DtArealObjectStatePdu()

destructor

static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtArealObjectStatePdu. Caller is responsible for deletion.

static void addCallback([DtExerciseConn](#)* conn, DtArealObjectStatePduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

static void removeCallback([DtExerciseConn](#)* conn, DtArealObjectStatePduCb cb, void* usr)

[DtArealObjectStatePdu](#)& operator=(const [DtArealObjectStatePdu](#)& orig)

Assignment operator

virtual void setObjectID(const [DtEntityIdentifier](#)& id)

Set/Get object ID

virtual const [DtEntityIdentifier](#)& objectID() const

virtual void setReferencedObjectID(const [DtEntityIdentifier](#)& id)

Set/Get referenced object ID

virtual const [DtEntityIdentifier](#)& referencedObjectID() const

virtual void setUpdateNumber(DtU16 field)

Set/Get update number

virtual DtU16 updateNumber() const

virtual void setForceID([DtForceType](#) force)

Set/Get the force ID

virtual [DtForceType](#) forceID() const

virtual void setModifications(DtU8 value)

Set/Get the modifications (by 8-bit integer)

virtual DtU8 modifications() const

virtual void setModificationsField([DtModificationField](#) field, [DtBoolean](#) value)

Set/Get the modifications (by field)

virtual [DtBoolean](#) modificationsField([DtModificationField](#) field) const

virtual void setObjectType(const [DtObjectType](#)& type)

Set/Get the object type

virtual const [DtObjectType](#)& objectType() const

virtual void setAppearance(DtU16 app)

Set/Get appearance, the lower 16 bits (as an integer)

virtual DtU16 appearance() const

virtual void setAppearanceAttribute([DtAppearAttribute](#) attr, [DtAppearAttrVal](#) val)

Set/Get appearance by attribute for the lower 16 bits

virtual [DtAppearAttrVal](#) appearanceAttribute([DtAppearAttribute](#) attr) const

virtual void setSpecificAppearance(DtU32 app)

Set/Get specific appearance, the upper 32 bits (as a 32 bit integer)

virtual DtU32 specificAppearance() const

virtual void setSpecificAppearanceAttribute(const [DtAppearAttribute](#) attr, const [DtAppearAttrVal](#) val)

Set/Get specific appearance by attribute, the upper 32 bits

virtual [DtAppearAttrVal](#) specificAppearanceAttribute(const [DtAppearAttribute](#) attr) const

virtual [DtBoolean](#) setNumPoints(DtU16 num)

Set/Get the number of points

virtual DtU16 numPoints() const

- `virtual void setRequesterID(const DtSimulationAddress& type)`
Set/Get the requester ID
- `virtual const DtSimulationAddress& requesterID() const`
- `virtual void setReceivingID(const DtSimulationAddress& type)`
Set/Get the receiving ID
- `virtual const DtSimulationAddress& receivingID() const`
- `virtual DtBoolean setPointLocation(int index, DtConstVector init)`
Set/Get a point location. Returns DtFalse if the index is out of range.
- `virtual DtBoolean getPointLocation(int index, DtVectorRef record) const`
- `virtual void printData() const`
Print the data contained by this PDU
- `virtual DtPduKind internalGetPduKind() const`
Returns the internal PDU kind
- `virtual DtNetArealObjectStatePdu* netPduStruct() const`
Casts the netPdu element to our network struct
- `DtArealObjectStatePdu* self() const`
Casts the 'this' pointer for changing elements in a const function Non-virtual since you can't override based on return type
- `virtual char* var() const`
Returns a pointer to the first byte of variable length data
- `static int theStaticSize`
Size of PDU with zero parameter records
- `DtEntityIdentifier myEntityID`
Temporary objects for reference returns
- `DtObjectType myObjectType`
- `DtSimulationAddress mySimulationAddress`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file espdu.h:

class DtEntityStatePdu: public DtStateMsg

Inheritance:

DtEntityStatePdu - DtStateMsg - DtPdu - DtBaseMsg

Public Methods

	DtEntityStatePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtEntityStatePdu (const DtNetEntityStatePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static <u>DtPdu</u> *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
static void	addCallback (<u>DtExerciseConn</u> *conn, DtEntityStatePduCb cb, void *usr)
static void	removeCallback (<u>DtExerciseConn</u> *conn, DtEntityStatePduCb cb, void *usr)
	DtEntityStatePdu (const <u>DtEntityIdentifier</u> &id, const <u>DtEntityType</u> &type, const <u>DtEntityType</u> &guise, <u>DtDeadReckonTypes</u> algorithm, <u>DtForceType</u> forceld, const DtArtPartSpec *specList, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtEntityStatePdu (const <u>DtEntityIdentifier</u> &id, const <u>DtEntityType</u> &type, const <u>DtEntityType</u> &guise, <u>DtDeadReckonTypes</u> algorithm, <u>DtForceType</u> forceld, const DtArtPartSpec *artList, int *stationList, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtEntityStatePdu (const DtArtPartSpec *specList, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
const <u>DtEntityIdentifier</u> &	id () const
<u>DtForceType</u>	forceld () const
int	numArtParams () const
int	numAttachedParams () const
int	numArtPlusAttachedParams () const
const <u>DtEntityType</u> &	entityType () const
const <u>DtEntityType</u> &	guise () const
DtConstVector	velocity () const
DtConstVector	location () const
const <u>DtTaitBryan</u> &	orientation () const
DtU32	appearance () const

DtAppearAttrVal	appearanceAttribute (const DtAppearAttribute &attr) const
DtDeadReckonTypes	algorithm () const
DtConstVector	acceleration () const
DtConstVector	rotationalVelocity () const
int	markingCharSet () const
const char*	markingText () const
DtU32	capabilities () const
const DtArtPartSpec*	artPartSpecs () const
const DtArtPartList &	artPartList () const <i>returns a DtArtPartList containing the parts described in this PDU</i>
int*	attachedPartStations () const
int	getArtParamVal (int paramIndex, DtFloat32 *value) const
int	getArtParamVal (int partType, int param, DtFloat32 *value) const
int	getArtParamChange (int paramIndex, int *change) const
int	getArtParamChange (int partType, int param, int *change) const
int	attachedPartChange (int station) const
int	getAttachedType (int station, DtEntityType *type) const
void	setId (const DtEntityIdentifier &id)
void	setForceId (DtForceType force)
void	setEntityType (const DtEntityType &type)
void	setGuise (const DtEntityType &type)
void	setVelocity (DtConstVector Velocity)
void	setLocation (DtConstVector Location)
void	setOrientation (const DtTaitBryan &Orient)
void	setAppearance (DtU32 app)
void	setAppearanceAttribute (const DtAppearAttribute attr, const DtAppearAttrVal val)
void	setAlgorithm (DtDeadReckonTypes alg)
void	setAcceleration (DtConstVector Acceleration)
void	setRotationalVelocity (DtConstVector BodyRates)
void	setMarkingText (const char *text)
void	setMarkingCharSet (int charSet)
void	setCapabilities (DtU32 cap)
int	setArtParamVal (int partType, int param, DtFloat32 value, int change = DtINCREMENT)
int	setArtParamVal (int paramIndex, DtFloat32 value, int change = DtINCREMENT)
void	initArtAndAttachedParts (const DtArtPartSpec *artList, const int* stationList)
void	initArtAndAttachedParts (const DtArtPartList &apList, const int* stationList)

void	initArtParts (const DtArtPartList &apList)
int	setAttachedPartChange (int station, int change)
int	setAttachedPartType (int station, const DtEntityType &type)
void	printData () const

Protected Fields

DtEntityIdentifier	myId
DtEntityType	type
DtEntityType	theGuise
DtVector	vel
DtVector	loc
DtVector	acc
DtTaitBryan	ori
DtBodyRates	rvel
char	markingBuff [DtMaxEntityMarkingLength+1]
DtArtPartList	myArtPartList <i>Does not store state</i>

Protected Methods

DtPduKind	internalGetPduKind () const
void	setNumArtPlusAttachedParams (int num)
DtNetArtPartRecord*	artPartRecord (int partType, int param) const
DtNetArtPartRecord*	attachedPartRecord (int station) const
void	setArtParamVal (DtNetArtPartRecord *part, DtFloat32 value, int change)
float	getArtParamVal (DtNetArtPartRecord *part) const

Inherited from [DtStateMsg](#):

Public Methods

DtStateMsg &	operator= (const DtStateMsg & orig)
------------------------------	--

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

• `int myOKStatus`

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
- void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
- virtual void **changeSize**(int size)
- virtual void **insertBytes**(int offset, int numBytes)

- virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtEntityStatePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtEntityStatePdu(const DtNetEntityStatePdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `static void addCallback(DtExerciseConn *conn, DtEntityStatePduCb cb, void *usr)`
- `static void removeCallback(DtExerciseConn *conn, DtEntityStatePduCb cb, void *usr)`
- `DtEntityStatePdu(const DtEntityIdentifier &id, const DtEntityType &type, const DtEntityType &guise, DtDeadReckonTypes algorithm, DtForceType forceId, const DtArtPartSpec *specList, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtEntityStatePdu(const DtEntityIdentifier &id, const DtEntityType &type, const DtEntityType &guise, DtDeadReckonTypes algorithm, DtForceType forceId, const DtArtPartSpec *artList, int *stationList, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtEntityStatePdu(const DtArtPartSpec *specList, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `const DtEntityIdentifier& id() const`
- `DtForceType forceId() const`
- `int numArtParams() const`
- `int numAttachedParams() const`
- `int numArtPlusAttachedParams() const`
- `const DtEntityType& entityType() const`
- `const DtEntityType& guise() const`
- `DtConstVector velocity() const`
- `DtConstVector location() const`
- `const DtTaitBryan& orientation() const`
- `DtU32 appearance() const`

- [DtAppearAttrVal](#) appearanceAttribute(const [DtAppearAttribute](#) &attr) const
- [DtDeadReckonTypes](#) algorithm() const
- DtConstVector acceleration() const
- DtConstVector rotationalVelocity() const
- int markingCharSet() const
- const char* markingText() const
- DtU32 capabilities() const
- const DtArtPartSpec* artPartSpecs() const
- const [DtArtPartList](#)& artPartList() const
returns a DtArtPartList containing the parts described in this PDU
- int* attachedPartStations() const
- int getArtParamVal(int paramIndex, DtFloat32 *value) const
- int getArtParamVal(int partType, int param, DtFloat32 *value) const
- int getArtParamChange(int paramIndex, int *change) const
- int getArtParamChange(int partType, int param, int *change) const
- int attachedPartChange(int station) const
- int getAttachedType(int station, [DtEntityType](#) *type) const
- void setId(const [DtEntityIdentifier](#) &id)
- void setForceId([DtForceType](#) force)
- void setEntityType(const [DtEntityType](#) &type)
- void setGuise(const [DtEntityType](#) &type)
- void setVelocity(DtConstVector Velocity)
- void setLocation(DtConstVector Location)
- void setOrientation(const [DtTaitBryan](#) &Orient)
- void setAppearance(DtU32 app)
- void setAppearanceAttribute(const [DtAppearAttribute](#) attr, const [DtAppearAttrVal](#) val)
- void setAlgorithm([DtDeadReckonTypes](#) alg)
- void setAcceleration(DtConstVector Acceleration)
- void setRotationalVelocity(DtConstVector BodyRates)
- void setMarkingText(const char *text)
- void setMarkingCharSet(int charSet)

- `void setCapabilities(DtU32 cap)`
- `int setArtParamVal(int partType, int param, DtFloat32 value, int change = DtINCREMENT)`
- `int setArtParamVal(int paramIndex, DtFloat32 value, int change = DtINCREMENT)`
- `void initArtAndAttachedParts(const DtArtPartSpec *artList, const int* stationList)`
- `void initArtAndAttachedParts(const DtArtPartList &apList, const int* stationList)`
- `void initArtParts(const DtArtPartList &apList)`
- `int setAttachedPartChange(int station, int change)`
- `int setAttachedPartType(int station, const DtEntityType &type)`
- `void printData() const`
- `DtPduKind internalGetPduKind() const`
- `void setNumArtPlusAttachedParams(int num)`
- `DtNetArtPartRecord* artPartRecord(int partType, int param) const`
- `DtNetArtPartRecord* attachedPartRecord(int station) const`
- `void setArtParamVal(DtNetArtPartRecord *part, DtFloat32 value, int change)`
- `float getArtParamVal(DtNetArtPartRecord *part) const`
- `DtEntityIdentifier myId`
- `DtEntityType type`
- `DtEntityType theGuise`
- `DtVector vel`
- `DtVector loc`
- `DtVector acc`
- `DtTaitBryan ori`
- `DtBodyRates rvel`
- `char markingBuff[DtMaxEntityMarkingLength+1]`
- `DtArtPartList myArtPartList`

Does not store state

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file emissionpdu.h:

class DtEmissionPdu: public DtStateMsg

Inheritance:

DtEmissionPdu - DtStateMsg - DtPdu - DtBaseMsg

Public Methods

	DtEmissionPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtEmissionPdu (const DtNetEmissionPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static DtPdu *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static void	addCallback (DtExerciseConn *conn, DtEmissionPduCb cb, void *usr)
 static void	removeCallback (DtExerciseConn *conn, DtEmissionPduCb cb, void *usr)
 const DtEntityIdentifier &	emittingEntityId () const
 const DtEventID&	eventId () const
 DtEmissionPduType	stateUpdateIndicator () const
 int	numSystems () const
 int	systemDataWords (int sysNum, int *retcode = NULL) const
 int	numBeams (int sysNum, int *retcode = NULL) const
 DtEmitterSysName	systemName (int sysNum, int *retcode = NULL) const
 DtEmissionFunc	systemFunction (int sysNum, int *retcode = NULL) const
 int	emitterId (int sysNum, int *retcode = NULL) const
 DtConstVector	location (int sysNum, int *retcode = NULL) const
 int	beamDataWords (int sysNum, int beamNum, int *retcode = NULL) const
 int	beamId (int sysNum, int beamNum, int *retcode = NULL) const
 int	beamParameterIndex (int sysNum, int beamNum, int *retcode = NULL) const
 const DtFundamentalParameterData &	parameterData (int sysNum, int beamNum, int *retcode = NULL) const
 int	beamFunction (int sysNum, int beamNum, int *retcode = NULL) const
 int	numTargetsInBeam (int sysNum, int beamNum, int *retcode = NULL) const

-  [DtTrackingJamming](#)
-  unsigned long
-  const [DtTrackJamTarget](#)&
-  void
-  void
-  void
-  void
-  void
-  void
-  void
-  void
-  void
-  void
-  void
-  void
-  void
-  void
-  void
-  void
-  void

```

trackingAllTargets (int sysNum, int beamNum, int *retcode = NULL) const
jammingModeSequence (int sysNum, int beamNum, int *retcode = NULL)
const
trackJamTarget (int sysNum, int beamNum, int targetNum, int *retcode =
NULL) const
setEmittingEntityId (const DtEntityIdentifier &id)
setEventId (const DtEventID &id)
setStateUpdateIndicator (DtEmissionPduType indicator)
setNumSystems (int num)
setNumBeams (int sysNum, int numBeams, int *retcode = NULL)
setSystemName (int sysNum, DtEmitterSysName name, int *retcode = NULL)
setSystemFunction (int sysNum, DtEmissionFunc func, int *retcode = NULL)
setEmitterId (int sysNum, int id, int *retcode = NULL)
setLocation (int sysNum, DtConstVector loc, int *retcode = NULL)
setBeamId (int sysNum, int beamNum, int id, int *retcode = NULL)
setBeamParameterIndex (int sysNum, int beamNum, int bpIndex, int *retcode
= NULL)
setParameterData (int sysNum, int beamNum, const
DtFundamentalParameterData &data, int *retcode = NULL)
setBeamFunction (int sysNum, int beamNum, int func, int *retcode = NULL)
setNumTargetsInBeam (int sysNum, int beamNum, int num, int *retcode =
NULL)
setTrackingAllTargets (int sysNum, int beamNum, DtTrackingJamming
trackJamAll, int *retcode = NULL)
setJammingModeSequence (int sysNum, int beamNum, unsigned long seq,
int *retcode = NULL)
setTrackJamTarget (int sysNum, int beamNum, int targetNum, const
DtTrackJamTarget &targ, int *retcode = NULL)
printData () const

```

Protected Fields

 DtEntityIdentifier	cacheId
 DtEventID	cacheEvent
 DtVector	cacheLoc
 DtFundamentalParameterData	cacheData
 DtTrackJamTarget	cacheTarget

Protected Methods

 DtPduKind	internalGetPduKind () const
 DtNetSystemData*	emitterSystemAddr (int sysNum, int *retcode = NULL) const
 DtNetBeamData*	emitterBeamAddr (int sysNum, int beamNum, int *retcode = NULL) const
 DtNetEmitterTrackJam*	targetAddr (int sysNum, int beamNum, int targetNum, int *retcode = NULL) const
 int	computedSystemDataLength (int sysNum) const
 int	computedBeamDataLength (int sysNum, int beamNum) const
 void	printSystem (int i) const
 void	printBeam (int i, int j) const
 int	addressInPacket (void *addr) const
 int	computeStatus () const

Inherited from [DtStateMsg](#):

Public Methods

 [DtStateMsg](#)& **operator=**(const [DtStateMsg](#)& orig)

Inherited from [DtPdu](#):

Public Methods

-  virtual void* **packet**() const
-  virtual int **status**() const
-  virtual void **setVersion**(int vers)
-  virtual int **protocolVersion**() const
-  virtual void **setExerciseld**(int id)
-  virtual int **exerciseld**() const
-  virtual [DtPduKind](#) **kind**() const
-  virtual [DtProtocolFamily](#) **protocolFamily**() const
-  virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
-  virtual DtTime **timeStamp**() const
-  virtual [DtTimeStampType](#) **timeStampType**() const

- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtEmissionPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtEmissionPdu(const DtNetEmissionPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `static void addCallback(DtExerciseConn *conn, DtEmissionPduCb cb, void *usr)`
- `static void removeCallback(DtExerciseConn *conn, DtEmissionPduCb cb, void *usr)`

```

const DtEntityIdentifier& emittingEntityId() const

const DtEventID& eventId() const

DtEmissionPduType stateUpdateIndicator() const

int numSystems() const

int systemDataWords(int sysNum, int *retcode = NULL) const

int numBeams(int sysNum, int *retcode = NULL) const

DtEmitterSysName systemName(int sysNum, int *retcode = NULL) const

DtEmissionFunc systemFunction(int sysNum, int *retcode = NULL) const

int emitterId(int sysNum, int *retcode = NULL) const

DtConstVector location(int sysNum, int *retcode = NULL) const

int beamDataWords(int sysNum, int beamNum, int *retcode = NULL) const

int beamId(int sysNum, int beamNum, int *retcode = NULL) const

int beamParameterIndex(int sysNum, int beamNum, int *retcode = NULL) const

const DtFundamentalParameterData& parameterData(int sysNum, int beamNum, int *retcode =
NULL) const

int beamFunction(int sysNum, int beamNum, int *retcode = NULL) const

int numTargetsInBeam(int sysNum, int beamNum, int *retcode = NULL) const

DtTrackingJamming trackingAllTargets(int sysNum, int beamNum, int *retcode = NULL)
const

unsigned long jammingModeSequence(int sysNum, int beamNum, int *retcode = NULL) const

const DtTrackJamTarget& trackJamTarget(int sysNum, int beamNum, int targetNum, int
*retcode = NULL) const

void setEmittingEntityId(const DtEntityIdentifier &id)

void setEventId(const DtEventID &id)

void setStateUpdateIndicator(DtEmissionPduType indicator)

void setNumSystems(int num)

void setNumBeams(int sysNum, int numBeams, int *retcode = NULL)

void setSystemName(int sysNum, DtEmitterSysName name, int *retcode = NULL)

void setSystemFunction(int sysNum, DtEmissionFunc func, int *retcode = NULL)

void setEmitterId(int sysNum, int id, int *retcode = NULL)

void setLocation(int sysNum, DtConstVector loc, int *retcode = NULL)

void setBeamId(int sysNum, int beamNum, int id, int *retcode = NULL)

```

- `void setBeamParameterIndex(int sysNum, int beamNum, int bpIndex, int *retcode = NULL)`
- `void setParameterData(int sysNum, int beamNum, const DtFundamentalParameterData &data, int *retcode = NULL)`
- `void setBeamFunction(int sysNum, int beamNum, int func, int *retcode = NULL)`
- `void setNumTargetsInBeam(int sysNum, int beamNum, int num, int *retcode = NULL)`
- `void setTrackingAllTargets(int sysNum, int beamNum, DtTrackingJamming trackJamAll, int *retcode = NULL)`
- `void setJammingModeSequence(int sysNum, int beamNum, unsigned long seq, int *retcode = NULL)`
- `void setTrackJamTarget(int sysNum, int beamNum, int targetNum, const DtTrackJamTarget &targ, int *retcode = NULL)`
- `void printData() const`
- `DtPduKind internalGetPduKind() const`
- `DtEntityIdentifier cacheId`
- `DtEventID cacheEvent`
- `DtVector cacheLoc`
- `DtFundamentalParameterData cacheData`
- `DtTrackJamTarget cacheTarget`
- `DtNetSystemData* emitterSystemAddr(int sysNum, int *retcode = NULL) const`
- `DtNetBeamData* emitterBeamAddr(int sysNum, int beamNum, int *retcode = NULL) const`
- `DtNetEmitterTrackJam* targetAddr(int sysNum, int beamNum, int targetNum, int *retcode = NULL) const`
- `int computedSystemDataLength(int sysNum) const`
- `int computedBeamDataLength(int sysNum, int beamNum) const`
- `void printSystem(int i) const`
- `void printBeam(int i, int j) const`
- `int addressInPacket(void *addr) const`
- `int computeStatus() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtEmissionPduType

-
-  DtStateUpdate
 -  DtChangeDataUpdate
-

Documentation

-  DtStateUpdate
-  DtChangeDataUpdate

In file disenums.h:

enum DtEmitterSysName

 DtEmitterSysOther

Documentation

 DtEmitterSysOther

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtEmissionFunc

- DtEmissionFuncOther
 - DtEmissionFuncMultiFunction
 - DtEmissionFuncEarlyWarningSurveillance
 - DtEmissionFuncHeightFinding
 - DtEmissionFuncFireControl
 - DtEmissionFuncAcquisitionDetection
 - DtEmissionFuncTracking
 - DtEmissionFuncGuidanceIllumination
 - DtEmissionFuncFiringPointLaunchPointLocation
 - DtEmissionFuncRanging
 - DtEmissionFuncRadarAltimeter
 - DtEmissionFuncImaging
 - DtEmissionFuncMotionDetection
 - DtEmissionFuncNavigation
 - DtEmissionFuncWeather
 - DtEmissionFuncInstrumentation
 - DtEmissionFuncIdentificationClassification
 - DtEmissionFuncJammingNoise
 - DtEmissionFuncJammingDeception
 - DtEmissionFuncDecoy
 - DtEmissionFuncWeaponNonLethal
 - DtEmissionFuncWeaponLethal
-

Documentation

- DtEmissionFuncOther
- DtEmissionFuncMultiFunction
- DtEmissionFuncEarlyWarningSurveillance
- DtEmissionFuncHeightFinding
- DtEmissionFuncFireControl
- DtEmissionFuncAcquisitionDetection
- DtEmissionFuncTracking
- DtEmissionFuncGuidanceIllumination
- DtEmissionFuncFiringPointLaunchPointLocation
- DtEmissionFuncRanging
- DtEmissionFuncRadarAltimeter
- DtEmissionFuncImaging
- DtEmissionFuncMotionDetection
- DtEmissionFuncNavigation
- DtEmissionFuncWeather
- DtEmissionFuncInstrumentation
- DtEmissionFuncIdentificationClassification
- DtEmissionFuncJammingNoise
- DtEmissionFuncJammingDeception
- DtEmissionFuncDecoy
- DtEmissionFuncWeaponNonLethal
- DtEmissionFuncWeaponLethal

[Alphabetic index](#) [Hierarchy of classes](#)

In file emissionpdu.h:

class DtFundamentalParameterData

Public Fields

- float **frequency**
- float **frequencyRange**
- float **effectiveRadiatedPower**
- float **pulseRepetitionFrequency**
- float **pulseWidth**
- float **beamAzCenter**
- float **beamAzSweep**
- float **beamElCenter**
- float **beamElSweep**
- float **beamSweepSync**

Public Methods

- void **printData** () const
-

Documentation

- float **frequency**
- float **frequencyRange**
- float **effectiveRadiatedPower**
- float **pulseRepetitionFrequency**
- float **pulseWidth**
- float **beamAzCenter**
- float **beamAzSweep**
- float **beamElCenter**
- float **beamElSweep**

• float beamSweepSync

• void printData() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtTrackingJamming

 DtNotTrackingAllTargets

 DtTrackingAllTargets

Documentation

 DtNotTrackingAllTargets

 DtTrackingAllTargets

In file emissionpdu.h:

class DtTrackJamTarget

Public Fields

 [DtEntityIdentifier](#) entityId
 int emitterId
 int beamId

Public Methods

 void **printData** () const

Documentation

 [DtEntityIdentifier](#) entityId
 int emitterId
 int beamId
 void printData() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file `elecemitpdu.h`:

class [DtElectromagneticEmissionPdu](#): public [DtStateMsg](#)

Inheritance:

DtElectromagneticEmissionPdu - [DtStateMsg](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtElectromagneticEmissionPdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtElectromagneticEmissionPdu (const DtNetElectromagneticEmissionPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static <u>DtPdu</u> *	create (const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
 static void	addCallback (<u>DtExerciseConn</u> *conn, DtElectromagneticEmissionPduCb cb, void *usr)
 static void	removeCallback (<u>DtExerciseConn</u> *conn, DtElectromagneticEmissionPduCb cb, void *usr)
 const <u>DtEntityIdentifier</u> &	emittingEntityId () const
 const DtEventId&	eventId () const
 <u>DtEmissionPduType</u>	stateUpdateIndicator () const
 int	numSystems () const
 int	systemDataWords (int sysNum, int *retcode = NULL) const
 int	numBeams (int sysNum, int *retcode = NULL) const
 <u>DtEmitterSysName</u>	systemName (int sysNum, int *retcode = NULL) const
 <u>DtEmissionFunc</u>	systemFunction (int sysNum, int *retcode = NULL) const
 int	emitterId (int sysNum, int *retcode = NULL) const
 DtConstVector	location (int sysNum, int *retcode = NULL) const
 int	beamDataWords (int sysNum, int beamNum, int *retcode = NULL) const
 int	beamId (int sysNum, int beamNum, int *retcode = NULL) const
 int	beamParameterIndex (int sysNum, int beamNum, int *retcode = NULL) const

const DtFundamentalParameterData &	parameterData (int sysNum, int beamNum, int *retcode = NULL) const
int	beamFunction (int sysNum, int beamNum, int *retcode = NULL) const
int	numTargetsInBeam (int sysNum, int beamNum, int *retcode = NULL) const
DtTrackingJamming	trackingAllTargets (int sysNum, int beamNum, int *retcode = NULL) const
unsigned long	jammingModeSequence (int sysNum, int beamNum, int *retcode = NULL) const
const DtTrackJamTarget &	trackJamTarget (int sysNum, int beamNum, int targetNum, int *retcode = NULL) const
void	setEmittingEntityId (const DtEntityIdentifier &id)
void	setEventId (const DtEventID &id)
void	setStateUpdateIndicator (DtEmissionPduType indicator)
void	setNumSystems (int num)
void	setNumBeams (int sysNum, int numBeams, int *retcode = NULL)
void	setSystemName (int sysNum, DtEmitterSysName name, int *retcode = NULL)
void	setSystemFunction (int sysNum, DtEmissionFunc func, int *retcode = NULL)
void	setEmitterId (int sysNum, int id, int *retcode = NULL)
void	setLocation (int sysNum, DtConstVector loc, int *retcode = NULL)
void	setBeamId (int sysNum, int beamNum, int id, int *retcode = NULL)
void	setBeamParameterIndex (int sysNum, int beamNum, int bplIndex, int *retcode = NULL)
void	setParameterData (int sysNum, int beamNum, const DtFundamentalParameterData &data, int *retcode = NULL)
void	setBeamFunction (int sysNum, int beamNum, int func, int *retcode = NULL)
void	setNumTargetsInBeam (int sysNum, int beamNum, int num, int *retcode = NULL)
void	setTrackingAllTargets (int sysNum, int beamNum, DtTrackingJamming trackJamAll, int *retcode = NULL)
void	setJammingModeSequence (int sysNum, int beamNum, unsigned long seq, int *retcode = NULL)
void	setTrackJamTarget (int sysNum, int beamNum, int targetNum, const DtTrackJamTarget &targ, int *retcode = NULL)
void	printData () const

Protected Fields

DtEntityIdentifier	cacheId
DtEventID	cacheEvent
DtVector	cacheLoc
DtFundamentalParameterData	cacheData
DtTrackJamTarget	cacheTarget

Protected Methods

 DtPduKind	internalGetPduKind () const
 DtNetSystemData*	emitterSystemAddr (int sysNum, int *retcode = NULL) const
 DtNetBeamData*	emitterBeamAddr (int sysNum, int beamNum, int *retcode = NULL) const
 DtNetEmitterTrackJam*	targetAddr (int sysNum, int beamNum, int targetNum, int *retcode = NULL) const
 int	computedSystemDataLength (int sysNum) const
 int	computedBeamDataLength (int sysNum, int beamNum) const
 void	printSystem (int i) const
 void	printBeam (int i, int j) const
 int	addressInPacket (void *addr) const
 int	computeStatus () const

Inherited from [DtStateMsg](#):

Public Methods

 [DtStateMsg](#)& **operator=**(const [DtStateMsg](#)& orig)

Inherited from [DtPdu](#):

Public Methods

 virtual void* **packet**() const

 virtual int **status**() const

 virtual void **setVersion**(int vers)

 virtual int **protocolVersion**() const

 virtual void **setExerciseld**(int id)

 virtual int **exerciseld**() const

 virtual [DtPduKind](#) **kind**() const

 virtual [DtProtocolFamily](#) **protocolFamily**() const

 virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)

 virtual DtTime **timeStamp**() const

 virtual [DtTimeStampType](#) **timeStampType**() const

- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)
- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- `DtElectromagneticEmissionPdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `DtElectromagneticEmissionPdu(const DtNetElectromagneticEmissionPdu *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `static DtPdu* create(const DtNetPacket *initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)`
- `static void addCallback(DtExerciseConn *conn, DtElectromagneticEmissionPduCb cb, void *usr)`

```

static void removeCallback(DtExerciseConn *conn, DtElectromagneticEmissionPduCb cb,
void *usr)

const DtEntityIdentifier& emittingEntityId() const

const DtEventID& eventId() const

DtEmissionPduType stateUpdateIndicator() const

int numSystems() const

int systemDataWords(int sysNum, int *retcode = NULL) const

int numBeams(int sysNum, int *retcode = NULL) const

DtEmitterSysName systemName(int sysNum, int *retcode = NULL) const

DtEmissionFunc systemFunction(int sysNum, int *retcode = NULL) const

int emitterId(int sysNum, int *retcode = NULL) const

DtConstVector location(int sysNum, int *retcode = NULL) const

int beamDataWords(int sysNum, int beamNum, int *retcode = NULL) const

int beamId(int sysNum, int beamNum, int *retcode = NULL) const

int beamParameterIndex(int sysNum, int beamNum, int *retcode = NULL) const

const DtFundamentalParameterData& parameterData(int sysNum, int beamNum, int *retcode =
NULL) const

int beamFunction(int sysNum, int beamNum, int *retcode = NULL) const

int numTargetsInBeam(int sysNum, int beamNum, int *retcode = NULL) const

DtTrackingJamming trackingAllTargets(int sysNum, int beamNum, int *retcode = NULL)
const

unsigned long jammingModeSequence(int sysNum, int beamNum, int *retcode = NULL) const

const DtTrackJamTarget& trackJamTarget(int sysNum, int beamNum, int targetNum, int
*retcode = NULL) const

void setEmittingEntityId(const DtEntityIdentifier &id)

void setEventId(const DtEventID &id)

void setStateUpdateIndicator(DtEmissionPduType indicator)

void setNumSystems(int num)

void setNumBeams(int sysNum, int numBeams, int *retcode = NULL)

void setSystemName(int sysNum, DtEmitterSysName name, int *retcode = NULL)

void setSystemFunction(int sysNum, DtEmissionFunc func, int *retcode = NULL)

void setEmitterId(int sysNum, int id, int *retcode = NULL)

void setLocation(int sysNum, DtConstVector loc, int *retcode = NULL)

```

- `void setBeamId(int sysNum, int beamNum, int id, int *retcode = NULL)`
- `void setBeamParameterIndex(int sysNum, int beamNum, int bpIndex, int *retcode = NULL)`
- `void setParameterData(int sysNum, int beamNum, const DtFundamentalParameterData &data, int *retcode = NULL)`
- `void setBeamFunction(int sysNum, int beamNum, int func, int *retcode = NULL)`
- `void setNumTargetsInBeam(int sysNum, int beamNum, int num, int *retcode = NULL)`
- `void setTrackingAllTargets(int sysNum, int beamNum, DtTrackingJamming trackJamAll, int *retcode = NULL)`
- `void setJammingModeSequence(int sysNum, int beamNum, unsigned long seq, int *retcode = NULL)`
- `void setTrackJamTarget(int sysNum, int beamNum, int targetNum, const DtTrackJamTarget &targ, int *retcode = NULL)`
- `void printData() const`
- `DtPduKind internalGetPduKind() const`
- `DtEntityIdentifier cacheId`
- `DtEventID cacheEvent`
- `DtVector cacheLoc`
- `DtFundamentalParameterData cacheData`
- `DtTrackJamTarget cacheTarget`
- `DtNetSystemData* emitterSystemAddr(int sysNum, int *retcode = NULL) const`
- `DtNetBeamData* emitterBeamAddr(int sysNum, int beamNum, int *retcode = NULL) const`
- `DtNetEmitterTrackJam* targetAddr(int sysNum, int beamNum, int targetNum, int *retcode = NULL) const`
- `int computedSystemDataLength(int sysNum) const`
- `int computedBeamDataLength(int sysNum, int beamNum) const`
- `void printSystem(int i) const`
- `void printBeam(int i, int j) const`
- `int addressInPacket(void *addr) const`
- `int computeStatus() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file aggstatepdu.h:

class [DtAggregateStatePdu](#): public [DtStateMsg](#)

Inheritance:

DtAggregateStatePdu - [DtStateMsg](#) - [DtPdu](#) - [DtBaseMsg](#)

Public Methods

	DtAggregateStatePdu (DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Constructors</i>
	DtAggregateStatePdu (const DtNetAggregateStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
	DtAggregateStatePdu (const <u>DtAggregateStatePdu</u> & orig)
 virtual	~DtAggregateStatePdu () <i>destructor</i>
 <u>DtAggregateStatePdu</u> &	operator= (const <u>DtAggregateStatePdu</u> & orig) <i>Assignment operator</i>
 virtual void	setAggregateID (const <u>DtEntityIdentifier</u> & entity) <i>Set/Get aggregate entity identifier</i>
 virtual const <u>DtEntityIdentifier</u> &	aggregateID () const
 virtual void	setForceID (<u>DtForceType</u> force) <i>Set/Get the force ID</i>
 virtual <u>DtForceType</u>	forceID () const
 virtual void	setAggregateState (<u>DtAggregateState</u> value) <i>Set/Get the aggregate state</i>
 virtual <u>DtAggregateState</u>	aggregateState () const
 virtual void	setAggregateType (const <u>DtEntityType</u> & type) <i>Set/Get the aggregate type</i>
 virtual const <u>DtEntityType</u> &	aggregateType () const
 virtual void	setFormation (<u>DtAggregateFormation</u> value) <i>Set/Get the formation</i>
 virtual <u>DtAggregateFormation</u>	formation () const
 virtual void	setMarkingCharacterSet (<u>DtCharacterSet</u> value) <i>Set/Get marking character set</i>
 virtual <u>DtCharacterSet</u>	markingCharacterSet () const

virtual void	setMarkingText (DtU8* copyFrom) <i>Set/Get marking text (copies 31 bytes)</i>
virtual void	getMarkingText (DtU8* copyTo) const
virtual void	setDimensions (DtConstVector dims) <i>Set/Get the dimensions</i>
virtual DtConstVector	dimensions () const
virtual void	setOrientation (const DtTaitBryan & orientation) <i>Set/Get orientation</i>
virtual const DtTaitBryan &	orientation () const
virtual void	setLocation (DtConstVector coords) <i>Set/Get the location of center of mass</i>
virtual DtConstVector	location () const
virtual void	setVelocity (DtConstVector coords) <i>Set/Get the velocity</i>
virtual DtConstVector	velocity () const
virtual DtBoolean	setNumberOfDIS_Subaggregates (DtU16 newNumberOfAggregates) <i>Set/Get the number of DIS sub-aggregates</i>
virtual DtU16	numberOfDIS_Subaggregates () const
virtual DtBoolean	setNumberOfDIS_Entities (DtU16 newNumberOfEntities) <i>Set/Get the number of DIS entities</i>
virtual DtU16	numberOfDIS_Entities () const
virtual DtBoolean	setNumberOfSilentSubaggregates (DtU16 newNumberOfAggregates) <i>Set/Get the number of silent aggregates</i>
virtual DtU16	numberOfSilentSubaggregates () const
virtual DtBoolean	setNumberOfSilentEntities (DtU16 newNumberOfEntities) <i>Set/Get the number of silent entities</i>
virtual DtU16	numberOfSilentEntities () const
virtual DtBoolean	setSubaggregateID (int index, const DtEntityIdentifier & aggID) <i>Set/Get a DIS sub-aggregate ID</i>
virtual DtBoolean	getSubaggregateID (int index, DtEntityIdentifier & aggID) const
virtual DtBoolean	setEntityID (int index, const DtEntityIdentifier & entID) <i>Set/Get a DIS entity ID</i>
virtual DtBoolean	getEntityID (int index, DtEntityIdentifier & entID) const
virtual DtBoolean	setSilentAggregateList (int index, const DtSilentAggregateList & aggList) <i>Set/Get a silent aggregate list</i>
virtual DtBoolean	getSilentAggregateList (int index, DtSilentAggregateList & aggList) const
virtual DtBoolean	setSilentEntityList (int index, const DtSilentEntityList & entList) <i>Set/Get a silent entity list</i>
virtual DtBoolean	getSilentEntityList (int index, DtSilentEntityList & entList) const
virtual DtBoolean	setNumberOfDatumRecords (DtU32 newNumberOfRecords) <i>Set/Get the number of datum records</i>
virtual DtU32	numberOfDatumRecords () const
virtual DtBoolean	setVariableDatumRecord (int index, const DtVariableDatumRecord & rec) <i>Set/Get a variable datum record</i>

virtual DtBoolean	getVariableDatumRecord (int index, DtVariableDatumRecord & rec) const
virtual const DtVariableDatumRecord &	variableDatumRecordConstRef (int index) const <i>Get a constant reference to a variable datum record</i>
virtual void	printData () const <i>Print the data contained by this PDU</i>
virtual DtBoolean	setDatumBitLength (int index, DtU32 newRecordBitLength) <i>Set/Get the datum length (in bits)</i>
virtual DtBoolean	getDatumBitLength (int index, DtU32& length) const
virtual DtBoolean	setDatumID (int index, DtDatumParam value) <i>Set/Get the variable datum ID</i>
virtual DtBoolean	getDatumID (int index, DtDatumParam & value) const
virtual DtBoolean	setDatumValFloat64 (int index, DtFloat64 value) <i>Set/Get datum value as a DtFloat64</i>
virtual DtBoolean	getDatumValFloat64 (int index, DtFloat64& value) const
virtual DtBoolean	setDatumValFloat32 (int index, DtFloat32 value) <i>Set/Get datum value as a DtFloat32</i>
virtual DtBoolean	getDatumValFloat32 (int index, DtFloat32& value) const
virtual DtBoolean	setDatumValInt32 (int index, DtInt32 value) <i>Set/Get datum value as a DtInt32</i>
virtual DtBoolean	getDatumValInt32 (int index, DtInt32& value) const
virtual DtBoolean	setDatumValU32 (int index, DtU32 value) <i>Set/Get datum value as a DtU32</i>
virtual DtBoolean	getDatumValU32 (int index, DtU32& value) const
virtual DtBoolean	setDatumValInt16 (int index, DtInt16 value) <i>Set/Get datum value as a DtInt16</i>
virtual DtBoolean	getDatumValInt16 (int index, DtInt16& value) const
virtual DtBoolean	setDatumValU16 (int index, DtU16 value) <i>Set/Get datum value as a DtU16</i>
virtual DtBoolean	getDatumValU16 (int index, DtU16& value) const
virtual DtBoolean	setDatumValU8 (int index, DtU8 value) <i>Set/Get datum value as a DtU8</i>
virtual DtBoolean	getDatumValU8 (int index, DtU8& value) const
virtual DtBoolean	setDatumValByteArray (int index, const DtU8* setFrom) <i>Set/Get datum value as a byte array</i>
virtual DtBoolean	getDatumValByteArray (int index, DtU8* copyTo) const
virtual DtBoolean	getDatumValPtr (int index, DtNetU8*& datumPtr) <i>Get a pointer to datum value</i>
static DtPdu *	create (const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create a DtAggregateStatePdu</i>
static void	addCallback (DtExerciseConn * conn, DtAggregateStatePduCb cb, void* usr) <i>Add (register)/Remove (deregister) function to be called when pdu is received</i>
static void	removeCallback (DtExerciseConn * conn, DtAggregateStatePduCb cb, void* usr)

Public

WARNING: These methods are obsolete but have been retained for backward compatibility

Protected Fields

 static int	theStaticSize <i>Size of PDU with zero data bytes</i>
 DtEntityIdentifier	myEntityID <i>Temporary objects for reference returns</i>
 DtEntityType	myEntityType
 DtVector	myVector
 DtTaitBryan	myAngles
 DtVariableDatumRecord	myRefRec
 DtVariableDatumRecord	myDumRec

Protected Methods

 virtual DtPduKind	internalGetPduKind () const <i>Returns the internal PDU kind</i>
 virtual DtNetAggregateStatePdu*	netPduStruct () const <i>Casts the netPdu element to our network struct Non-virtual since you can't override based on return type</i>
 DtAggregateStatePdu *	self () const <i>Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type</i>
 virtual char*	var () const <i>Returns a pointer to the first byte of variable length data</i>
 virtual int	subaggregateID_Offset (int index) const <i>Returns an offset from the beginning of variable length data to a particular subaggregate ID</i>
 virtual int	entityID_Offset (int index) const <i>Returns an offset from the beginning of variable length data to a particular entity ID</i>
 virtual int	paddingOffset () const <i>Returns an offset from the beginning of variable length data to the padding following the entity ID records</i>
 virtual int	paddingBytes () const <i>Returns the number of padding bytes which must follow the entity ID records</i>
 virtual int	silentAggregateOffset (int index) const <i>Returns an offset from the beginning of variable length data to a particular silent aggregate list</i>
 virtual int	silentEntityOffset (int index) const <i>Returns an offset from the beginning of variable length data to a particular silent entity system list</i>
 virtual int	numberOfDatumRecordsOffset () const <i>Returns an offset from the beginning of variable length data to the number of variable datum records field</i>

virtual int

datumRecordOffset (int index) const

Returns an offset from the beginning of variable length data to a particular variable datum record

Inherited from [DtStateMsg](#):

Inherited from [DtPdu](#):

Public Methods

- virtual void* **packet**() const
- virtual int **status**() const
- virtual void **setVersion**(int vers)
- virtual int **protocolVersion**() const
- virtual void **setExerciseld**(int id)
- virtual int **exerciseld**() const
- virtual [DtPduKind](#) **kind**() const
- virtual [DtProtocolFamily](#) **protocolFamily**() const
- virtual void **setTimeStamp**(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)
- virtual DtTime **timeStamp**() const
- virtual [DtTimeStampType](#) **timeStampType**() const
- virtual DtTime **guessTimeValid**(DtTime referenceTime) const
- virtual void **print**() const
- virtual void **printHeader**() const
- inline int **exerciseld**() const
- inline [DtPduKind](#) **kind**() const
- inline DtNetPduHeader* **netHeader**() const
- inline void* **packet**() const
- inline [DtProtocolFamily](#) **protocolFamily**() const
- inline int **protocolVersion**() const
- inline void **setExerciseld**(int id)
- inline void **setStatus**(int okStatus)
- inline void **setTimeStamp**(DtTime time, [DtTimeStampType](#) relOrAbs)

- inline void **setVersion**(int vers)
- inline int **status**() const
- inline DtTime **timeStamp**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- int **myOKStatus**

Protected Methods

- virtual void **setStatus**(int okStatus)
 - virtual DtNetPduHeader* **netHeader**() const
 - virtual void **initPdu**(int size, DtBufferPtr buffer)
 - virtual void **initPdu**(const void* initial, DtBufferPtr buffer)
 - virtual [DtProtocolFamily](#) **kindToFamily**([DtPduKind](#) kind) const
 - virtual void **sizeChanged**(int size)
 - virtual void **clearData**()
 - virtual void **initHeader**([DtPduKind](#) kind, int size)
-

Inherited from [DtBaseMsg](#):

Public Methods

- virtual const char* **netRep**() const
- virtual int **length**() const
- inline const char* **netRep**() const

Protected Fields

- void* **myNetMessage**
- void* **myInternalBuff**
- int **myIntBuffSize**
- int **myMsgSize**

Protected Methods

- void **initMsg**(int size, DtBufferPtr bufferPtr)
 - void **initMsg**(const void *initial, int size, DtBufferPtr bufferPtr)
 - virtual void **changeSize**(int size)
 - virtual void **insertBytes**(int offset, int numBytes)
 - virtual void **insertBytes**(void *address, int numBytes)
 - virtual void **deleteBytes**(int offset, int numBytes)
 - virtual void **deleteBytes**(void *address, int numBytes)
 - virtual void **clearAll**()
 - virtual void* **newInternalBuffer**(int size)
-

Documentation

- DtAggregateStatePdu(DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
Constructors
- DtAggregateStatePdu(const DtNetAggregateStatePdu* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- DtAggregateStatePdu(const [DtAggregateStatePdu](#)& orig)
- virtual ~DtAggregateStatePdu()
destructor
- [DtAggregateStatePdu](#)& operator=(const [DtAggregateStatePdu](#)& orig)
Assignment operator
- virtual void setAggregateID(const [DtEntityIdentifier](#)& entity)
Set/Get aggregate entity identifier
- virtual const [DtEntityIdentifier](#)& aggregateID() const
- virtual void setForceID([DtForceType](#) force)
Set/Get the force ID
- virtual [DtForceType](#) forceID() const
- virtual void setAggregateState([DtAggregateState](#) value)
Set/Get the aggregate state
- virtual [DtAggregateState](#) aggregateState() const
- virtual void setAggregateType(const [DtEntityType](#)& type)
Set/Get the aggregate type

- virtual const [DtEntityType](#)& aggregateType() const
- virtual void setFormation([DtAggregateFormation](#) value)
Set/Get the formation
- virtual [DtAggregateFormation](#) formation() const
- virtual void setMarkingCharacterSet([DtCharacterSet](#) value)
Set/Get marking character set
- virtual [DtCharacterSet](#) markingCharacterSet() const
- virtual void setMarkingText(DtU8* copyFrom)
Set/Get marking text (copies 31 bytes)
- virtual void getMarkingText(DtU8* copyTo) const
- virtual void setDimensions(DtConstVector dims)
Set/Get the dimensions
- virtual DtConstVector dimensions() const
- virtual void setOrientation(const [DtTaitBryan](#)& orientation)
Set/Get orientation
- virtual const [DtTaitBryan](#)& orientation() const
- virtual void setLocation(DtConstVector coords)
Set/Get the location of center of mass
- virtual DtConstVector location() const
- virtual void setVelocity(DtConstVector coords)
Set/Get the velocity
- virtual DtConstVector velocity() const
- virtual [DtBoolean](#) setNumberOfDIS_Subaggregates(DtU16 newNumberOfAggregates)
Set/Get the number of DIS sub-aggregates. Set returns DtFalse if the requested number is too large.
- virtual DtU16 numberOfDIS_Subaggregates() const
- virtual [DtBoolean](#) setNumberOfDIS_Entities(DtU16 newNumberOfEntities)
Set/Get the number of DIS entities. Set returns DtFalse if the requested number is too large.
- virtual DtU16 numberOfDIS_Entities() const
- virtual [DtBoolean](#) setNumberOfSilentSubaggregates(DtU16 newNumberOfAggregates)
Set/Get the number of silent aggregates. Set returns DtFalse if the requested number is too large.
- virtual DtU16 numberOfSilentSubaggregates() const
- virtual [DtBoolean](#) setNumberOfSilentEntities(DtU16 newNumberOfEntities)
Set/Get the number of silent entities. Set returns DtFalse if the requested number is too large.
- virtual DtU16 numberOfSilentEntities() const

- `virtual DtBoolean setSubaggregateID(int index, const DtEntityIdentifier& aggID)`
Set/Get a DIS sub-aggregate ID. Returns DtFalse if the index is invalid.
- `virtual DtBoolean getSubaggregateID(int index, DtEntityIdentifier& aggID) const`
- `virtual DtBoolean setEntityID(int index, const DtEntityIdentifier& entID)`
Set/Get a DIS entity ID. Returns DtFalse if the index is invalid.
- `virtual DtBoolean getEntityID(int index, DtEntityIdentifier& entID) const`
- `virtual DtBoolean setSilentAggregateList(int index, const DtSilentAggregateList& aggList)`
Set/Get a silent aggregate list. Returns DtFalse if the index is invalid.
- `virtual DtBoolean getSilentAggregateList(int index, DtSilentAggregateList& aggList) const`
- `virtual DtBoolean setSilentEntityList(int index, const DtSilentEntityList& entList)`
Set/Get a silent entity list. Returns DtFalse if the index is invalid.
- `virtual DtBoolean getSilentEntityList(int index, DtSilentEntityList& entList) const`
- `virtual DtBoolean setNumberOfDatumRecords(DtU32 newNumberOfRecords)`
Set/Get the number of datum records. Set returns DtFalse if the requested number is too large.
- `virtual DtU32 numberOfDatumRecords() const`
- `virtual DtBoolean setVariableDatumRecord(int index, const DtVariableDatumRecord& rec)`
Set/Get a variable datum record. Returns DtFalse if the index is invalid.
- `virtual DtBoolean getVariableDatumRecord(int index, DtVariableDatumRecord& rec) const`
- `virtual const DtVariableDatumRecord& variableDatumRecordConstRef( int index) const`
Get a constant reference to a variable datum record
- `virtual void printData() const`
Print the data contained by this PDU

WARNING: These methods are obsolete but have been retained for backward compatibility. These methods may be removed in a future release. Instead, the methods `setVariableDatumRecord`, `getVariableDatumRecord`, and `variableDatumRecordConstRef` should be used.

- `virtual DtBoolean setDatumBitLength(int index, DtU32 newRecordBitLength)`
Set/Get the datum length (in bits). Returns DtFalse if the index is invalid. Set may destroy data in the record.
- `virtual DtBoolean getDatumBitLength(int index, DtU32& length) const`
- `virtual DtBoolean setDatumID(int index, DtDatumParam value)`
Set/Get the variable datum ID. Returns DtFalse if the index is invalid.
- `virtual DtBoolean getDatumID(int index, DtDatumParam& value) const`
- `virtual DtBoolean setDatumValFloat64(int index, DtFloat64 value)`

Set/Get datum value as a DtFloat64. Returns DtFalse if the index is invalid or if the field length is not 64 bits.

virtual [DtBoolean](#) getDatumValFloat64(int index, DtFloat64& value) const

virtual [DtBoolean](#) setDatumValFloat32(int index, DtFloat32 value)

Set/Get datum value as a DtFloat32. Returns DtFalse if the index is invalid or if the field length is not 32 bits.

virtual [DtBoolean](#) getDatumValFloat32(int index, DtFloat32& value) const

virtual [DtBoolean](#) setDatumValInt32(int index, DtInt32 value)

Set/Get datum value as a DtInt32. Returns DtFalse if the index is invalid or if the field length is not 32 bits.

virtual [DtBoolean](#) getDatumValInt32(int index, DtInt32& value) const

virtual [DtBoolean](#) setDatumValU32(int index, DtU32 value)

Set/Get datum value as a DtU32. Returns DtFalse if the index is invalid or if the field length is not 32 bits.

virtual [DtBoolean](#) getDatumValU32(int index, DtU32& value) const

virtual [DtBoolean](#) setDatumValInt16(int index, DtInt16 value)

Set/Get datum value as a DtInt16. Returns DtFalse if the index is invalid or if the field length is not 16 bits.

virtual [DtBoolean](#) getDatumValInt16(int index, DtInt16& value) const

virtual [DtBoolean](#) setDatumValU16(int index, DtU16 value)

Set/Get datum value as a DtU16. Returns DtFalse if the index is invalid or if the field length is not 16 bits.

virtual [DtBoolean](#) getDatumValU16(int index, DtU16& value) const

virtual [DtBoolean](#) setDatumValU8(int index, DtU8 value)

Set/Get datum value as a DtU8. Returns DtFalse if the index is invalid or if the field length is not 8 bits.

virtual [DtBoolean](#) getDatumValU8(int index, DtU8& value) const

virtual [DtBoolean](#) setDatumValByteArray(int index, const DtU8* setFrom)

Set/Get datum value as a byte array. Returns DtFalse if the index is invalid. Data is copied to and from the network PDU structure according to the datum length of the record.

virtual [DtBoolean](#) getDatumValByteArray(int index, DtU8* copyTo) const

virtual [DtBoolean](#) getDatumValPtr(int index, DtNetU8*& datumPtr)

Get a pointer to datum value. Returns DtFalse if the index is invalid.

static [DtPdu](#)* create(const DtNetPacket* initial, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create a DtAggregateStatePdu. Caller is responsible for deletion.

static void addCallback([DtExerciseConn](#)* conn, DtAggregateStatePduCb cb, void* usr)

Add (register)/Remove (deregister) function to be called when pdu is received

static void removeCallback([DtExerciseConn](#)* conn, DtAggregateStatePduCb cb, void* usr)

virtual [DtPduKind](#) internalGetPduKind() const

Returns the internal PDU kind

- 
virtual DtNetAggregateStatePdu* netPduStruct() const
 Casts the netPdu element to our network struct Non-virtual since you can't override based on return type
- 
[DtAggregateStatePdu](#)* self() const
 Casts the 'this' pointer for changing elements in a const function Non virtual since you can't override based on return type
- 
virtual char* var() const
 Returns a pointer to the first byte of variable length data
- 
virtual int subaggregateID_Offset(int index) const
 Returns an offset from the beginning of variable length data to a particular subaggregate ID
- 
virtual int entityID_Offset(int index) const
 Returns an offset from the beginning of variable length data to a particular entity ID
- 
virtual int paddingOffset() const
 Returns an offset from the beginning of variable length data to the padding following the entity ID records
- 
virtual int paddingBytes() const
 Returns the number of padding bytes which must follow the entity ID records
- 
virtual int silentAggregateOffset(int index) const
 Returns an offset from the beginning of variable length data to a particular silent aggregate list
- 
virtual int silentEntityOffset(int index) const
 Returns an offset from the beginning of variable length data to a particular silent entity system list
- 
virtual int numberOfDatumRecordsOffset() const
 Returns an offset from the beginning of variable length data to the number of variable datum records field
- 
virtual int datumRecordOffset(int index) const
 Returns an offset from the beginning of variable length data to a particular variable datum record
- 
static int theStaticSize
 Size of PDU with zero data bytes
- 
[DtEntityIdentifier](#) myEntityID
 Temporary objects for reference returns
- 
[DtEntityType](#) myEntityType
- 
[DtVector](#) myVector
- 
[DtTaitBryan](#) myAngles
- 
[DtVariableDatumRecord](#) myRefRec
- 
[DtVariableDatumRecord](#) myDumRec

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtAggregateState

Aggregate State PDU enums

-  DtOtherAggregateState
 -  DtAggregated
 -  DtDisaggregated
 -  DtFullyDisaggregated
 -  DtPseudoDisaggregated
 -  DtPartiallyDisaggregated
-

Documentation

Aggregate State PDU enums

-  DtOtherAggregateState
-  DtAggregated
-  DtDisaggregated
-  DtFullyDisaggregated
-  DtPseudoDisaggregated
-  DtPartiallyDisaggregated

In file disenums.h:

enum DtAggregateFormation

-
-  DtOtherAggregateFormation
 -  DtAssemblyFormation
 -  DtVeeFormation
 -  DtWedgeFormation
 -  DtLineFormation
 -  DtColumnFormation
-

Documentation

-  DtOtherAggregateFormation
-  DtAssemblyFormation
-  DtVeeFormation
-  DtWedgeFormation
-  DtLineFormation
-  DtColumnFormation

In file `aggagglst.h`:

class DtSilentAggregateList

class DtSilentAggregateList: DtSilentAggregateList is a class which contains the information needed to describe a list of silent aggregates for the aggregate state PDU

Public Methods

	DtSilentAggregateList () <i>Constructors</i>
	DtSilentAggregateList (const DtSilentAggregateList & orig)
 virtual	~DtSilentAggregateList () <i>Destructor</i>
 DtSilentAggregateList &	operator= (const DtSilentAggregateList & orig) <i>Assignment operator</i>
 DtSilentAggregateList &	operator= (const DtNetSilentAggregateList& orig)
 int	operator== (const DtSilentAggregateList & otherguy) const <i>Comparison operators</i>
 int	operator!= (const DtSilentAggregateList & otherguy) const
 virtual void	setNumberOfAggregates (const DtU16 value) <i>Set/Get the number of aggregates</i>
 virtual DtU16	numberOfAggregates () const
 virtual void	setAggregateType (const DtEntityType & type) <i>Set/Get the aggregate type</i>
 virtual const DtEntityType &	aggregateType () const
 virtual void	print () const <i>Print the record</i>
 virtual void	unloadNetRecord (const DtNetSilentAggregateList& record) <i>Load/unload data from a packet record</i>
 virtual void	loadNetRecord (DtNetSilentAggregateList& record) const
 static int	byteLengthOfNetworkRecord () <i>Get the length of the network representation of a record</i>

Protected Fields

 DtU16	myNumberOfAggregates <i>Data members:</i>
 DtEntityType	myEntityType

Documentation

class DtSilentAggregateList: DtSilentAggregateList is a class which contains the information needed to describe a list of silent aggregates for the aggregate state PDU

- `DtSilentAggregateList()`
Constructors
- `DtSilentAggregateList(const DtSilentAggregateList& orig)`
- `virtual ~DtSilentAggregateList()`
Destructor
- `DtSilentAggregateList& operator=(const DtSilentAggregateList& orig)`
Assignment operator
- `DtSilentAggregateList& operator=(const DtNetSilentAggregateList& orig)`
- `int operator==(const DtSilentAggregateList& otherguy) const`
Comparison operators
- `int operator!=(const DtSilentAggregateList& otherguy) const`
- `virtual void setNumberOfAggregates(const DtU16 value)`
Set/Get the number of aggregates
- `virtual DtU16 numberOfAggregates() const`
- `virtual void setAggregateType(const DtEntityType& type)`
Set/Get the aggregate type
- `virtual const DtEntityType& aggregateType() const`
- `virtual void print() const`
Print the record
- `virtual void unloadNetRecord(const DtNetSilentAggregateList& record)`
Load/unload data from a packet record
- `virtual void loadNetRecord(DtNetSilentAggregateList& record) const`
- `static int byteLengthOfNetworkRecord()`
Get the length of the network representation of a record
- `DtU16 myNumberOfAggregates`
Data members:
- `DtEntityType myEntityType`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file agentlist.h:

class DtSilentEntityList

class DtSilentEntityList: DtSilentEntityList is a class which contains the information needed to describe a list of silent entities for the aggregate state PDU

Public Methods

	DtSilentEntityList () <i>Constructors</i>
	DtSilentEntityList (const DtSilentEntityList & orig)
virtual	~DtSilentEntityList () <i>Destructor</i>
DtSilentEntityList &	operator= (const DtSilentEntityList & orig) <i>Assignment operator</i>
DtSilentEntityList &	operator= (const DtNetSilentEntityList& orig)
int	operator== (const DtSilentEntityList & otherguy) const <i>Comparison operators</i>
int	operator!= (const DtSilentEntityList & otherguy) const
virtual void	setNumberOfEntities (const DtU16 value) <i>Set/Get the number of aggregates</i>
virtual DtU16	numberOfEntities () const
virtual void	setEntityType (const DtEntityType & type) <i>Set/Get the entity type</i>
virtual const DtEntityType &	entityType () const
virtual DtBoolean	setNumberOfAppearanceRecords (const DtU16 value) <i>Set/Get the number of appearance records</i>
virtual DtU16	numberOfAppearanceRecords () const
virtual DtBoolean	setAppearance (int index, DtU32 app) <i>Set/Get the appearance of an entity (as a 32 bit integer)</i>
virtual DtBoolean	getAppearance (int index, DtU32& app) const
virtual DtBoolean	setAppearanceAttribute (int index, const DtAppearAttribute attr, const DtAppearAttrVal val) <i>Set/Get appearance by attribute</i>
virtual DtBoolean	getAppearanceAttribute (int index, const DtAppearAttribute attr, DtAppearAttrVal & val) const
virtual void	print () const <i>Print the record</i>
virtual void	unloadNetRecord (const DtNetSilentEntityList& record) <i>Load/unload data from a packet record</i>
virtual void	loadNetRecord (DtNetSilentEntityList& record) const

- virtual int **networkByteLength** () const
Get the network byte length of this record
- static int **byteLengthOfNetworkRecord** (DtNetSilentEntityList* recPtr)
Get the length of the network representation of a record

Protected Fields

- DtU16 **myNumberOfEntities**
Data members:
 - [DtEntityType](#) **myEntityType**
 - DtU16 **myNumberOfAppearanceRecords**
 - DtU32* **myEntityAppearanceList**
-

Documentation

class DtSilentEntityList: DtSilentEntityList is a class which contains the information needed to describe a list of silent entities for the aggregate state PDU

- DtSilentEntityList()**
Constructors
- DtSilentEntityList(const [DtSilentEntityList](#)& orig)**
- virtual ~DtSilentEntityList()**
Destructor
- [DtSilentEntityList](#)& operator=(const [DtSilentEntityList](#)& orig)**
Assignment operator
- [DtSilentEntityList](#)& operator=(const DtNetSilentEntityList& orig)**
- int operator==(const [DtSilentEntityList](#)& otherguy) const**
Comparison operators
- int operator!=(const [DtSilentEntityList](#)& otherguy) const**
- virtual void setNumberOfEntities(const DtU16 value)**
Set/Get the number of aggregates
- virtual DtU16 numberOfEntities() const**
- virtual void setEntityType(const [DtEntityType](#)& type)**
Set/Get the entity type
- virtual const [DtEntityType](#)& entityType() const**
- virtual [DtBoolean](#) setNumberOfAppearanceRecords(const DtU16 value)**
Set/Get the number of appearance records. Set returns DtFalse if the value is too large, old values are not retained when the number changes.

- `virtual DtU16 numberOfAppearanceRecords() const`
- `virtual DtBoolean setAppearance(int index, DtU32 app)`
Set/Get the appearance of an entity (as a 32 bit integer). Returns DtFalse if the index is out of range. Set will erase all previous values.
- `virtual DtBoolean getAppearance(int index, DtU32& app) const`
- `virtual DtBoolean setAppearanceAttribute(int index, const DtAppearAttribute attr, const DtAppearAttrVal val)`
Set/Get appearance by attribute. Returns DtFalse if the index is out of range.
- `virtual DtBoolean getAppearanceAttribute(int index, const DtAppearAttribute attr, DtAppearAttrVal& val) const`
- `virtual void print() const`
Print the record
- `virtual void unloadNetRecord(const DtNetSilentEntityList& record)`
Load/unload data from a packet record
- `virtual void loadNetRecord(DtNetSilentEntityList& record) const`
- `virtual int networkByteLength() const`
Get the network byte length of this record
- `static int byteLengthOfNetworkRecord(DtNetSilentEntityList* recPtr)`
Get the length of the network representation of a record
- `DtU16 myNumberOfEntities`
Data members:
- `DtEntityType myEntityType`
- `DtU16 myNumberOfAppearanceRecords`
- `DtU32* myEntityAppearanceList`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vardatumrec.h:

class DtVariableDatumRecord

class DtVariableDatumRecord: Instances of DtVariableDatumRecord are used to hold a single variable datum record and to provide access to its network representation

Public Methods

	DtVariableDatumRecord () <i>default constructor</i>
virtual	~DtVariableDatumRecord () <i>destructor</i>
	DtVariableDatumRecord (const DtVariableDatumRecord & orig) <i>copy constructor</i>
DtVariableDatumRecord &	operator= (const DtVariableDatumRecord & orig) <i>assignment operator</i>
int	operator== (const DtVariableDatumRecord & otherguy) const <i>comparison operators</i>
int	operator!= (const DtVariableDatumRecord & otherguy) const
virtual void	setExternalBuffer (char* bufferPtr) <i>Set the working buffer to an externally supplied buffer</i>
virtual void	setDatumBitLength (DtU32 newDatumBitLength) <i>Set/Get the datum length in bits</i>
virtual DtU32	datumBitLength () const
virtual int	datumByteLength () const <i>Get the datum length in bytes (padded to an 8 byte boundary)</i>
virtual void	setDatumID (DtDatumParam value) <i>Set/Get the variable datum ID</i>
virtual DtDatumParam	datumID () const
virtual void	setDatumValFloat64 (DtFloat64 value) <i>Set/Get datum value as a DtFloat64 Set will set the datum bit length to 64</i>
virtual DtFloat64	datumValFloat64 () const
virtual void	setDatumValFloat32 (DtFloat32 value) <i>Set/Get datum value as a DtFloat32 Set will set the datum bit length to 32</i>
virtual DtFloat32	datumValFloat32 () const
virtual void	setDatumValInt32 (DtInt32 value) <i>Set/Get datum value as a DtInt32 Set will set the datum bit length to 32</i>
virtual DtInt32	datumValInt32 () const
virtual void	setDatumValU32 (DtU32 value) <i>Set/Get datum value as a DtU32 Set will set the datum bit length to 32</i>
virtual DtU32	datumValU32 () const
virtual void	setDatumValInt16 (DtInt16 value) <i>Set/Get datum value as a DtInt16 Set will set the datum bit length to 16</i>

virtual DtInt16	datumValInt16 () const
virtual void	setDatumValU16 (DtU16 value) <i>Set/Get datum value as a DtU16 Set will set the datum bit length to 16</i>
virtual DtU16	datumValU16 () const
virtual void	setDatumValU8 (DtU8 value) <i>Set/Get datum value as a DtU8 Set will set the datum bit length to 8</i>
virtual DtU8	datumValU8 () const
virtual void	setDatumValByteArray (const DtU8* setFrom) <i>Set/Get datum value as a byte array</i>
virtual void	getDatumValByteArray (DtU8* copyTo) const
virtual char*	datumValPtr () <i>Get a pointer to the datum value inside of the record</i>
virtual const char*	constDatumValPtr () const <i>Get a const pointer to the datum value inside of the record</i>
virtual int	networkByteLength () const <i>Get the network byte length of the entire datum record</i>
virtual const char*	networkDataBuffer () const <i>Get a pointer to the buffer which stores the network representation of the datum record</i>
virtual void	print () const <i>Print the contents of the record</i>

Protected Fields

- char* **myRecordBuffer**
- char* **myExternalBuffer**

Protected Methods

virtual DtNetVarDatum*	netStruct () <i>Cast the buffer to a DtNetVarDatum struct</i>
virtual const DtNetVarDatum*	constNetStruct () const <i>Cast the buffer to a const DtNetVarDatum struct</i>
virtual void	deleteIfNoExternal () <i>Delete the internal buffer if there is no external buffer</i>
virtual int	datumByteSize (int datumBitLength) const <i>Compute the total datum size in bytes from the datum bit length</i>
virtual int	recordByteSize (int datumBitLength) const <i>Compute the total record size in bytes from the datum bit length</i>

Documentation

class DtVariableDatumRecord: Instances of DtVariableDatumRecord are used to hold a single variable datum record and to provide access to its network representation

- DtVariableDatumRecord()**
default constructor

- `virtual ~DtVariableDatumRecord()`
destructor
- `DtVariableDatumRecord(const DtVariableDatumRecord& orig)`
copy constructor
- `DtVariableDatumRecord& operator=(const DtVariableDatumRecord& orig)`
assignment operator
- `int operator==(const DtVariableDatumRecord& otherguy) const`
comparison operators
- `int operator!=(const DtVariableDatumRecord& otherguy) const`
- `virtual void setExternalBuffer(char* bufferPtr)`
Set the working buffer to an externally supplied buffer
- `virtual void setDatumBitLength(DtU32 newDatumBitLength)`
Set/Get the datum length in bits. This method will re-allocate memory if the buffer is an internal buffer.
- `virtual DtU32 datumBitLength() const`
- `virtual int datumByteLength() const`
Get the datum length in bytes (padded to an 8 byte boundary)
- `virtual void setDatumID(DtDatumParam value)`
Set/Get the variable datum ID
- `virtual DtDatumParam datumID() const`
- `virtual void setDatumValFloat64(DtFloat64 value)`
Set/Get datum value as a DtFloat64 Set will set the datum bit length to 64
- `virtual DtFloat64 datumValFloat64() const`
- `virtual void setDatumValFloat32(DtFloat32 value)`
Set/Get datum value as a DtFloat32 Set will set the datum bit length to 32
- `virtual DtFloat32 datumValFloat32() const`
- `virtual void setDatumValInt32(DtInt32 value)`
Set/Get datum value as a DtInt32 Set will set the datum bit length to 32
- `virtual DtInt32 datumValInt32() const`
- `virtual void setDatumValU32(DtU32 value)`
Set/Get datum value as a DtU32 Set will set the datum bit length to 32
- `virtual DtU32 datumValU32() const`
- `virtual void setDatumValInt16(DtInt16 value)`
Set/Get datum value as a DtInt16 Set will set the datum bit length to 16
- `virtual DtInt16 datumValInt16() const`
- `virtual void setDatumValU16(DtU16 value)`

Set/Get datum value as a DtU16 Set will set the datum bit length to 16

• `virtual DtU16 datumValU16() const`

• `virtual void setDatumValU8(DtU8 value)`

Set/Get datum value as a DtU8 Set will set the datum bit length to 8

• `virtual DtU8 datumValU8() const`

• `virtual void setDatumValByteArray(const DtU8* setFrom)`

Set/Get datum value as a byte array. Data is copied to and from the network data buffer according to the specified length of the record.

• `virtual void getDatumValByteArray(DtU8* copyTo) const`

• `virtual char* datumValPtr()`

Get a pointer to the datum value inside of the record

• `virtual const char* constDatumValPtr() const`

Get a const pointer to the datum value inside of the record

• `virtual int networkByteLength() const`

Get the network byte length of the entire datum record

• `virtual const char* networkDataBuffer() const`

Get a pointer to the buffer which stores the network representation of the datum record

• `virtual void print() const`

Print the contents of the record

• `virtual DtNetVarDatum* netStruct()`

Cast the buffer to a DtNetVarDatum struct

• `virtual const DtNetVarDatum* constNetStruct() const`

Cast the buffer to a const DtNetVarDatum struct

• `virtual void deleteIfNoExternal()`

Delete the internal buffer if there is no external buffer

• `virtual int datumByteSize(int datumBitLength) const`

Compute the total datum size in bytes from the datum bit length

• `virtual int recordByteSize(int datumBitLength) const`

Compute the total record size in bytes from the datum bit length

• `char* myRecordBuffer`

• `char* myExternalBuffer`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vllist.h:

class DtList

A DtList is a list of elements of arbitrary type

Inheritance:

DtList

Public Methods

	DtList () <i>Constructor</i>
virtual	~DtList () <i>Destructor - does NOT free memory of data associated with the items</i>
	DtList (const DtList & orig) <i>Copy constructor</i>
DtList &	operator= (const DtList & orig) <i>Assignment operator</i>
DtListItem *	first () const <i>Return first and last item on the list</i>
DtListItem *	last () const
unsigned	count () const <i>Returns the number of elements in the list</i>
DtListItem *	add (void* data) <i>Add element to the end of the list</i>
DtListItem *	addToStart (void* data) <i>Add element to the beginning of the list</i>
DtListItem *	addToEnd (void* data) <i>Add element to the end of the list</i>
DtListItem *	addBefore (void* data, DtListItem * before) <i>Add element to the list just before the indicated item</i>
DtListItem *	addAfter (void* data, DtListItem * after) <i>Add element to the list just after the indicated item</i>
void*	remove (DtListItem * item) <i>Remove an element from the list</i>
void	removeAll () <i>Calls remove for all elements in the list</i>
DtListItem *	itemLookup (void* data) const <i>Returns the DtListItem envelope object containing the indicated data element</i>
inline DtListItem *	add (void* data)

- inline [DtListItem](#)* **addBefore** (void* data, [DtListItem](#)* before)
- inline [DtListItem](#)* **addToEnd** (void* data)
- inline [DtListItem](#)* **addToStart** (void* data)
- inline unsigned **count** () const
- inline [DtListItem](#)* **first** () const
- inline [DtListItem](#)* **last** () const

Protected Fields

- [DtListItem](#)* **myFirst**
- [DtListItem](#)* **myLast**
- unsigned **myCount**

Protected Methods

- virtual void **addToList** ([DtListItem](#)* item, [DtListItem](#)* after)
Add item to the list after after
 - virtual void **removeFromList** ([DtListItem](#)* item)
Remove item from the list
-

Documentation

A DtList is a list of elements of arbitrary type. Data elements are added to, and extracted from a DtList as void*. DtList encloses each element in a DtListItem object. You can iterate through the elements in a DtList like this: for (DtListItem* item = list.first(); item; item = item->next()) { void* dataElement = item->data(); }

- **DtList()**
Constructor
- **virtual ~DtList()**
Destructor - does NOT free memory of data associated with the items. It only frees the items.
- **DtList(const [DtList](#)& orig)**
Copy constructor. Shallow copy. Copies the pointers you've added to the list, but not the items that are pointed to.
- **[DtList](#)& operator=(const [DtList](#)& orig)**
Assignment operator. Shallow copy. Copies the pointers you've added to the list, but not the items that are pointed to. Old list items are removed from the list, and the elements they contain are not deleted.
- **[DtListItem](#)* first() const**
Return first and last item on the list. DtListItem is an envelope for your data. The data can be obtained using DtListItem::data().
- **[DtListItem](#)* last() const**

- `unsigned count() const`
 Returns the number of elements in the list
- `DtListItem* add(void* data)`
 Add element to the end of the list. Returns the DtListItem envelope object that contains your new element.
- `DtListItem* addToStart(void* data)`
 Add element to the beginning of the list. Returns the DtListItem envelope object that contains your new element.
- `DtListItem* addToEnd(void* data)`
 Add element to the end of the list. Returns the DtListItem envelope object that contains your new element.
- `DtListItem* addBefore(void* data, DtListItem* before)`
 Add element to the list just before the indicated item. Returns the DtListItem envelope object that contains your new element.
- `DtListItem* addAfter(void* data, DtListItem* after)`
 Add element to the list just after the indicated item. Returns the DtListItem envelope object that contains your new element.
- `void* remove(DtListItem* item)`
 Remove an element from the list. Item is the envelope containing the data element you would like to remove.
 Note: remove does not free memory of data associated with the item. It only frees the item. The data is returned to the caller for deletion if required.
- `void removeAll()`
 Calls remove for all elements in the list. It does NOT delete the data in the list. Merely removes the data from the list.
- `DtListItem* itemLookup(void* data) const`
 Returns the DtListItem envelope object containing the indicated data element
- `virtual void addToList(DtListItem* item, DtListItem* after)`
 Add item to the list after after
- `virtual void removeFromList(DtListItem* item)`
 Remove item from the list. Does NOT delete the data contained in item.
- `DtListItem* myFirst`
- `DtListItem* myLast`
- `unsigned myCount`
- `inline DtListItem* add(void* data)`
- `inline DtListItem* addBefore(void* data, DtListItem* before)`
- `inline DtListItem* addToEnd(void* data)`
- `inline DtListItem* addToStart(void* data)`
- `inline unsigned count() const`
- `inline DtListItem* first() const`

 `inline DtListItem* last() const`

Direct child classes:

- [DtScript](#)
- [DtIntrusiveList](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file vllist.h:

class DtListItem

DtListItem is an envelope class used by DtList

Public Methods

-  **virtual** **~DtListItem ()**
Destructor
-  **[DtListItem](#)*** **prev () const**
Return previous (next) item
-  **[DtListItem](#)*** **next () const**
-  **void*** **data () const**
Returns the data element contained in the DtListItem envelope
-  **inline void*** **data () const**
-  **inline [DtListItem](#)*** **next () const**
-  **inline [DtListItem](#)*** **prev () const**
Inline Functions

Protected Fields

-  **[DtListItem](#)*** **myPrev**
-  **[DtListItem](#)*** **myNext**
-  **void*** **myData**

Protected Methods

-  **DtListItem (void* d)**
Constructor - called only by DtList
-

Documentation

DtListItem is an envelope class used by DtList. When you add an element to a DtList, it creates a DtListItem to contain the element. The DtListItem stores the next and previous pointer, as well as the data element itself.

-  **virtual ~DtListItem()**
Destructor
-  **[DtListItem](#)*** **prev() const**

Return previous (next) item. If you hit the beginning (end) of the list, returns NULL.

[DtListItem](#)* next() const

void* data() const

Returns the data element contained in the DtListItem envelope

DtListItem(void* d)

Constructor - called only by DtList

[DtListItem](#)* myPrev

[DtListItem](#)* myNext

void* myData

inline void* data() const

inline [DtListItem](#)* next() const

inline [DtListItem](#)* prev() const

Inline Functions

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file script.h:

class DtScript: public DtList

A DtScript is a DtList of StealthInstructions

Inheritance:

DtScript - DtList

Public Methods

- **DtScript** (char *fname)
 - **DtScript** ()
 - **~DtScript** ()
 - void **addInstruction** (DtTime time, [DtViewerCommand](#) command, char *args)
 - void **addInstruction** (DtTime time, [DtViewControlPdu](#) *pdu)
 - void **clear** ()
-

Inherited from DtList:

Public Methods

- [DtList](#)& **operator**=(const [DtList](#)& orig)
- [DtListItem](#)* **first**() const
- [DtListItem](#)* **last**() const
- unsigned **count**() const
- [DtListItem](#)* **add**(void* data)
- [DtListItem](#)* **addToStart**(void* data)
- [DtListItem](#)* **addToEnd**(void* data)
- [DtListItem](#)* **addBefore**(void* data, [DtListItem](#)* before)
- [DtListItem](#)* **addAfter**(void* data, [DtListItem](#)* after)

- void* **remove**([DtListItem](#)* item)
- void **removeAll**()
- [DtListItem](#)* **itemLookup**(void* data) const
- inline [DtListItem](#)* **add**(void* data)
- inline [DtListItem](#)* **addBefore**(void* data, [DtListItem](#)* before)
- inline [DtListItem](#)* **addToEnd**(void* data)
- inline [DtListItem](#)* **addToStart**(void* data)
- inline unsigned **count**() const
- inline [DtListItem](#)* **first**() const
- inline [DtListItem](#)* **last**() const

Protected Fields

- [DtListItem](#)* **myFirst**
- [DtListItem](#)* **myLast**
- unsigned **myCount**

Protected Methods

- virtual void **addToList**([DtListItem](#)* item, [DtListItem](#)* after)
- virtual void **removeFromList**([DtListItem](#)* item)

Documentation

A DtScript is a DtList of StealthInstructions

- `DtScript(char *fname)`
- `DtScript()`
- `~DtScript()`
- `void addInstruction(DtTime time, DtViewerCommand command, char *args)`
- `void addInstruction(DtTime time, DtViewControlPdu *pdu)`
- `void clear()`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vllist.h:

class DtIntrusiveList: public DtList

An intrusive list is a list that assumes that the elements that you add are pointers to instances of classes derived from DtListedObject

Inheritance:

DtIntrusiveList - DtList

Public Methods

- virtual void **addToList** ([DtListItem](#)* item, [DtListItem](#)* after)
Override of virtual function
 - virtual void **removeFromList** ([DtListItem](#)* item)
Override of virtual function
-

Inherited from DtList:

Public Methods

- [DtList](#)& **operator=**(const [DtList](#)& orig)
- [DtListItem](#)* **first**() const
- [DtListItem](#)* **last**() const
- unsigned **count**() const
- [DtListItem](#)* **add**(void* data)
- [DtListItem](#)* **addToStart**(void* data)
- [DtListItem](#)* **addToEnd**(void* data)
- [DtListItem](#)* **addBefore**(void* data, [DtListItem](#)* before)
- [DtListItem](#)* **addAfter**(void* data, [DtListItem](#)* after)
- void* **remove**([DtListItem](#)* item)
- void **removeAll**()

- [DtListItem](#)* **itemLookup**(void* data) const
- inline [DtListItem](#)* **add**(void* data)
- inline [DtListItem](#)* **addBefore**(void* data, [DtListItem](#)* before)
- inline [DtListItem](#)* **addToEnd**(void* data)
- inline [DtListItem](#)* **addToStart**(void* data)
- inline unsigned **count**() const
- inline [DtListItem](#)* **first**() const
- inline [DtListItem](#)* **last**() const

Protected Fields

- [DtListItem](#)* **myFirst**
 - [DtListItem](#)* **myLast**
 - unsigned **myCount**
-

Documentation

An intrusive list is a list that assumes that the elements that you add are pointers to instances of classes derived from DtListedObject. When an intrusive list is use, you can get previous and next pointers directly from the listed elements, (through the base DtListedObject class), without having to go through the DtList itself.

- **virtual void addToList**([DtListItem](#)* item, [DtListItem](#)* after)
Override of virtual function. Saves pointer to the list and to the item in the DtListedObj.
 - **virtual void removeFromList**([DtListItem](#)* item)
Override of virtual function. Removes the item from this list, and cleans up the DtListedObj's item and list pointers.
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file socket.h:

class DtSocket

Inheritance:

DtSocket

Public Fields

- int **blocking**
- int **pkts_rcvd**

Public Methods

- DtSocket** ()
- virtual **~DtSocket** ()
- virtual int **DtSend** (caddr_t packet, size_t size, DtInetAddr addr = DtUSE_DEFAULT_IP_ADDR)
- virtual int **sendTo** (caddr_t packet, size_t size, DtInetAddr addr = DtUSE_DEFAULT_IP_ADDR)
- virtual int **DtRecv** (caddr_t *buffptr, int *status=NULL)
Depending on the implementation in derived DtSockets, one of the forms of DtRecv may be faster than the other
- virtual int **DtRecv** (caddr_t buff, size_t)
- caddr_t **DtRecv** ()
for those that don't care about length:
- int **flush** ()
- void **setBundling** (int maxSize)
- int **isBundling** ()
- virtual void **setUnbundling** (int)
- virtual DtInetAddr **lastSourceInetAddr** ()
- virtual int **DtReportPktsRcvd** (void)
- virtual void **DtBecomeNonBlocking** (void)
- virtual void **notifyMe** ()
- virtual int **listenToMulticastGroup** (DtInetAddr)
- virtual int **dropMulticastGroup** (DtInetAddr)

- virtual int **setMcastTtl** (int)
- virtual [DtFilterTest](#)* **exerciseld** ()
- virtual [DtFilterTest](#)* **protocolVersion** ()
- virtual [DtFilterTest](#)* **pduKind** ()
- virtual [DtFilterTest](#)* **ipAddr** ()
- virtual int **readFd** ()
not all DtSocket have associated file descriptors those that do not will have readFd()/writeFd() return -1
- virtual int **writeFd** ()
- virtual int **isOk** ()

Protected Fields

- char **outBuff** [DtMaxPacketSize]
- int **outSize**
- int **bundling**
- int **maxBundleSize**

Protected Methods

- virtual [DtBoolean](#) **accept** (char *buf, DtInetAddr iaddr)

Private Fields

- [DtFilterTest](#) **version**
- [DtFilterTest](#) **exercise**
- [DtFilterTest](#) **kind**
- [DtFilterTest](#) **addr**

Documentation

- DtSocket** ()
- virtual **~DtSocket** ()
- virtual int **DtSend** (caddr_t packet, size_t size, DtInetAddr addr = DtUSE_DEFAULT_IP_ADDR)
- virtual int **sendTo** (caddr_t packet, size_t size, DtInetAddr addr =

DtUSE_DEFAULT_IP_ADDR)

virtual int DtRecv(caddr_t *buffptr, int *status=NULL)

Depending on the implementation in derived DtSockets, one of the forms of DtRecv may be faster than the other.

NOTE: if a derived class complains "... hides virtual DtRecv() ...", make sure you have BOTH overloads of DtRecv declared and defined in the subclass.

DtRecv has an optional status argument that could more elegantly be implemented as different return codes, but this would break older code.

virtual int DtRecv(caddr_t buff, size_t)

caddr_t DtRecv()

for those that don't care about length:

int flush()

void setBundling(int maxSize)

int isBundling()

virtual void setUnbundling(int)

virtual DtInetAddr lastSourceInetAddr()

virtual int DtReportPktsRcvd(void)

virtual void DtBecomeNonBlocking(void)

virtual void notifyMe()

virtual int listenToMulticastGroup(DtInetAddr)

virtual int dropMulticastGroup(DtInetAddr)

virtual int setMcastTtl(int)

virtual [DtFilterTest](#)* exerciseId()

virtual [DtFilterTest](#)* protocolVersion()

virtual [DtFilterTest](#)* pduKind()

virtual [DtFilterTest](#)* ipAddr()

int blocking

int pkts_rcvd

virtual int readFd()

not all DtSocket have associated file descriptors those that do not will have readFd()/writeFd() return -1

virtual int writeFd()

virtual int isOk()

virtual [DtBoolean](#) accept(char *buf, DtInetAddr iaddr)

char outBuff[DtMaxPacketSize]

- `int outSize`
- `int bundling`
- `int maxBundleSize`
- [DtFilterTest](#) `version`
- [DtFilterTest](#) `exercise`
- [DtFilterTest](#) `kind`
- [DtFilterTest](#) `addr`

Direct child classes:

- [DtTcpSocket](#)
- [DtSocketPairSocket](#)
- [DtNetSocket](#)
- [DtClientSocket](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file socket.h:

class DtFilterTest

Inheritance:

DtFilterTest

Public Methods

- **DtFilterTest** ()
- virtual **~DtFilterTest** ()
- void **disableTest** ()
- void **setTest** ([DtTestOperator](#) op, DtU32 value)
- void **extendTest** (DtU32 value)
- void **reduceTest** (DtU32 value)
- virtual void **setOperator** ([DtTestOperator](#) op)
- virtual void **setTest** ([DtTestOperator](#) op, int n, DtU32 *values)
- virtual void **extendTest** (int n, DtU32 *values)
- virtual void **reduceTest** (int n, DtU32 *values)
- virtual [DtBoolean](#) **evalTest** (char *buf, int offset, [DtFilterType](#) type)

Protected Fields

- [DtTestOperator](#) **testOp**
- [DtList](#) **values**

Protected Methods

- virtual void **clearList** ()
-

Documentation

- `DtFilterTest()`
 - `virtual ~DtFilterTest()`
 - `void disableTest()`
 - `void setTest(DtTestOperator op, DtU32 value)`
 - `void extendTest(DtU32 value)`
 - `void reduceTest(DtU32 value)`
 - `virtual void setOperator(DtTestOperator op)`
 - `virtual void setTest(DtTestOperator op, int n, DtU32 *values)`
 - `virtual void extendTest(int n, DtU32 *values)`
 - `virtual void reduceTest(int n, DtU32 *values)`
 - `virtual DtBoolean evalTest(char *buf, int offset, DtFilterType type)`
 - `virtual void clearList()`
 - `DtTestOperator testOp`
 - `DtList values`
-

Direct child classes:

[DtCsFilterTest](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file socket.h:

enum DtTestOperator

 Dto_EQ

 Dto_NEQ

Documentation

 Dto_EQ

 Dto_NEQ

In file socket.h:

enum DtFilterType

 DtFtlNetAddr

 DtFtNetU8

Documentation

 DtFtInetAddr

 DtFtNetU8

[Alphabetic index](#) [Hierarchy of classes](#)

In file clientsocket.h:

class DtCsFilterTest: public DtFilterTest

Inheritance:

DtCsFilterTest - DtFilterTest

Public Methods

- **DtCsFilterTest** (int fd, DtU8 filterType, DtU32 clientId)
- virtual **~DtCsFilterTest** ()
- void **setOperator** ([DtTestOperator](#) op)
- void **extendTest** (int n, DtU32 *values)
- void **reduceTest** (int n, DtU32 *vals)
- [DtBoolean](#) **evalTest** (char *, int, [DtFilterType](#))
No-op

Private Fields

- DtCsFilterMsg **filterMsg**
-

Inherited from DtFilterTest:

Public Methods

- void **disableTest**()
- void **setTest**([DtTestOperator](#) op, DtU32 value)
- virtual void **setTest**([DtTestOperator](#) op, int n, DtU32 *values)

Protected Fields

- [DtTestOperator](#) testOp
- [DtList](#) values

Protected Methods

virtual void **clearList**()

Documentation

DtCsFilterTest(int fd, DtU8 filterType, DtU32 clientId)

virtual ~DtCsFilterTest()

void setOperator([DtTestOperator](#) op)

void extendTest(int n, DtU32 *values)

void reduceTest(int n, DtU32 *vals)

[DtBoolean](#) evalTest(char *, int, [DtFilterType](#))

No-op. Real test done on server

DtCsFilterMsg filterMsg

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file tcpsocket.h:

class DtTcpSocket: public DtSocket

Inheritance:

DtTcpSocket - [DtSocket](#)

Public Fields

 **int myOk**

Public Methods

-  **DtTcpSocket** (int sockfd, struct sockaddr_in addr)
Server Constructor
-  **DtTcpSocket** (DtInetAddr addr, int port=DtDEFAULT_PORT, int nonBlockingConnect=0)
Client Constructor
-  **virtual ~DtTcpSocket** ()
Destructor
-  **int sendTo** (caddr_t packet, size_t packetSize, DtInetAddr addr=DtUSE_DEFAULT_IP_ADDR)
-  **int DtRecv** (caddr_t *buffptr, int *status=NULL)
-  **int DtRecv** (caddr_t buff, size_t bufsz)
-  **int readFd** ()
-  **int writeFd** ()
-  **void DtBecomeNonBlocking** (void)
designate the socket as non-blocking
-  **int isOk** ()
-  **int connectionEstablished** ()
Determines if the tcp socket has connected successfully
-  **int sizeOffset** ()
Get/Set mySizeOffset
-  **void setSizeOffset** (int offset)
-  **int sizeLength** ()
Get/Set mySizeLength
-  **void setSizeLength** (int length)

Private Fields

- char **myPktBuff** [2000]
- SOCKET **mySockfd**
- int **mySockfd**
- int **mySizeOffset**
- int **mySizeLength**
- static [DtBoolean](#) **theAnyInstancesFlag**

Private Methods

- int **recvWithStatus** (caddr_t *buffptr, int *status)
-

Inherited from [DtSocket](#):

Public Fields

- int **blocking**
- int **pkts_rcvd**

Public Methods

- virtual int **DtSend**(caddr_t packet, size_t size, DtInetAddr addr = DtUSE_DEFAULT_IP_ADDR)
- int **flush**()
- void **setBundling**(int maxSize)
- int **isBundling**()
- virtual void **setUnbundling**(int)
- virtual DtInetAddr **lastSourceInetAddr**()
- virtual int **DtReportPktsRcvd**(void)
- virtual void **notifyMe**()
- virtual int **listenToMulticastGroup**(DtInetAddr)
- virtual int **dropMulticastGroup**(DtInetAddr)
- virtual int **setMcastTtl**(int)
- virtual [DtFilterTest](#)* **exerciseld**()
- virtual [DtFilterTest](#)* **protocolVersion**()

virtual [DtFilterTest](#)* pduKind()

virtual [DtFilterTest](#)* ipAddr()

Protected Fields

char **outBuff**[DtMaxPacketSize]

int **outSize**

int **bundling**

int **maxBundleSize**

Protected Methods

virtual [DtBoolean](#) **accept**(char *buf, DtInetAddr iaddr)

Private Fields

[DtFilterTest](#) **version**

[DtFilterTest](#) **exercise**

[DtFilterTest](#) **kind**

[DtFilterTest](#) **addr**

Documentation

DtTcpSocket(int sockfd, struct sockaddr_in addr)
Server Constructor

DtTcpSocket(DtInetAddr addr, int port=DtDEFAULT_PORT, int nonBlockingConnect=0)
Client Constructor

virtual ~DtTcpSocket()
Destructor

int sendTo(caddr_t packet, size_t packetSize, DtInetAddr addr=DtUSE_DEFAULT_IP_ADDR)

int DtRecv(caddr_t *buffptr, int *status=NULL)

int DtRecv(caddr_t buff, size_t bufsize)

int readFd()

int writeFd()

void DtBecomeNonBlocking(void)
designate the socket as non-blocking

- `int isOk()`
- `int connectionEstablished()`
Determines if the tcp socket has connected successfully
- `int sizeOffset()`
Get/Set mySizeOffset
- `void setSizeOffset(int offset)`
- `int sizeLength()`
Get/Set mySizeLength
- `void setSizeLength(int length)`
- `int recvWithStatus(caddr_t *buffptr, int *status)`
- `char myPktBuff[2000]`
- `SOCKET mySockfd`
- `int mySockfd`
- `int myOk`
- `int mySizeOffset`
- `int mySizeLength`
- `static DtBoolean theAnyInstancesFlag`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file pairsock.h:

class DtSocketPairSocket: public DtSocket

Inheritance:

DtSocketPairSocket - DtSocket

Public Methods

- **DtSocketPairSocket** ()
- **DtSocketPairSocket** (int sockNum)
- int **sendTo** (caddr_t packet, size_t size, DtlnetAddr addr)
- int **recvWithStatus** (caddr_t buffptr, size_t size, int *status=NULL)
- int **DtRecv** (caddr_t *buffptr, int *status)
- int **DtRecv** (caddr_t buff, size_t size)
- void **DtBecomeNonBlocking** (void)
- int **otherFd** ()
- int **readFd** ()
- int **writeFd** ()

Private Fields

- int **myFd**
 - int **ofd**
 - char **buffer** [1500]
-

Inherited from DtSocket:

Public Fields

- int **blocking**
- int **pkts_rcvd**

Public Methods

- virtual int **DtSend**(caddr_t packet, size_t size, DtInetAddr addr = DtUSE_DEFAULT_IP_ADDR)
- int **flush**()
- void **setBundling**(int maxSize)
- int **isBundling**()
- virtual void **setUnbundling**(int)
- virtual DtInetAddr **lastSourceInetAddr**()
- virtual int **DtReportPktsRcvd**(void)
- virtual void **notifyMe**()
- virtual int **listenToMulticastGroup**(DtInetAddr)
- virtual int **dropMulticastGroup**(DtInetAddr)
- virtual int **setMcastTtl**(int)
- virtual [DtFilterTest](#)* **exerciseld**()
- virtual [DtFilterTest](#)* **protocolVersion**()
- virtual [DtFilterTest](#)* **pduKind**()
- virtual [DtFilterTest](#)* **ipAddr**()
- virtual int **isOk**()

Protected Fields

- char **outBuff**[DtMaxPacketSize]
- int **outSize**
- int **bundling**
- int **maxBundleSize**

Protected Methods

- virtual [DtBoolean](#) **accept**(char *buf, DtInetAddr iaddr)

Private Fields

- [DtFilterTest](#) **version**
- [DtFilterTest](#) **exercise**
- [DtFilterTest](#) **kind**

Documentation

-  `DtSocketPairSocket()`
-  `DtSocketPairSocket(int sockNum)`
-  `int sendTo(caddr_t packet, size_t size, DtInetAddr addr)`
-  `int recvWithStatus(caddr_t buffptr, size_t size, int *status=NULL)`
-  `int DtRecv(caddr_t *buffptr, int *status)`
-  `int DtRecv(caddr_t buff, size_t size)`
-  `void DtBecomeNonBlocking(void)`
-  `int otherFd()`
-  `int readFd()`
-  `int writeFd()`
-  `int myFd`
-  `int ofd`
-  `char buffer[1500]`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file netsocket.h:

class DtNetSocket: public DtSocket

Inheritance:

DtNetSocket - DtSocket

Public Fields

- static [DtInitErrorAction](#) **initErrorAction**
- static DtU32 **DtInetBroadcastMask**
- int **udpPort**
- char **outBuf** [2048]
- DtInetAddr **from_addr**

Public Methods

- DtNetSocket** (int port=DtDEFAULT_PORT, DtU8 flags=DtReadWrite)
- DtNetSocket** (int port, DtInetAddr destAddr, DtU8 flags)
No default for flags since 3rd arg is needed to disambiguate
- DtNetSocket** (int port, const struct in_addr *dest_addr, DtU8 flags=DtReadWrite)
This constructor is obsolete, and will eventually disappear Use DtInitNetSocket(port, dest_addr->s_addr) instead
- DtNetSocket** (char* dummy, int sockfd, DtU8 flags = DtReadWrite)
- DtNetSocket** (int port, DtInetAddr fromAddr, DtInetAddr destAddr, const char *device, DtU8 flags = DtReadWrite)
It only makes sense to set both destAddr and device on the GT
- virtual **~DtNetSocket** ()
- int **sendTo** (caddr_t packet, size_t size, DtInetAddr addr)
- int **DtRecv** (caddr_t buff, size_t)
- int **DtRecv** (caddr_t *buffptr, int *status=NULL)
- int **DtReportPktsRcvd** (void)
- DtInetAddr **lastSourceInetAddr** ()
- void **DtBecomeNonBlocking** (void)

void	setUnbundling (int onOff)
int	isUnbundling ()
int	dataLeftInCurrentBundle ()
int	readFd ()
int	writeFd ()
int	isOk ()
void	DtGetBroadcastMask ()
inline int	listenToMulticastGroup (DtInetAddr multicastAddr) <i>must be kept around for GT</i>
int	dropMulticastGroup (DtInetAddr multicastAddr)
int	setMcastTtl (int ttl)
int	DtGtPacketDequeue (caddr_t)
void	DtGtPacketEnqueue (char *)
int	DtGtSend (caddr_t, size_t) <i>No, but Ethernet does, and that's the only thing the GT100 will ever work with</i>

Protected Fields

void*	nextPdu
int	sizeLeft
int	unbundling
int	m_bExternalSocket
DtInetAddr	m_devAddr

Protected Methods

void	DtInitNetSocket (int port, DtInetAddr dest_addr, DtU8 rw_flags, const char *device)
int	recvWithStatus (caddr_t *buffptr, int *status)
int	recvFromNet (caddr_t *buffptr)
int	recvWithUnbundling (caddr_t *buffptr)

Private Fields

DtNetFloat64	pktBuff8 [DtMaxPacketSize/sizeof(DtNetFloat64)]
DtNetFloat64	unbundleBuff8 [DtMaxPacketSize/sizeof(DtNetFloat64)]
DtU8	RW_flags
int	boundToDestAddr
DtInetAddr	destInetAddr

- int **firstInBundle**
- int **firstExId**
- int **firstProtocol**
- int **ok**
- int **skt_num**
- GtQueueElement* **PacketQueueOut**
- GtQueueElement* **PacketQueueIn**

Private Methods

- int **init_iodev** (const char *)
- int **init_write** (DtInetAddr *dest)
- int **init_read** (DtInetAddr dest)
- int **PrivateListenToMulticastGroup** (DtInetAddr multicastAddr, int on_init=0)
- void **findNextPdu** ()
- unsigned int **currentHeaderMatchesFirstInBundle** ()
- int **DtGtRecv** (caddr_t bufp, size_t bufsize)

Private

- does DIS have a defined maximum?*

Inherited from [DtSocket](#):

Public Fields

- int **blocking**
- int **pkts_rcvd**

Public Methods

- virtual int **DtSend**(caddr_t packet, size_t size, DtInetAddr addr = DtUSE_DEFAULT_IP_ADDR)
- int **flush**()
- void **setBundling**(int maxSize)
- int **isBundling**()
- virtual void **notifyMe**()

- virtual [DtFilterTest](#)* **exerciseld()**
- virtual [DtFilterTest](#)* **protocolVersion()**
- virtual [DtFilterTest](#)* **pduKind()**
- virtual [DtFilterTest](#)* **ipAddr()**

Protected Fields

- char **outBuff**[DtMaxPacketSize]
- int **outSize**
- int **bundling**
- int **maxBundleSize**

Protected Methods

- virtual [DtBoolean](#) **accept**(char *buf, DtInetAddr iaddr)

Private Fields

- [DtFilterTest](#) **version**
 - [DtFilterTest](#) **exercise**
 - [DtFilterTest](#) **kind**
 - [DtFilterTest](#) **addr**
-

Documentation

- DtNetSocket(int port=DtDEFAULT_PORT, DtU8 flags=DtReadWrite)**
- DtNetSocket(int port, DtInetAddr destAddr, DtU8 flags)**
No default for flags since 3rd arg is needed to disambiguate. Note a real loss, however, since any point-to-point NetSocket should probably be write-only anyway.
- DtNetSocket(int port, const struct in_addr *dest_addr, DtU8 flags=DtReadWrite)**
This constructor is obsolete, and will eventually disappear Use DtInitNetSocket(port, dest_addr->s_addr) instead
- DtNetSocket(char* dummy, int sockfd, DtU8 flags = DtReadWrite)**
- DtNetSocket(int port, DtInetAddr fromAddr, DtInetAddr destAddr, const char *device, DtU8 flags = DtReadWrite)**
It only makes sense to set both destAddr and device on the GT. Under UNIX (with IP sockets), the device is implied by the destAddr It also only makes sense to specify from_addr on the _GT_. On UNIX boxes, the from IP address gets filled in automatically.

```

virtual ~DtNetSocket()

int sendTo(caddr_t packet, size_t size, DtInetAddr addr)

int DtRecv(caddr_t buff, size_t)

int DtRecv(caddr_t *buffptr, int *status=NULL)

int DtReportPktsRcvd(void)

DtInetAddr lastSourceInetAddr()

void DtBecomeNonBlocking(void)

void setUnbundling(int onOff)

int isUnbundling()

int dataLeftInCurrentBundle()

int readFd()

int writeFd()

int isOk()

static DtInitErrorAction initErrorAction

static DtU32 DtInetBroadcastMask

void DtGetBroadcastMask()

int udpPort

inline int listenToMulticastGroup(DtInetAddr multicastAddr)
    must be kept around for GT

int dropMulticastGroup(DtInetAddr multicastAddr)

int setMcastTtl(int ttl)

int DtGtPacketDequeue(caddr_t)

void DtGtPacketEnqueue(char *)

void DtInitNetSocket(int port, DtInetAddr dest_addr, DtU8 rw_flags, const char *device)

int recvWithStatus(caddr_t *buffptr, int *status)

int recvFromNet(caddr_t *buffptr)

int recvWithUnbundling(caddr_t *buffptr)

void* nextPdu

int sizeLeft

int unbundling

int m_bExternalSocket

DtInetAddr m_devAddr

```

- [DtNetFloat64](#) pktBuff8[DtMaxPacketSize/sizeof([DtNetFloat64](#))]
- [DtNetFloat64](#) unbundleBuff8[DtMaxPacketSize/sizeof([DtNetFloat64](#))]
- int init_iodev(const char *)
- int init_write(DtInetAddr *dest)
- int init_read(DtInetAddr dest)
- int PrivateListenToMulticastGroup(DtInetAddr multicastAddr, int on_init=0)
- void findNextPdu()
- unsigned int currentHeaderMatchesFirstInBundle()
- DtU8 RW_flags
- int boundToDestAddr
- DtInetAddr destInetAddr
- int firstInBundle
- int firstExId
- int firstProtocol
- int ok
- int skt_num
- char outBuf[2048]
-
- does DIS have a defined maximum?
- int DtGtSend(caddr_t, size_t)
- No, but Ethernet does, and that's the only thing the GT100 will ever work with
- int DtGtRecv(caddr_t bufp, size_t bufsize)
- GtQueueElement* PacketQueueOut
- GtQueueElement* PacketQueueIn
- DtInetAddr from_addr

Direct child classes:

[DtGroupSocket](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file netsocket.h:

enum DtInitErrorAction

-
-  DtInitErrorFatal
 -  DtInitErrorWarn
 -  DtInitErrorIgnore
-

Documentation

-  DtInitErrorFatal
-  DtInitErrorWarn
-  DtInitErrorIgnore

In file nettypes.h:

class DtNetFloat64

Network-order 64-bit IEEE-754 floating-point quantity

Public Fields

 [DtNetU32](#) mslong

 [DtNetU32](#) lslong
Most Significant Long

Public Methods

 **DtNetFloat64** ()
Least Significant Least

 **DtNetFloat64** (DtFloat64)

 **DtNetFloat64** (const [DtNetFloat64](#)&)

 DtFloat64 **operator =** (DtFloat64)

 **operator DtFloat64** () const

Documentation

Network-order 64-bit IEEE-754 floating-point quantity

 [DtNetU32](#) mslong

 [DtNetU32](#) lslong

Most Significant Long

 **DtNetFloat64** ()
Least Significant Least

 **DtNetFloat64** (DtFloat64)

 **DtNetFloat64** (const [DtNetFloat64](#)&)

 DtFloat64 **operator =** (DtFloat64)

 **operator DtFloat64** () const

This class has no child classes.

In file nettypes.h:

class DtNetU32

Network-order unsigned 32-bit quantity

Public Fields

-  **DtNetU16 msword**
-  **DtNetU16 lsword**
Most Significant Word

Public Methods

-  **DtNetU32 ()**
Least Significant Word
 -  **DtNetU32 (DtU32)**
 -  **DtNetU32 (const [DtNetU32](#)&)**
 -  **DtU32 operator = (DtU32)**
 -  **operator DtU32 () const**
-

Documentation

Network-order unsigned 32-bit quantity

-  **DtNetU16 msword**
 -  **DtNetU16 lsword**
Most Significant Word
 -  **DtNetU32 ()**
Least Significant Word
 -  **DtNetU32 (DtU32)**
 -  **DtNetU32 (const [DtNetU32](#)&)**
 -  **DtU32 operator = (DtU32)**
 -  **operator DtU32 () const**
-

This class has no child classes.

In file groupsocket.h:

class DtGroupSocket: public DtNetSocket

Inheritance:

DtGroupSocket - DtNetSocket - DtSocket

Public Methods

- **DtGroupSocket** (int port = DtDEFAULT_PORT, DtU8 flags = DtReadWrite)
Constructor - uses default address for listening
- **DtGroupSocket** (int port, DtInetAddr addr, DtU8 flags)
Constructor - uses addr for listening
- virtual **~DtGroupSocket** ()
Destructor
- virtual void **removeAllRemoteSites** ()
Clear the list of remote sites
- virtual int **DtSend** (caddr_t pkt, size_t size, DtInetAddr addr)
Virtual function override - send to all remote sites
- virtual void **addRemoteSite** (DtInetAddr addr)
Add/remove a remote site to send to
- virtual void **removeRemoteSite** (DtInetAddr addr)
- virtual int **sendToAllRemoteSites** (caddr_t pkt, size_t size, DtInetAddr dwSkipAddr = 0)
Send to all remote sites except dwSkipAddr
- virtual const [DtList](#)* **remoteSitesList** () const
Return the remote sites list
- inline const [DtList](#)* **remoteSitesList** () const

Protected Fields

- [DtList](#) **myRemoteSites**
DtList of DtInetAddrs (which do not need to be deleted)

Protected Methods

- [DtListItem](#)* **findItem** (DtInetAddr addr)
-

Inherited from [DtNetSocket](#):

Public Fields

- static [DtInitErrorAction](#) **initErrorAction**
- static DtU32 **DtlnetBroadcastMask**
- int **udpPort**
- char **outBuf**[2048]
- DtlnetAddr **from_addr**

Public Methods

- int **sendTo**(caddr_t packet, size_t size, DtlnetAddr addr)
- int **DtRecv**(caddr_t buff, size_t)
- int **DtRecv**(caddr_t *buffptr, int *status=NULL)
- int **DtReportPktsRcvd**(void)
- DtlnetAddr **lastSourceInetAddr**()
- void **DtBecomeNonBlocking**(void)
- void **setUnbundling**(int onOff)
- int **isUnbundling**()
- int **dataLeftInCurrentBundle**()
- int **readFd**()
- int **writeFd**()
- int **isOk**()
- void **DtGetBroadcastMask**()
- inline int **listenToMulticastGroup**(DtlnetAddr multicastAddr)
- int **dropMulticastGroup**(DtlnetAddr multicastAddr)
- int **setMcastTtl**(int ttl)
- int **DtGtPacketDequeue**(caddr_t)
- void **DtGtPacketEnqueue**(char *)
- int **DtGtSend**(caddr_t, size_t)

Protected Fields

- void* **nextPdu**
- int **sizeLeft**
- int **unbundling**
- int **m_bExternalSocket**
- DtInetAddr **m_devAddr**

Protected Methods

- void **DtInitNetSocket**(int port, DtInetAddr dest_addr, DtU8 rw_flags, const char *device)
- int **recvWithStatus**(caddr_t *buffptr, int *status)
- int **recvFromNet**(caddr_t *buffptr)
- int **recvWithUnbundling**(caddr_t *buffptr)

Private Fields

- [DtNetFloat64](#) **pktBuff8**[DtMaxPacketSize/sizeof([DtNetFloat64](#))]
- [DtNetFloat64](#) **unbundleBuff8**[DtMaxPacketSize/sizeof([DtNetFloat64](#))]
- DtU8 **RW_flags**
- int **boundToDestAddr**
- DtInetAddr **destInetAddr**
- int **firstInBundle**
- int **firstExId**
- int **firstProtocol**
- int **ok**
- int **skt_num**
- GtQueueElement* **PacketQueueOut**
- GtQueueElement* **PacketQueueIn**

Private Methods

- int **init_iodev**(const char *)
- int **init_write**(DtInetAddr *dest)
- int **init_read**(DtInetAddr dest)

- int **PrivateListenToMulticastGroup**(DtInetAddr multicastAddr, int on_init=0)
- void **findNextPdu**()
- unsigned int **currentHeaderMatchesFirstInBundle**()
- int **DtGtRecv**(caddr_t bufp, size_t bufsz)

Private

Inherited from [DtSocket](#):

Public Fields

- int **blocking**
- int **pkts_rcvd**

Public Methods

- int **flush**()
- void **setBundling**(int maxSize)
- int **isBundling**()
- virtual void **notifyMe**()
- virtual [DtFilterTest](#)* **exerciseld**()
- virtual [DtFilterTest](#)* **protocolVersion**()
- virtual [DtFilterTest](#)* **pduKind**()
- virtual [DtFilterTest](#)* **ipAddr**()

Protected Fields

- char **outBuff**[DtMaxPacketSize]
- int **outSize**
- int **bundling**
- int **maxBundleSize**

Protected Methods

- virtual [DtBoolean](#) **accept**(char *buf, DtInetAddr iaddr)

Private Fields

- [DtFilterTest](#) **version**

- [DtFilterTest](#) **exercise**

- [DtFilterTest](#) **kind**

- [DtFilterTest](#) **addr**

Documentation

- DtGroupSocket(int port = DtDEFAULT_PORT, DtU8 flags = DtReadWrite)**

Constructor - uses default address for listening. Will not send to any addresses until addRemoteSite is called.

- DtGroupSocket(int port, DtInetAddr addr, DtU8 flags)**

Constructor - uses addr for listening. Will not send to any addresses until addRemoteSite is called.

- virtual ~DtGroupSocket()**

Destructor

- virtual void removeAllRemoteSites()**

Clear the list of remote sites

- virtual int DtSend(caddr_t pkt, size_t size, DtInetAddr addr)**

Virtual function override - send to all remote sites

- virtual void addRemoteSite(DtInetAddr addr)**

Add/remove a remote site to send to

- virtual void removeRemoteSite(DtInetAddr addr)**

- virtual int sendToAllRemoteSites(caddr_t pkt, size_t size, DtInetAddr dwSkipAddr = 0)**

Send to all remote sites except dwSkipAddr

- virtual const [DtList](#)* remoteSitesList() const**

Return the remote sites list

- [DtListItem](#)* findItem(DtInetAddr addr)**

- [DtList](#) myRemoteSites**

DtList of DtInetAddrs (which do not need to be deleted)

- inline const [DtList](#)* remoteSitesList() const**

This class has no child classes.

In file clientsocket.h:

class DtClientSocket: public DtSocket

Inheritance:

DtClientSocket - [DtSocket](#)

Public Fields

• `endif static DtTime psMsgTimeout`

Public Methods

- **DtClientSocket** (int port, int recvQSize = DtCsQDefaultSize, unsigned short csFlags = DtCsFlagsDefault)
- **DtClientSocket** (int port, DtInetAddr destAddr, int recvQSize = DtCsQDefaultSize, unsigned short csFlags = DtCsFlagsDefault)
- **~DtClientSocket** ()
- int **sendTo** (caddr_t packet, size_t size, DtInetAddr addr)
- int **DtRecv** (caddr_t, size_t)
- int **DtRecv** (caddr_t *buffptr, int *status=NULL)
this is slower as it does a memcpy()
- int **listenToMulticastGroup** (DtInetAddr multicastAddr)
- int **dropMulticastGroup** (DtInetAddr multicastAddr)
- int **setMcastTtl** (int ttl)
- void **setUnbundling** (int onOff)
- DtInetAddr **lastSourceInetAddr** ()
int isUnbundling(); is not currently implemented
- void **DtBecomeNonBlocking** (void)
- int **readFd** ()
- int **writeFd** ()
- int **isOk** ()
- [DtBoolean](#) **isInternal** ()
- void **setPsMsgTimeout** (int seconds)

- void **notifyMe** ()
- void **disconnectFromServer** ()
- void **setPacketServerInternal** ()
- void **setPacketServerExternal** ()
- void **pingClients** ()
- void **testClients** ()
- void **terminatePacketServer** ()
- void **requestServerState** ()
- void **requestServerCStates** ()
- void **setPacketServerUnbundling** (int onOff)
- [DtFilterTest](#)* **exerciseld** ()
- [DtFilterTest](#)* **protocolVersion** ()
- [DtFilterTest](#)* **pduKind** ()
- [DtFilterTest](#)* **ipAddr** ()

Protected Fields

- [DtShmMsgQueueReader](#)* **reader**
- [DtNetSocket](#)* **netSock**
- [DtBoolean](#) **notifyPending**
- [DtBoolean](#) **ok**
- int **recvFd**
- int **sendFd**
- long **clientId**

Protected Methods

- void **init** (int port, DtlNetAddr destAddr, int recvQSize, unsigned short csFlags)
- int **timedRecv** (caddr_t buf, size_t size, char *psKind)
- void **drainNotifySocket** ()
- void **checkGoAway** ()

Private Fields

- [DtCsFilterTest](#)* **exercise**
- [DtCsFilterTest](#)* **version**

 DtCsFilterTest *	kind
 DtCsFilterTest *	addr
 DtInetAddr	destAddr
 char	clientSocketPath [DtCsClientPathSize]
 int	serverPort
 DtBoolean	internal
 DtInetAddr	sourceAddr
 DtNetFloat64	pktBuff8 [(DtMaxPacketSize + sizeof(DtCsServerPacket) + 7) / sizeof(DtNetFloat64)]
 char*	shmAddr
 DtCsControlReqMsg*	controlMsg

Private Methods

 void	processControlMsg (DtCsMsg *msg)
 DtCsMsg*	getPsMsg (DtCsMsg *buf, char desiredKind)
 void	cleanup (void)

Inherited from [DtSocket](#):

Public Fields

 int	blocking
 int	pkts_rcvd

Public Methods

 virtual int	DtSend (caddr_t packet, size_t size, DtInetAddr addr = DtUSE_DEFAULT_IP_ADDR)
 int	flush ()
 void	setBundling (int maxSize)
 int	isBundling ()
 virtual int	DtReportPktsRcvd (void)

Protected Fields

 char	outBuff [DtMaxPacketSize]
 int	outSize

- `int bundling`
- `int maxBundleSize`

Protected Methods

- virtual [DtBoolean](#) `accept(char *buf, DtInetAddr iaddr)`
-

Documentation

- `DtClientSocket(int port, int recvQSize = DtCsQDefaultSize, unsigned short csFlags = DtCsFlagsDefault)`
- `DtClientSocket(int port, DtInetAddr destAddr, int recvQSize = DtCsQDefaultSize, unsigned short csFlags = DtCsFlagsDefault)`
- `~DtClientSocket()`
- `int sendTo(caddr_t packet, size_t size, DtInetAddr addr)`
- `int DtRecv(caddr_t, size_t)`
- `int DtRecv(caddr_t *bufptr, int *status=NULL)`
this is slower as it does a memcpy()
- `int listenToMulticastGroup(DtInetAddr multicastAddr)`
- `int dropMulticastGroup(DtInetAddr multicastAddr)`
- `int setMcastTtl(int ttl)`
- `void setUnbundling(int onOff)`
- `DtInetAddr lastSourceInetAddr()`
int isUnbundling(); is not currently implemented. The answer depends on the state of the Packet Server.
- `void DtBecomeNonBlocking(void)`
- `int readFd()`
- `int writeFd()`
- `int isOk()`
- [DtBoolean](#) `isInternal()`
- `void setPsMsgTimeout(int seconds)`
- `void notifyMe()`
- `void disconnectFromServer()`
- `void setPacketServerInternal()`
- `void setPacketServerExternal()`

```

void pingClients()

void testClients()

void terminatePacketServer()

void requestServerState()

void requestServerCStates()

void setPacketServerUnbundling(int onOff)

DtFilterTest* exerciseId()

DtFilterTest* protocolVersion()

DtFilterTest* pduKind()

DtFilterTest* ipAddr()

endif static DtTime psMsgTimeout

void init(int port, DtInetAddr destAddr, int recvQSize, unsigned short csFlags)

int timedRecv(caddr_t buf, size_t size, char *psKind)

void drainNotifySocket()

void checkGoAway()

DtShmMsgQueueReader* reader

DtNetSocket* netSock

DtBoolean notifyPending

DtBoolean ok

int recvFd

int sendFd

long clientId

void processControlMsg(DtCsMsg *msg)

DtCsMsg* getPMsg(DtCsMsg *buf, char desiredKind)

void cleanup(void)

DtCsFilterTest* exercise

DtCsFilterTest* version

DtCsFilterTest* kind

DtCsFilterTest* addr

DtInetAddr destAddr

char clientSocketPath[DtCsClientPathSize]

```

- `int serverPort`
- `DtBoolean internal`
- `DtInetAddr sourceAddr`
- `DtNetFloat64 pktBuff8[(DtMaxPacketSize + sizeof(DtCsServerPacket) + 7) / sizeof(DtNetFloat64) ]`
- `char* shmAddr`
- `DtCsControlReqMsg* controlMsg`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file pdufactory.h:

class DtPduFactory

Inheritance:

DtPduFactory

Public Methods

	DtPduFactory () <i>default constructor</i>
virtual	~DtPduFactory () <i>destructor</i>
	DtPduFactory (const DtPduFactory & orig) <i>copy constructor</i>
DtPduFactory &	operator= (const DtPduFactory & orig) <i>assignment operator</i>
virtual DtPdu *	createPdu (const DtNetPacket* packet, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER) <i>Create the specified pdu</i>
virtual DtPduCreator	addCreator (DtPduKind kind, DtPduCreator creator) <i>Add a pdu creation function to the factory</i>
virtual int	packetGood (const DtNetPacket* packet) const <i>Returns whether packet is appropriate DIS version</i>
static int	dfltPacketGood (const DtNetPacket* packet)

Protected Fields

DtPduCreator*	myCreators <i>myCreators is an array of theMaxPduKinds items</i>
static const int	theMaxPduKinds

Documentation

	DtPduFactory () default constructor
virtual	~DtPduFactory ()

destructor

DtPduFactory(const [DtPduFactory](#)& orig)

copy constructor

DtPduFactory& operator=(const [DtPduFactory](#)& orig)

assignment operator

virtual [DtPdu](#)* createPdu(const DtNetPacket* packet, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)

Create the specified pdu. Note: Caller is responsible for deleting. Returns NULL if pdu creation function is not in the factory.

virtual DtPduCreator addCreator([DtPduKind](#) kind, DtPduCreator creator)

Add a pdu creation function to the factory. The function must return a new'ed pdu object which will be deleted by the user. Returns any previous pdu creation function which was replaced by the add.

virtual int packetGood(const DtNetPacket* packet) const

Returns whether packet is appropriate DIS version

static int dfltPacketGood(const DtNetPacket* packet)

DtPduCreator* myCreators

myCreators is an array of theMaxPduKinds items

static const int theMaxPduKinds

Direct child classes:

[DtDisPduFactory](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file `dispdufactory.h`:

class DtDisPduFactory: public DtPduFactory

class DtDisPduFactory: Instances of DtDisPduFactory are used to create DIS PDUs

Inheritance:

DtDisPduFactory - DtPduFactory

Public Methods

- **DtDisPduFactory** ()
default constructor Adds creators for the various DIS pdus
 - virtual **~DtDisPduFactory** ()
destructor
 - **DtDisPduFactory** (const [DtDisPduFactory](#)& orig)
copy constructor
 - [DtDisPduFactory](#)& **operator=** (const [DtDisPduFactory](#)& orig)
assignment operator
 - static void **addDis203Creators** ([DtPduFactory](#)* fact)
Adds creators for all DIS 2
 - static void **addDis204Creators** ([DtPduFactory](#)* fact)
Adds creators for all DIS 2
 - static void **addDis214Creators** ([DtPduFactory](#)* fact)
Adds creators for all DIS 2
-

Inherited from DtPduFactory:

Public Methods

- virtual [DtPdu](#)* **createPdu**(const DtNetPacket* packet, DtBufferPtr buffer = DtUSE_INTERNAL_BUFFER)
- virtual DtPduCreator **addCreator**([DtPduKind](#) kind, DtPduCreator creator)
- virtual int **packetGood**(const DtNetPacket* packet) const
- static int **dfltPacketGood**(const DtNetPacket* packet)

Protected Fields

- DtPduCreator* **myCreators**
 - static const int **theMaxPduKinds**
-

Documentation

class DtDisPduFactory:

Instances of DtDisPduFactory are used to create DIS PDUs. This class knows how to create the various DIS pdus.

- **DtDisPduFactory()**
default constructor Adds creators for the various DIS pdus
 - **virtual ~DtDisPduFactory()**
destructor
 - **DtDisPduFactory(const [DtDisPduFactory](#)& orig)**
copy constructor
 - **[DtDisPduFactory](#)& operator=(const [DtDisPduFactory](#)& orig)**
assignment operator
 - **static void addDis203Creators([DtPduFactory](#)* fact)**
Adds creators for all DIS 2.0.3 PDUs to fact. Includes ViewControl and LgrControl
 - **static void addDis204Creators([DtPduFactory](#)* fact)**
Adds creators for all DIS 2.0.4 / IEEE1278.1 PDUs to fact. Includes ViewControl and LgrControl
 - **static void addDis214Creators([DtPduFactory](#)* fact)**
Adds creators for all DIS 2.1.4 / IEEE1278.1a PDUs to fact. (Does not include 2.0.4 / IEEE1278.1 PDUs).
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file pducbmgr.h:

class DtPduCallbackManager

class DtPduCallbackManager: Instances of DtPduCallbackManager are used to manage callbacks on the PDU kinds for the exercise connection

Public Methods

	DtPduCallbackManager () <i>default constructor</i>
virtual	~DtPduCallbackManager () <i>destructor</i>
	DtPduCallbackManager (const DtPduCallbackManager & orig) <i>copy constructor</i>
DtPduCallbackManager &	operator= (const DtPduCallbackManager & orig) <i>assignment operator</i>
virtual void	addPduCallback (DtPduKind k, DtPduCallbackFcn fcn, void* usr) <i>Add/Remove a function to be called when a particular PDU kind is received</i>
virtual void	removePduCallback (DtPduKind k, DtPduCallbackFcn fcn, void* usr)
virtual void	processPdu (const DtPdu & pdu) const <i>Call the callbacks registered for the PDU's kind</i>
virtual int	numCallbacks (DtPduKind k) const <i>Get the number of callbacks registered for the PDU kind</i>

Protected Fields

DtList	myCbList [DtMAX_PDU_KINDS]
------------------------	-----------------------------------

Protected Methods

virtual void	addPduCallback (DtPduKind k, DtPduCallbackInfo * info) <i>Add/Remove callback info for a particular PDU kind</i>
virtual void	removePduCallback (DtPduKind k, const DtPduCallbackInfo & info)
void	emptyList () <i>Empty the list</i>
void	copyList (const DtPduCallbackManager & orig) <i>Copy the list</i>

Protected

• `#define DtMAX_PDU_KINDS`

Documentation

class DtPduCallbackManager: Instances of DtPduCallbackManager are used to manage callbacks on the PDU kinds for the exercise connection

- `DtPduCallbackManager()`
default constructor
- `virtual ~DtPduCallbackManager()`
destructor
- `DtPduCallbackManager(const DtPduCallbackManager& orig)`
copy constructor
- `DtPduCallbackManager& operator=(const DtPduCallbackManager& orig)`
assignment operator
- `virtual void addPduCallback(DtPduKind k, DtPduCallbackFcn fcn, void* usr)`
Add/Remove a function to be called when a particular PDU kind is received
- `virtual void removePduCallback(DtPduKind k, DtPduCallbackFcn fcn, void* usr)`
- `virtual void processPdu(const DtPdu& pdu) const`
Call the callbacks registered for the PDU's kind
- `virtual int numCallbacks(DtPduKind k) const`
Get the number of callbacks registered for the PDU kind
- `virtual void addPduCallback(DtPduKind k, DtPduCallbackInfo* info)`
Add/Remove callback info for a particular PDU kind
- `virtual void removePduCallback(DtPduKind k, const DtPduCallbackInfo& info)`
- `void emptyList()`
Empty the list. Non-virtual since called in dtor.
- `void copyList(const DtPduCallbackManager& orig)`
Copy the list. Non-virtual since called in ctor.
- `#define DtMAX_PDU_KINDS`
- `DtList myCbList[DtMAX_PDU_KINDS]`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file pducbinfo.h:

class DtPduCallbackInfo

class DtPduCallbackInfo: Instances of DtPduCallbackInfo are used by DtPduCallbackManager to keep track of PDU callback functions

Public Methods

	DtPduCallbackInfo (DtPduCallbackFcn func, void* usr) <i>constructor</i>
virtual	~DtPduCallbackInfo () <i>destructor</i>
	DtPduCallbackInfo (const DtPduCallbackInfo & orig) <i>copy constructor</i>
DtPduCallbackInfo &	operator= (const DtPduCallbackInfo & orig) <i>assignment operator</i>
virtual int	match (const DtPduCallbackInfo & other) const <i>Get whether one callback info equals another</i>
virtual DtPduCallbackFcn	func () const <i>Get the callback function</i>
virtual void*	usr () const <i>Get the user data</i>
inline DtPduCallbackFcn	func () const
inline void*	usr () const

Protected Fields

DtPduCallbackFcn	myFunc
void*	myUsr

Documentation

class DtPduCallbackInfo: Instances of DtPduCallbackInfo are used by DtPduCallbackManager to keep track of PDU callback functions

	DtPduCallbackInfo (DtPduCallbackFcn func, void* usr) constructor
virtual	~DtPduCallbackInfo () destructor
	DtPduCallbackInfo (const DtPduCallbackInfo & orig)

copy constructor

- `DtPduCallbackInfo& operator=(const DtPduCallbackInfo& orig)`

assignment operator

- `virtual int match(const DtPduCallbackInfo& other) const`

Get whether one callback info equals another

- `virtual DtPduCallbackFcn func() const`

Get the callback function

- `virtual void* usr() const`

Get the user data

- `DtPduCallbackFcn myFunc`

- `void* myUsr`

- `inline DtPduCallbackFcn func() const`

- `inline void* usr() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file hashlist.h:

class DtHashlist

Public Methods

	DtHashlist (int numBuckets = dfItNumBuckets) <i>Default constructor</i>
	DtHashlist (DtList * list, int numBuckets = dfItNumBuckets) <i>Constructor for a DtHashlist that uses an external DtList constructed by the user</i>
virtual	~DtHashlist () <i>Destructor</i>
	DtHashlist (const DtHashlist & orig) <i>Copy constructor - always uses an internal list, so if you want your copy to use an external list, do not use the copy constructor</i>
DtHashlist &	operator= (const DtHashlist & orig) <i>Assignment operator - Does not replace the DtHashlist's list</i>
virtual DtListItem *	add (const DtBaseHashKey & key, void* data, void** oldData = NULL) <i>Add something to the list</i>
virtual void*	remove (const DtBaseHashKey & key) <i>Remove something from the list</i>
virtual void*	lookup (const DtBaseHashKey & key) const <i>Lookup something in the list, return data pointer</i>
virtual void*	operator[] (const DtBaseHashKey & key) const
virtual const DtList *	list () const <i>Returns a DtList of void**s which are the data that you have added to the hashlist (independent of hash keys)</i>
virtual void	print () const <i>Prints the data in the hashlist</i>
virtual const DtList *	keys () const <i>Returns a DtList of DtHashKey**s representing the set of keys with which data is currently associated</i>
virtual void	empty () <i>Remove all keys and data from the list</i>
virtual void	setFromOther (const DtHashlist & orig) <i>Sets the contents of this hashlist to match those of other</i>
inline void*	operator[] (const DtBaseHashKey & key) const

Protected Fields

int	myNumBuckets
const DtBoolean	myListIsMine
DtList *	myList <i>Regular DtList containing all of the items added to the hashlist, without regard to key information</i>

- [DtList*](#) **myBuckets**
Array of buckets (DtLists of DtHashItems)
- [DtList](#) **myKeys**
DtList of keys (DtBaseHashKeys)
- static int **theNumBucketsDefault**

Protected Methods

- virtual [DtListItem*](#) **itemLookup** (const [DtBaseHashKey](#)& key) const
Lookup something in the list, return DtListItem
 - virtual [DtList](#)& **bucketFromHashKey** (const [DtBaseHashKey](#)& key) const
Determine the bucket from the hashkey
 - virtual [DtListItem*](#) **bucketItemLookup** (const [DtBaseHashKey](#)& key) const
Lookup an item in a bucket
-

Documentation

- **DtHashlist(int numBuckets = dfltNumBuckets)**
Default constructor. Creates and uses a regular DtList to store the list of data items that you add.
- **DtHashlist([DtList*](#) list, int numBuckets = dfltNumBuckets)**
Constructor for a DtHashlist that uses an external DtList constructed by the user
- virtual **~DtHashlist()**
Destructor
- **DtHashlist(const [DtHashlist](#)& orig)**
Copy constructor - always uses an internal list, so if you want your copy to use an external list, do not use the copy constructor. Instead, use the standard with-external-list constructor to create an empty hash list, then use setFromOther or the assignment operator to copy the contents.
- **[DtHashlist](#)& operator=(const [DtHashlist](#)& orig)**
Assignment operator - Does not replace the DtHashlist's list. Just empties the DtHashlist, and then sets its contents to those of the original using setFromOther.
- virtual **[DtListItem*](#) add(const [DtBaseHashKey](#)& key, void* data, void** oldData = NULL)**
Add something to the list. The actual pointer is stored, so you should make sure the pointer remains valid until after you remove the data from the hash list. At any given time, only one data item may be associated with a particular key. If add is called with a key with which an item is already associated, the old item is replaced with the new, and the old item is returned in the optional oldData argument for deletion by the user. Otherwise, oldData will be set to NULL.
- virtual **void* remove(const [DtBaseHashKey](#)& key)**
Remove something from the list. Returns the data that was associated with the key, so that you can cast it to the right type and delete it.
- virtual **void* lookup(const [DtBaseHashKey](#)& key) const**
Lookup something in the list, return data pointer

- `virtual void* operator[] (const DtBaseHashKey& key) const`
- `virtual const DtList* list() const`
Returns a DtList of void*'s which are the data that you have added to the hashlist (independent of hash keys)
- `virtual void print() const`
Prints the data in the hashlist.
- `virtual const DtList* keys() const`
Returns a DtList of DtHashKey*'s representing the set of keys with which data is currently associated
- `virtual void empty()`
Remove all keys and data from the list
- `virtual void setFromOther(const DtHashlist& orig)`
Sets the contents of this hashlist to match those of other. Data pointers that have been added to orig are copied, but the items that they point to are not.
- `virtual DtListItem* itemLookup(const DtBaseHashKey& key) const`
Lookup something in the list, return DtListItem
- `virtual DtList& bucketFromHashKey(const DtBaseHashKey& key) const`
Determine the bucket from the hashkey
- `virtual DtListItem* bucketItemLookup(const DtBaseHashKey& key) const`
Lookup an item in a bucket
- `int myNumBuckets`
- `const DtBoolean myListIsMine`
- `DtList* myList`
Regular DtList containing all of the items added to the hashlist, without regard to key information
- `DtList* myBuckets`
Array of buckets (DtLists of DtHashItems)
- `DtList myKeys`
DtList of keys (DtBaseHashKeys)
- `static int theNumBucketsDefault`
- `inline void* operator[] (const DtBaseHashKey& key) const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file basehashkey.h:

class DtBaseHashKey

class DtBaseHashKey: DtBaseHashKey is an abstract base class for hashkeys

Inheritance:

DtBaseHashKey

Public Methods

	DtBaseHashKey () <i>default constructor</i>
 virtual	~DtBaseHashKey () <i>destructor</i>
	DtBaseHashKey (const DtBaseHashKey & orig) <i>copy constructor</i>
 DtBaseHashKey &	operator= (const DtBaseHashKey & orig) <i>assignment operator</i>
 virtual DtBoolean	operator== (const DtBaseHashKey & other) const <i>equality operator</i>
 virtual DtBoolean	operator!= (const DtBaseHashKey & other) const <i>equality operator</i>
 virtual DtU32	hashValue () const <i>get the hashed value for this key</i>
 virtual DtBaseHashKey *	clone () const <i>make a copy of this hash key</i>
 virtual const DtString &	stringRep () const <i>get the string representation of this key</i>

Protected Fields

 DtBoolean	myHashValueIsComputed
 DtU32	myHashValue

Protected Methods

 virtual DtU32	computeHashValue () <i>compute the hashed value for this key</i>
---	--

Documentation

class DtBaseHashKey: DtBaseHashKey is an abstract base class for hashkeys

- **DtBaseHashKey()**
default constructor
- **virtual ~DtBaseHashKey()**
destructor
- **DtBaseHashKey(const [DtBaseHashKey](#)& orig)**
copy constructor
- **[DtBaseHashKey](#)& operator=(const [DtBaseHashKey](#)& orig)**
assignment operator
- **virtual [DtBoolean](#) operator==(const [DtBaseHashKey](#)& other) const**
equality operator
- **virtual [DtBoolean](#) operator!=(const [DtBaseHashKey](#)& other) const**
equality operator
- **virtual DtU32 hashValue() const**
get the hashed value for this key
- **virtual [DtBaseHashKey](#)* clone() const**
make a copy of this hash key
- **virtual const [DtString](#)& stringRep() const**
get the string representation of this key
- **virtual DtU32 computeHashValue()**
compute the hashed value for this key
- **[DtBoolean](#) myHashValueIsComputed**
- **DtU32 myHashValue**

Direct child classes:

[DtSystemHashKey](#)
[DtStringHashKey](#)
[DtObjectIdHashKey](#)
[DtIntHashKey](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file str.h:

class DtString

class DtString: Instances of DtString are

Inheritance:

DtString

Public Methods

	DtString () <i>default constructor</i>
	DtString (const char* str)
 virtual	~DtString () <i>destructor</i>
	DtString (const DtString& orig) <i>copy constructor</i>
 DtString &	operator= (const DtString& orig) <i>assignment operators</i>
 DtString &	operator= (const char* orig)
 virtual	operator const char* () const <i>cast operator</i>
 friend DtBoolean	operator== (const DtString& str1, const DtString& str2) <i>comparison operators</i>
 friend DtBoolean	operator< (const DtString& str1, const DtString& str2)
 friend DtBoolean	operator<= (const DtString& str1, const DtString& str2)
 friend DtBoolean	operator> (const DtString& str1, const DtString& str2)
 friend DtBoolean	operator>= (const DtString& str1, const DtString& str2)
 friend DtBoolean	operator!= (const DtString& str1, const DtString& str2)
 virtual DtBoolean	operator== (const char* str) const
 virtual DtBoolean	operator< (const char* str) const
 virtual DtBoolean	operator<= (const char* str) const
 virtual DtBoolean	operator> (const char* str) const
 virtual DtBoolean	operator>= (const char* str) const

virtual DtBoolean	operator!= (const char* str) const
virtual DtBoolean	caseCompare (const DtString & str) <i>Case-insensitive string compare</i>
friend DtString	operator+ (const DtString & str1, const DtString & str2) <i>concatenation operators</i>
virtual DtString	operator+ (const char* str) const
virtual DtString &	operator+= (const char* str)
virtual DtString &	operator+= (const DtString & str)
virtual const char&	operator[] (int index) const <i>index operator</i>
virtual char&	operator[] (int index)
virtual DtBoolean	snipFront (const char ch, DtBoolean toLastCh = DtFalse) <i>truncates everything before or after the character, including the character, respectively</i> <i>Normally, snipFront stops at the first occurrence of "ch", and snipBack stops at the last occurrence of "ch"</i>
virtual DtBoolean	snipBack (const char ch, DtBoolean fromFirstCh = DtFalse)
DtBoolean	snipFrontToLast (const char ch)
DtBoolean	snipBackToFirst (const char ch)
virtual int	count (const char ch) const <i>returns the number of occurrences of the character</i>
virtual int	length () const <i>return the length of the string</i>
const char*	string () const <i>returns the char* internal representation of the string</i>
static const DtString &	nullString ()

Protected Fields

char*	myString
static const int	theMaxInsertionStringSize

Protected Methods

void	clear () <i>deletes the string, and sets the myString pointer to NULL non-virtual because only called by destructor, and non-virtual members myString is set to NULL, so it should be allocated some memory if operators are to be used on the string</i>
void	setString (const char* str) <i>Set/Get for myString - char pointer that holds the string non-virtual because only called by constructor</i>

Private Methods

- friend ostream& **operator<<** (ostream& os, const [DtString](#)& str)
 - friend istream& **operator>>** (istream& is, [DtString](#)& str)
-

Documentation

class DtString: Instances of DtString are

- friend ostream& **operator<<**(ostream& os, const [DtString](#)& str)
- friend istream& **operator>>**(istream& is, [DtString](#)& str)
- **DtString()**
default constructor
- **DtString(const char* str)**
- **virtual ~DtString()**
destructor
- **DtString(const [DtString](#)& orig)**
copy constructor
- **[DtString](#)& operator=(const [DtString](#)& orig)**
assignment operators
- **[DtString](#)& operator=(const char* orig)**
- **virtual operator const char*() const**
cast operator
- friend [DtBoolean](#) **operator==(const [DtString](#)& str1, const [DtString](#)& str2)**
comparison operators
- friend [DtBoolean](#) **operator<(const [DtString](#)& str1, const [DtString](#)& str2)**
- friend [DtBoolean](#) **operator<=(const [DtString](#)& str1, const [DtString](#)& str2)**
- friend [DtBoolean](#) **operator>(const [DtString](#)& str1, const [DtString](#)& str2)**
- friend [DtBoolean](#) **operator>=(const [DtString](#)& str1, const [DtString](#)& str2)**
- friend [DtBoolean](#) **operator!=(const [DtString](#)& str1, const [DtString](#)& str2)**
- **virtual [DtBoolean](#) operator==(const char* str) const**
- **virtual [DtBoolean](#) operator<(const char* str) const**
- **virtual [DtBoolean](#) operator<=(const char* str) const**

- `virtual DtBoolean operator>(const char* str) const`
- `virtual DtBoolean operator>=(const char* str) const`
- `virtual DtBoolean operator!=(const char* str) const`
- `virtual DtBoolean caseCompare(const DtString& str)`
Case-insensitive string compare. Returns DtTrue if the strings are equal except for differences in case.
- `friend DtString operator+(const DtString& str1, const DtString& str2)`
concatenation operators
- `virtual DtString operator+(const char* str) const`
- `virtual DtString& operator+=(const char* str)`
- `virtual DtString& operator+=(const DtString& str)`
- `virtual const char& operator[](int index) const`
index operator
- `virtual char& operator[](int index)`
- `virtual DtBoolean snipFront(const char ch, DtBoolean toLastCh = DtFalse)`
truncates everything before or after the character, including the character, respectively Normally, snipFront stops at the first occurrence of "ch", and snipBack stops at the last occurrence of "ch". The second argument flags reverse this. Return value is DtTrue iff "ch" was found in the string.
- `virtual DtBoolean snipBack(const char ch, DtBoolean fromFirstCh = DtFalse)`
- `DtBoolean snipFrontToLast(const char ch)`
- `DtBoolean snipBackToFirst(const char ch)`
- `virtual int count(const char ch) const`
returns the number of occurrences of the character
- `virtual int length() const`
return the length of the string
- `const char* string() const`
returns the char* internal representation of the string
- `static const DtString& nullString()`
- `void clear()`
deletes the string, and sets the myString pointer to NULL non-virtual because only called by destructor, and non-virtual members myString is set to NULL, so it should be allocated some memory if operators are to be used on the string
- `void setString(const char* str)`
Set/Get for myString - char pointer that holds the string non-virtual because only called by constructor
- `char* myString`
- `static const int theMaxInsertionStringSize`

Direct child classes:

[DtFilename](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file filename.h:

class DtFilename: public DtString

class DtFilename: Instances of DtFilename are DtStrings which convert filenames to the native format

Inheritance:

DtFilename - DtString

Public Methods

-  **DtFilename ()**
default constructor
-  **DtFilename (const char* str)**
constructor
-  virtual **~DtFilename ()**
destructor
-  **DtFilename (const DtFilename& orig)**
copy constructor
-  DtFilename& **operator= (const DtFilename& orig)**
assignment operator
-  virtual DtBoolean **absolute () const**
Returns DtTrue if the filename is an absolute filename, or a relative filename, respectively
-  virtual DtBoolean **relative () const**
-  static DtFilename **combine (const DtFilename& path, const DtFilename& filename)**
Constructs a new filename by combining the path with filename, which is an absolute or relative path
-  static char **separator ()**
returns the separator used in representing directories

Protected Fields

-  static const char **theSeparator**

Protected Methods

-  void **replaceSlashes ()**
Replaces slashes
-

Inherited from [DtString](#):

Public Methods

- virtual **operator** const char*() const
- friend [DtBoolean](#) **operator**==(const [DtString](#)& str1, const [DtString](#)& str2)
- friend [DtBoolean](#) **operator**<(const [DtString](#)& str1, const [DtString](#)& str2)
- friend [DtBoolean](#) **operator**<=(const [DtString](#)& str1, const [DtString](#)& str2)
- friend [DtBoolean](#) **operator**>(const [DtString](#)& str1, const [DtString](#)& str2)
- friend [DtBoolean](#) **operator**>=(const [DtString](#)& str1, const [DtString](#)& str2)
- friend [DtBoolean](#) **operator**!=(const [DtString](#)& str1, const [DtString](#)& str2)
- virtual [DtBoolean](#) **operator**==(const char* str) const
- virtual [DtBoolean](#) **operator**<(const char* str) const
- virtual [DtBoolean](#) **operator**<=(const char* str) const
- virtual [DtBoolean](#) **operator**>(const char* str) const
- virtual [DtBoolean](#) **operator**>=(const char* str) const
- virtual [DtBoolean](#) **operator**!=(const char* str) const
- virtual [DtBoolean](#) **caseCompare**(const [DtString](#)& str)
- friend [DtString](#) **operator**+(const [DtString](#)& str1, const [DtString](#)& str2)
- virtual [DtString](#) **operator**+(const char* str) const
- virtual [DtString](#)& **operator**+=(const char* str)
- virtual [DtString](#)& **operator**+=(const [DtString](#)& str)
- virtual const char& **operator**[](int index) const
- virtual char& **operator**[](int index)
- virtual [DtBoolean](#) **snipFront**(const char ch, [DtBoolean](#) toLastCh = DtFalse)
- virtual [DtBoolean](#) **snipBack**(const char ch, [DtBoolean](#) fromFirstCh = DtFalse)
- [DtBoolean](#) **snipFrontToLast**(const char ch)
- [DtBoolean](#) **snipBackToFirst**(const char ch)
- virtual int **count**(const char ch) const
- virtual int **length**() const
- const char* **string**() const

- static const [DtString](#)& nullString()

Protected Fields

- char* myString

- static const int theMaxInsertionStringSize

Protected Methods

- void clear()

- void setString(const char* str)

Private Methods

- friend ostream& operator<<(ostream& os, const [DtString](#)& str)

- friend istream& operator>>(istream& is, [DtString](#)& str)

Documentation

class DtFilename: Instances of DtFilename are DtStrings which convert filenames to the native format

- DtFilename()**

default constructor

- DtFilename(const char* str)**

constructor

- virtual ~DtFilename()**

destructor

- DtFilename(const [DtFilename](#)& orig)**

copy constructor

- [DtFilename](#)& operator=(const [DtFilename](#)& orig)**

assignment operator

- virtual [DtBoolean](#) absolute() const**

Returns DtTrue if the filename is an absolute filename, or a relative filename, respectively

- virtual [DtBoolean](#) relative() const**

- static [DtFilename](#) combine(const [DtFilename](#)& path, const [DtFilename](#)& filename)**

Constructs a new filename by combining the path with filename, which is an absolute or relative path. If the filename is an absolute filename, then filename is returned. If the filename is a relative filename, then path is prepended to it.

- `static char separator()`
returns the separator used in representing directories

- `void replaceSlashes()`
Replaces slashes. On unix, this replaces backslashes with forward slashes. Vice versa on win32. Hail Mr Gates.
This is not virtual because we call this from the constructors and destructor.

- `static const char theSeparator`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file syshashkey.h:

class [DtSystemHashKey](#): public [DtBaseHashKey](#)

class DtSystemHashKey: Instances of DtSystemHashKey are used as hash keys for an emitter system

Inheritance:

DtSystemHashKey - [DtBaseHashKey](#)

Public Methods

- **DtSystemHashKey** (const DtGlobalObjectDesignator& id, int systemId)
default constructor
- virtual **~DtSystemHashKey** ()
destructor
- **DtSystemHashKey** (const [DtSystemHashKey](#)& orig)
copy constructor
- [DtSystemHashKey](#)& **operator=** (const [DtSystemHashKey](#)& orig)
assignment operator
- virtual [DtBaseHashKey](#)* **clone** () const
make a copy of this hash key

Protected Fields

- [DtObjectIdHashKey](#) myDesHashKey
- [DtString](#) myStr
- int mySystemId

Protected Methods

- virtual const [DtString](#)& **stringRep** () const
get the string representation of this key
 - virtual DtU32 **computeHashValue** ()
compute the hashed value for this key
-

Inherited from [DtBaseHashKey](#):

Public Methods

- virtual [DtBoolean](#) **operator==(const [DtBaseHashKey](#)& other)** const
- virtual [DtBoolean](#) **operator!=(const [DtBaseHashKey](#)& other)** const
- virtual DtU32 **hashValue()** const

Protected Fields

- [DtBoolean](#) myHashValuelsComputed
 - DtU32 myHashValue
-

Documentation

class DtSystemHashKey: Instances of DtSystemHashKey are used as hash keys for an emitter system

- DtSystemHashKey(const DtGlobalObjectDesignator& id, int systemId)**
default constructor

- virtual ~DtSystemHashKey()**
destructor

- DtSystemHashKey(const [DtSystemHashKey](#)& orig)**
copy constructor

- [DtSystemHashKey](#)& operator=(const [DtSystemHashKey](#)& orig)**
assignment operator

- virtual [DtBaseHashKey](#)* clone() const**
make a copy of this hash key

- virtual const [DtString](#)& stringRep() const**
get the string representation of this key

- virtual DtU32 computeHashValue()**
compute the hashed value for this key

- [DtObjectIdHashKey](#) myDesHashKey**

- [DtString](#) myStr**

- int mySystemId**
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file objidhashkey.h:

class DtObjectIdHashKey: public DtBaseHashKey

class DtObjectIdHashKey: Instances of DtObjectIdHashKey are hash keys made from DtObjectIds

Inheritance:

DtObjectIdHashKey - [DtBaseHashKey](#)

Public Methods

- **DtObjectIdHashKey** (const DtObjectId& id)
default constructor
- virtual **~DtObjectIdHashKey** ()
destructor
- **DtObjectIdHashKey** (const [DtObjectIdHashKey](#)& orig)
copy constructor
- [DtObjectIdHashKey](#)& **operator=** (const [DtObjectIdHashKey](#)& orig)
assignment operator
- virtual [DtBaseHashKey](#)* **clone** () const
make a copy of this hash key

Protected Fields

- DtObjectId **myId**
- [DtString](#) **myStr**

Protected Methods

- virtual const [DtString](#)& **stringRep** () const
get the string representation of this key
 - virtual DtU32 **computeHashValue** ()
compute the hashed value for this key
-

Inherited from [DtBaseHashKey](#):

Public Methods

- virtual [DtBoolean](#) **operator==(const [DtBaseHashKey](#)& other)** const
- virtual [DtBoolean](#) **operator!=(const [DtBaseHashKey](#)& other)** const
- virtual DtU32 **hashValue()** const

Protected Fields

- [DtBoolean](#) myHashValuelsComputed
 - DtU32 myHashValue
-

Documentation

class DtObjectIdHashKey:

Instances of DtObjectIdHashKey are hash keys made from DtObjectIds. They can be used with DtBaseHashList to store arbitrary pieces of data hashed by DtObjectId.

- DtObjectIdHashKey(const DtObjectId& id)**
default constructor
 - virtual ~DtObjectIdHashKey()**
destructor
 - DtObjectIdHashKey(const [DtObjectIdHashKey](#)& orig)**
copy constructor
 - [DtObjectIdHashKey](#)& operator=(const [DtObjectIdHashKey](#)& orig)**
assignment operator
 - virtual [DtBaseHashKey](#)* clone() const**
make a copy of this hash key
 - virtual const [DtString](#)& stringRep() const**
get the string representation of this key
 - virtual DtU32 computeHashValue()**
compute the hashed value for this key
 - DtObjectId myId
 - [DtString](#) myStr
-

This class has no child classes.

In file strhashkey.h:

class DtStringHashKey: public DtBaseHashKey

class DtStringHashKey: Instances of DtStringHashKey are used as a hash key made from a string

Inheritance:

DtStringHashKey - DtBaseHashKey

Public Methods

- **DtStringHashKey** (const [DtString](#)& str)
default constructor
- **DtStringHashKey** (const char* str)
alternate constructor
- virtual **~DtStringHashKey** ()
destructor
- **DtStringHashKey** (const [DtStringHashKey](#)& orig)
copy constructor
- [DtStringHashKey](#)& **operator=** (const [DtStringHashKey](#)& orig)
assignment operator
- virtual [DtBaseHashKey](#)* **clone** () const
make a copy of this hash key

Protected Fields

- [DtString](#) myString

Protected Methods

- virtual const [DtString](#)& **stringRep** () const
get the string representation of this key
 - virtual DtU32 **computeHashValue** ()
compute the hashed value for this key
-

Inherited from [DtBaseHashKey](#):

Public Methods

- virtual [DtBoolean](#) operator==(const [DtBaseHashKey](#)& other) const
- virtual [DtBoolean](#) operator!=(const [DtBaseHashKey](#)& other) const
- virtual DtU32 hashValue() const

Protected Fields

- [DtBoolean](#) myHashValuelsComputed
 - DtU32 myHashValue
-

Documentation

class DtStringHashKey: Instances of DtStringHashKey are used as a hash key made from a string

- [DtStringHashKey](#)(const [DtString](#)& str)
default constructor
 - [DtStringHashKey](#)(const char* str)
alternate constructor
 - virtual ~[DtStringHashKey](#)()
destructor
 - [DtStringHashKey](#)(const [DtStringHashKey](#)& orig)
copy constructor
 - [DtStringHashKey](#)& operator=(const [DtStringHashKey](#)& orig)
assignment operator
 - virtual [DtBaseHashKey](#)* clone() const
make a copy of this hash key
 - virtual const [DtString](#)& stringRep() const
get the string representation of this key
 - virtual DtU32 computeHashValue()
compute the hashed value for this key
 - [DtString](#) myString
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file inthashkey.h:

class DtIntHashKey: public DtBaseHashKey

class DtIntHashKey: Instances of DtIntHashKey are used as a hash key made from an integer

Inheritance:

DtIntHashKey - DtBaseHashKey

Public Methods

- **DtIntHashKey** (unsigned long num)
default constructor
- virtual **~DtIntHashKey** ()
destructor
- **DtIntHashKey** (const [DtIntHashKey](#)& orig)
copy constructor
- [DtIntHashKey](#)& **operator=** (const [DtIntHashKey](#)& orig)
assignment operator
- virtual [DtBaseHashKey](#)* **clone** () const
make a copy of this hash key

Protected Fields

- unsigned long **myNum**
- [DtString](#) **myStr**

Protected Methods

- virtual const [DtString](#)& **stringRep** () const
get the string representation of this key
 - virtual DtU32 **computeHashValue** ()
compute the hashed value for this key
-

Inherited from [DtBaseHashKey](#):

Public Methods

- virtual [DtBoolean](#) operator==(const [DtBaseHashKey](#)& other) const
- virtual [DtBoolean](#) operator!=(const [DtBaseHashKey](#)& other) const
- virtual DtU32 hashValue() const

Protected Fields

- [DtBoolean](#) myHashValuelsComputed
 - DtU32 myHashValue
-

Documentation

class DtIntHashKey: Instances of DtIntHashKey are used as a hash key made from an integer

- DtIntHashKey(unsigned long num)
default constructor
 - virtual ~DtIntHashKey()
destructor
 - DtIntHashKey(const [DtIntHashKey](#)& orig)
copy constructor
 - [DtIntHashKey](#)& operator=(const [DtIntHashKey](#)& orig)
assignment operator
 - virtual [DtBaseHashKey](#)* clone() const
make a copy of this hash key
 - virtual const [DtString](#)& stringRep() const
get the string representation of this key
 - virtual DtU32 computeHashValue()
compute the hashed value for this key
 - unsigned long myNum
 - [DtString](#) myStr
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file aggregatesr.h:

class DtAggregateStateRepository: public DtBaseEntityStateRepository

class DtAggregateStateRepository: Instances of DtAggregateStateRepository are used to maintain the state of an aggregate

Inheritance:

DtAggregateStateRepository - DtBaseEntityStateRepository - DtStateRepository

Public Methods

	DtAggregateStateRepository () <i>default constructor</i>
 virtual	~DtAggregateStateRepository () <i>destructor</i>
	DtAggregateStateRepository (const DtAggregateStateRepository & orig) <i>copy constructor</i>
 DtAggregateStateRepository &	operator= (const DtAggregateStateRepository & orig) <i>assignment operator</i>
 virtual DtStateRepository *	clone () const <i>return an empty object of the same type as this</i>
 virtual void	setAggregateID (const DtEntityIdentifier & id) <i>Set/Get the aggregate's identifier</i>
 virtual const DtEntityIdentifier &	aggregateID () const
 virtual void	setForceId (DtForceType force) <i>Set/Get the force type</i>
 virtual DtForceType	forceId () const
 virtual void	setAggregateState (DtAggregateState value) <i>Set/Get the aggregate state</i>
 virtual DtAggregateState	aggregateState () const
 virtual void	setAggregateType (const DtEntityType & type) <i>Set/Get the aggregate type</i>
 virtual const DtEntityType &	aggregateType () const
 virtual void	setFormation (DtAggregateFormation value) <i>Set/Get the formation</i>

virtual DtAggregateFormation	formation () const
virtual void	setMarkingCharacterSet (DtCharacterSet value) <i>Set/Get marking character set</i>
virtual DtCharacterSet	markingCharacterSet () const
virtual void	setMarkingText (DtU8* copyFrom) <i>Set/Get marking text (copies 31 bytes)</i>
virtual void	getMarkingText (DtU8* copyTo) const
virtual const char*	markingText () const
virtual void	setDimensions (DtConstVector dims) <i>Set/Get the dimensions</i>
virtual DtConstVector	dimensions () const
const DtGlobalObjectDesignatorList &	entities () const <i>Return the list of published entities in the aggregate</i>
DtGlobalObjectDesignatorList *	entities ()
DtGlobalObjectDesignatorList *	subAggregates () <i>Return the list of published subaggregates in the aggregate</i>
const DtGlobalObjectDesignatorList &	subAggregates () const
virtual DtBoolean	setNumberOfDIS_Subaggregates (int newNumberOfAggregates) <i>Set/Get the number of DIS sub-aggregates</i>
virtual int	numberOfDIS_Subaggregates () const
virtual DtBoolean	setNumberOfDIS_Entities (int newNumberOfEntities) <i>Set/Get the number of DIS entities</i>
virtual int	numberOfDIS_Entities () const
virtual DtBoolean	setNumberOfSilentSubaggregates (int newNumberOfAggregates) <i>Set/Get the number of silent aggregates</i>
virtual int	numberOfSilentSubaggregates () const
virtual DtBoolean	setNumberOfSilentEntities (int newNumberOfEntities) <i>Set/Get the number of silent entities</i>
virtual int	numberOfSilentEntities () const
virtual DtBoolean	setSubaggregateID (int index, const DtGlobalObjectDesignator& agglID) <i>Set/Get a DIS sub-aggregate ID</i>
virtual DtBoolean	getSubaggregateID (int index, DtGlobalObjectDesignator& agglID) const
virtual DtBoolean	setEntityID (int index, const DtGlobalObjectDesignator& entID) <i>Set/Get a DIS entity ID</i>
virtual DtBoolean	getEntityID (int index, DtGlobalObjectDesignator& entID) const
virtual DtBoolean	setSilentAggregateList (int index, const DtSilentAggregateList & agglList) <i>Set/Get a silent aggregate list</i>
virtual DtBoolean	getSilentAggregateList (int index, DtSilentAggregateList & agglList) const
virtual DtBoolean	setSilentEntityList (int index, const DtSilentEntityList & entList) <i>Set/Get a silent entity list</i>
virtual DtBoolean	getSilentEntityList (int index, DtSilentEntityList & entList) const
virtual DtBoolean	setNumberOfDatumRecords (int newNumberOfRecords) <i>Set the number of variable datum records</i>

virtual int	numberOfDatumRecords () const <i>Get the number of variable datum records</i>
virtual DtBoolean	setVariableDatumRecord (int index, const DtVariableDatumRecord & rec) <i>Set/Get a variable datum record</i>
virtual DtBoolean	getVariableDatumRecord (int index, DtVariableDatumRecord & rec) const
virtual DtVariableDatumRecord &	variableDatumRecordRef (int index) <i>Get a reference to a variable datum record</i>
virtual const DtVariableDatumRecord &	variableDatumRecordConstRef (int index) const <i>Get a constant reference to a variable datum record</i>
virtual void	printData () const <i>Print the state data</i>
static DtAggregateStateRepository *	create ()

Protected Fields

DtForceType	myForceID
DtAggregateState	myAggregateState
DtAggregateFormation	myFormation
DtCharacterSet	myMarkingCharSet
char	myMarkingText [AGGREGATE_SR_MARKING_TEXT_LENGTH]
DtVector	myDimensions
DtGlobalObjectDesignatorList	myEntities
DtGlobalObjectDesignatorList	mySubAggs
int	myNumberOfSilentAggregateTypes
DtSilentAggregateList *	mySilentAggregates
int	myNumberOfSilentEntityTypes
DtSilentEntityList *	mySilentEntities
int	myNumberOfVariableDatumRecords
DtVariableDatumRecord *	myVariableDatumRecords
DtVariableDatumRecord	myDumRec

Protected Methods

DtAggregateStateRepository *	self () const <i>Cast off const-ness for trivial internal change</i>
--	--

Protected

```
#define AGGREGATE_SR_MARKING_TEXT_LENGTH  
Aggregate state PDU marking is 31 bytes long, RPR marking struct is 11  
#define AGGREGATE_SR_MARKING_TEXT_LENGTH
```

Inherited from [DtBaseEntityStateRepository](#):

Public Methods

```
void setTimeStampType(DtTimeStampType type)  
DtTimeStampType timeStampType() const  
virtual void setLocation(DtConstVector location, DtTime timeNow = DtGetSimTime())  
virtual DtConstVector location(DtTime timeNow = DtGetSimTime()) const  
virtual DtConstVector lastSetLocation() const  
virtual void setVelocity(DtConstVector Velocity, DtTime timeNow = DtGetSimTime())  
virtual DtConstVector velocity( DtTime timeNow = DtGetSimTime()) const  
virtual DtConstVector lastSetVelocity() const  
virtual void setAcceleration(DtConstVector Acceleration, DtTime timeNow = DtGetSimTime())  
virtual DtConstVector acceleration( DtTime timeNow = DtGetSimTime()) const  
virtual DtConstVector lastSetAcceleration() const  
virtual void setOrientation(const DtTaitBryan& orient, DtTime timeNow = DtGetSimTime())  
virtual const DtTaitBryan& orientation( DtTime timeNow = DtGetSimTime()) const  
virtual const DtTaitBryan& lastSetOrientation() const  
virtual void setRotationalVelocity(DtConstVector BodyRates, DtTime timeNow = DtGetSimTime())  
virtual DtConstVector rotationalVelocity( DtTime timeNow = DtGetSimTime()) const  
virtual DtConstVector lastSetRotationalVelocity() const  
virtual DtConstDcm bodyToGeoc( DtTime timeNow = DtGetSimTime()) const  
virtual void setAlgorithm(DtDeadReckonTypes alg)  
virtual DtDeadReckonTypes algorithm() const  
virtual void setEntityId(const DtEntityIdentifier& id)  
virtual const DtEntityIdentifier& entityId() const  
virtual void setEntityType(const DtEntityType& type)
```

- virtual const [DtEntityType](#)& **entityType**() const
- virtual void **setFrozen**([DtBoolean](#) onOff)
- virtual [DtBoolean](#) **frozen**() const
- virtual void **setApproximator**([DtApproximator](#)* approx)
- virtual [DtApproximator](#)* **approximator**() const
- virtual void **useDeadReckoner**([DtDeadReckoner](#)* dr)
- virtual void **useDeadReckoner**()
- virtual [DtDeadReckoner](#)* **usingDeadReckoner**()
- virtual void **useSmoother**([DtSmoother](#)* sm)
- virtual void **useSmoother**()
- virtual [DtSmoother](#)* **usingSmoother**()
- virtual [DtThresholder](#)* **thresholder**()
- virtual const [DtThresholder](#)& **thresholder**() const
- virtual void **setTimeForSets**(DtTime time)
- inline const [DtEntityIdentifier](#)& **entityId**() const
- inline const [DtEntityType](#)& **entityType**() const
- inline DtConstVector **lastSetAcceleration**() const
- inline DtConstVector **lastSetLocation**() const
- inline const [DtTaitBryan](#)& **lastSetOrientation**() const
- inline DtConstVector **lastSetRotationalVelocity**() const
- inline DtConstVector **lastSetVelocity**() const
- inline void **setEntityId**(const [DtEntityIdentifier](#)& id)
- inline void **setEntityType**(const [DtEntityType](#)& type)
- inline void **setTimeStampType**([DtTimeStampType](#) type)
- inline [DtThresholder](#)* **thresholder**()
- inline const [DtThresholder](#)& **thresholder**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- [DtApproximator](#)* myApproximator
- [DtDeadReckoner](#)* myDeadReckoner
- [DtSmoother](#)* mySmoother
- [DtDeadReckoner](#)* myDfltDeadReckoner
- [DtSmoother](#)* myDfltSmoother
- [DtVector](#) myLocation
- [DtVector](#) myVelocity
- [DtVector](#) myAcceleration
- [DtTaitBryan](#) myOrientation
- [DtVector](#) myRotationalVelocity
- [DtDeadReckonTypes](#) myAlgorithm
- [DtDcm](#) myBodyToGeoc
- [DtBoolean](#) myBodyToGeocDirty
- [DtTimeStampType](#) myTimeStampType
- [DtEntityIdentifier](#) myEntityId
- [DtEntityType](#) myEntityType
- [DtBoolean](#) myFrozen
- [DtThresholder](#) myThresholder
- DtTime myTimeForSets

Protected Methods

- virtual DtTime **timeForSets**() const
-

Inherited from [DtStateRepository](#):

Protected



Documentation

class DtAggregateStateRepository:

Instances of DtAggregateStateRepository are used to maintain the state of an aggregate. It provides an interface which covers the various features which define the state of an aggregate.

- `DtAggregateStateRepository()`
default constructor
- `virtual ~DtAggregateStateRepository()`
destructor
- `DtAggregateStateRepository(const DtAggregateStateRepository& orig)`
copy constructor
- `DtAggregateStateRepository& operator=( const DtAggregateStateRepository& orig)`
assignment operator
- `virtual DtStateRepository* clone() const`
return an empty object of the same type as this
- `virtual void setAggregateID(const DtEntityIdentifier& id)`
Set/Get the aggregate's identifier. Calls down to base class.
- `virtual const DtEntityIdentifier& aggregateID() const`
- `virtual void setForceId(DtForceType force)`
Set/Get the force type
- `virtual DtForceType forceId() const`
- `virtual void setAggregateState(DtAggregateState value)`
Set/Get the aggregate state
- `virtual DtAggregateState aggregateState() const`
- `virtual void setAggregateType(const DtEntityType& type)`
Set/Get the aggregate type. Calls down to base class.
- `virtual const DtEntityType& aggregateType() const`
- `virtual void setFormation(DtAggregateFormation value)`
Set/Get the formation
- `virtual DtAggregateFormation formation() const`
- `virtual void setMarkingCharacterSet(DtCharacterSet value)`
Set/Get marking character set
- `virtual DtCharacterSet markingCharacterSet() const`
- `virtual void setMarkingText(DtU8* copyFrom)`
Set/Get marking text (copies 31 bytes)

- `virtual void getMarkingText(DtU8* copyTo) const`
- `virtual const char* markingText() const`
- `virtual void setDimensions(DtConstVector dims)`
Set/Get the dimensions
- `virtual DtConstVector dimensions() const`
- `const DtGlobalObjectDesignatorList& entities() const`
Return the list of published entities in the aggregate
- `DtGlobalObjectDesignatorList* entities()`
- `DtGlobalObjectDesignatorList* subAggregates()`
Return the list of published subaggregates in the aggregate
- `const DtGlobalObjectDesignatorList& subAggregates() const`
- `virtual DtBoolean setNumberOfDIS_Subaggregates(int newNumberOfAggregates)`
Set/Get the number of DIS sub-aggregates. Set returns DtFalse if the requested number is too large. Changing this number will erase all previous data. Can be done through subAggregates() instead.
- `virtual int numberOfDIS_Subaggregates() const`
- `virtual DtBoolean setNumberOfDIS_Entities(int newNumberOfEntities)`
Set/Get the number of DIS entities. Set returns DtFalse if the requested number is too large. Changing this number will erase all previous data. Can be done through entities() instead.
- `virtual int numberOfDIS_Entities() const`
- `virtual DtBoolean setNumberOfSilentSubaggregates( int newNumberOfAggregates)`
Set/Get the number of silent aggregates. Set returns DtFalse if the requested number is too large. Changing this number will erase all previous data.
- `virtual int numberOfSilentSubaggregates() const`
- `virtual DtBoolean setNumberOfSilentEntities(int newNumberOfEntities)`
Set/Get the number of silent entities. Set returns DtFalse if the requested number is too large. Changing this number will erase all previous data.
- `virtual int numberOfSilentEntities() const`
- `virtual DtBoolean setSubaggregateID(int index, const DtGlobalObjectDesignator& aggID)`
Set/Get a DIS sub-aggregate ID. Returns DtFalse if the index is invalid. Can be done through subAggregates() instead.
- `virtual DtBoolean getSubaggregateID(int index, DtGlobalObjectDesignator& aggID) const`
- `virtual DtBoolean setEntityID(int index, const DtGlobalObjectDesignator& entID)`
Set/Get a DIS entity ID. Returns DtFalse if the index is invalid. Can be done through entities() instead.
- `virtual DtBoolean getEntityID(int index, DtGlobalObjectDesignator& entID) const`
- `virtual DtBoolean setSilentAggregateList(int index, const DtSilentAggregateList& aggList)`

Set/Get a silent aggregate list. Returns DtFalse if the index is invalid.

virtual [DtBoolean](#) getSilentAggregateList(int index, [DtSilentAggregateList](#)& aggList) const

virtual [DtBoolean](#) setSilentEntityList(int index, const [DtSilentEntityList](#)& entList)
Set/Get a silent entity list. Returns DtFalse if the index is invalid.

virtual [DtBoolean](#) getSilentEntityList(int index, [DtSilentEntityList](#)& entList) const

virtual [DtBoolean](#) setNumberOfDatumRecords(int newNumberOfRecords)

Set the number of variable datum records. Returns DtFalse if the number of records is too large. This will erase the contents of existing datum records.

virtual int numberOfDatumRecords() const
Get the number of variable datum records

virtual [DtBoolean](#) setVariableDatumRecord(int index, const [DtVariableDatumRecord](#)& rec)
Set/Get a variable datum record. Returns DtFalse if the index is invalid.

virtual [DtBoolean](#) getVariableDatumRecord(int index, [DtVariableDatumRecord](#)& rec) const

virtual [DtVariableDatumRecord](#)& variableDatumRecordRef(int index)
Get a reference to a variable datum record

virtual const [DtVariableDatumRecord](#)& variableDatumRecordConstRef(int index) const
Get a constant reference to a variable datum record

virtual void printData() const
Print the state data

static [DtAggregateStateRepository](#)* create()

[DtAggregateStateRepository](#)* self() const
Cast off const-ness for trivial internal change. Non-virtual since can't override on return type.

[DtForceType](#) myForceID

[DtAggregateState](#) myAggregateState

[DtAggregateFormation](#) myFormation

[DtCharacterSet](#) myMarkingCharSet

#define AGGREGATE_SR_MARKING_TEXT_LENGTH
Aggregate state PDU marking is 31 bytes long, RPR marking struct is 11

#define AGGREGATE_SR_MARKING_TEXT_LENGTH

char myMarkingText[AGGREGATE_SR_MARKING_TEXT_LENGTH]

[DtVector](#) myDimensions

[DtGlobalObjectDesignatorList](#) myEntities

[DtGlobalObjectDesignatorList](#) mySubAggs

- `int myNumberOfSilentAggregateTypes`
- `DtSilentAggregateList* mySilentAggregates`
- `int myNumberOfSilentEntityTypes`
- `DtSilentEntityList* mySilentEntities`
- `int myNumberOfVariableDatumRecords`
- `DtVariableDatumRecord* myVariableDatumRecords`
- `DtVariableDatumRecord myDumRec`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file baseentitysr.h:

class DtBaseEntityStateRepository: public DtStateRepository

class DtBaseEntityStateRepository: Instances of DtBaseEntityStateRepository are used to maintain basic state information needed by all kinds of entities

Inheritance:

DtBaseEntityStateRepository - DtStateRepository

Public Methods

	DtBaseEntityStateRepository () <i>default constructor</i>
virtual	~DtBaseEntityStateRepository () <i>destructor</i>
	DtBaseEntityStateRepository (const DtBaseEntityStateRepository & orig) <i>copy constructor</i>
DtBaseEntityStateRepository &	operator= (const DtBaseEntityStateRepository & orig) <i>assignment operator</i>
virtual DtStateRepository *	clone () const <i>return an empty object of the same type as this</i>
void	setTimeStampType (DtTimeStampType type) <i>Set/Get flag indicating whether we're using absolute timestamps for dead reckoning</i>
DtTimeStampType	timeStampType () const
virtual void	setLocation (DtConstVector location, DtTime timeNow = DtGetSimTime()) <i>Set/Get the location of the center of mass in world coordinates</i>
virtual DtConstVector	location (DtTime timeNow = DtGetSimTime()) const
virtual DtConstVector	lastSetLocation () const <i>Return the value last passed to setLocation, regardless of whether we have an approximator</i>
virtual void	setVelocity (DtConstVector Velocity, DtTime timeNow = DtGetSimTime()) <i>Set/Get the velocity</i>
virtual DtConstVector	velocity (DtTime timeNow = DtGetSimTime()) const
virtual DtConstVector	lastSetVelocity () const <i>Return the value last passed to setVelocity, regardless of whether we have an approximator</i>
virtual void	setAcceleration (DtConstVector Acceleration, DtTime timeNow = DtGetSimTime()) <i>Set/Get the acceleration</i>

virtual DtConstVector	acceleration (DtTime timeNow = DtGetSimTime()) const
virtual DtConstVector	lastSetAcceleration () const <i>Return the value last passed to setAcceleration, regardless of whether we have an approximator</i>
virtual void	setOrientation (const DtTaitBryan & orient, DtTime timeNow = DtGetSimTime()) <i>Set/Get the orientation</i>
virtual const DtTaitBryan &	orientation (DtTime timeNow = DtGetSimTime()) const
virtual const DtTaitBryan &	lastSetOrientation () const <i>Return the value last passed to setOrientation, regardless of whether we have an approximator</i>
virtual void	setRotationalVelocity (DtConstVector BodyRates, DtTime timeNow = DtGetSimTime()) <i>Set/Get the rotational velocity</i>
virtual DtConstVector	rotationalVelocity (DtTime timeNow = DtGetSimTime()) const
virtual DtConstVector	lastSetRotationalVelocity () const <i>Return the value last passed to setRotationalVelocity, regardless of whether we have an approximator</i>
virtual DtConstDcm	bodyToGeoc (DtTime timeNow = DtGetSimTime()) const <i>Get the body to geocentric coordinate matrix</i>
virtual void	setAlgorithm (DtDeadReckonTypes alg) <i>Set/Get the dead reckoning algorithm</i>
virtual DtDeadReckonTypes	algorithm () const
virtual void	setEntityId (const DtEntityIdentifier & id) <i>Set/Get the entity's identifier</i>
virtual const DtEntityIdentifier &	entityId () const
virtual void	setEntityType (const DtEntityType & type) <i>Set/Get the entity type</i>
virtual const DtEntityType &	entityType () const
virtual void	setFrozen (DtBoolean onOff) <i>Set/Get whether frozen or not</i>
virtual DtBoolean	frozen () const
virtual void	setApproximator (DtApproximator * approx) <i>Get and set the approximator used to provide approximated position and orientation information for the entity</i>
virtual DtApproximator *	approximator () const
virtual void	useDeadReckoner (DtDeadReckoner * dr) <i>Tells the ESR to use dr as its approximator, and to notify it when DR algorithm changes</i>
virtual void	useDeadReckoner () <i>Create and use a new (private) dead-reckoner as our approximator</i>
virtual DtDeadReckoner *	usingDeadReckoner () <i>If the ESR is using a dead reckoner, returns it</i>
virtual void	useSmoother (DtSmoother * sm) <i>Tells the ESR to use sm as its approximator, and to notify it when DR algorithm changes</i>
virtual void	useSmoother () <i>Create and use a new (private) smoother as our approximator</i>

virtual DtSmoother *	usingSmoother () <i>If the ESR is using a smoother, returns it</i>
virtual DtThresholder *	thresholder () <i>Returns the SR's thresholder, which contains thresholds needed for encoding</i>
virtual const DtThresholder &	thresholder () const <i>Const version</i>
virtual void	printData () const <i>Print the state data</i>
virtual void	setTimeForSets (DtTime time)
inline const DtEntityIdentifier &	entityId () const
inline const DtEntityType &	entityType () const
inline DtConstVector	lastSetAcceleration () const
inline DtConstVector	lastSetLocation () const
inline const DtTaitBryan &	lastSetOrientation () const
inline DtConstVector	lastSetRotationalVelocity () const
inline DtConstVector	lastSetVelocity () const
inline void	setEntityId (const DtEntityIdentifier & id)
inline void	setEntityType (const DtEntityType & type)
inline void	setTimeStampType (DtTimeStampType type)
inline DtThresholder *	thresholder ()
inline const DtThresholder &	thresholder () const
inline DtTimeStampType	timeStampType () const

Protected Fields

DtApproximator *	myApproximator <i>holds what's currently being used as our approximator</i>
DtDeadReckoner *	myDeadReckoner <i>holds what was most recently passed to useDeadReckoner</i>
DtSmoother *	mySmoother <i>holds what was most recently passed to useSmoother</i>
DtDeadReckoner *	myDfltDeadReckoner <i>Created and deleted by this ESR if needed</i>
DtSmoother *	myDfltSmoother
DtVector	myLocation
DtVector	myVelocity
DtVector	myAcceleration
DtTaitBryan	myOrientation
DtVector	myRotationalVelocity
DtDeadReckonTypes	myAlgorithm

DtDcm	myBodyToGeoc
DtBoolean	myBodyToGeocDirty
DtTimeStampType	myTimeStampType
DtEntityIdentifier	myEntityId
DtEntityType	myEntityType
DtBoolean	myFrozen
DtThresholder	myThresholder
DtTime	myTimeForSets

Protected Methods

DtBaseEntityStateRepository *	self () const <i>Cast off const-ness for trivial internal change</i>
virtual DtTime	timeForSets () const

Inherited from [DtStateRepository](#):

Protected



Documentation

class DtBaseEntityStateRepository: Instances of DtBaseEntityStateRepository are used to maintain basic state information needed by all kinds of entities

- DtBaseEntityStateRepository()**
default constructor
- virtual ~DtBaseEntityStateRepository()**
destructor
- DtBaseEntityStateRepository(const [DtBaseEntityStateRepository](#)& orig)**
copy constructor
- [DtBaseEntityStateRepository](#)& operator=(const [DtBaseEntityStateRepository](#)& orig)**
assignment operator
- virtual [DtStateRepository](#)* clone() const**
return an empty object of the same type as this
- void setTimeStampType([DtTimeStampType](#) type)**

Set/Get flag indicating whether we're using absolute timestamps for dead reckoning. This flag will be set to DtAbsolute by the decoders only if incoming state messages indicate that the sender is using absolute time stamps, AND our local exercise connection indicates that we are using absolute timestamping.

• [DtTimeStampType](#) timeStampType() const

• virtual void setLocation(DtConstVector location, DtTime timeNow = DtGetSimTime())
Set/Get the location of the center of mass in world coordinates

• virtual DtConstVector location(DtTime timeNow = DtGetSimTime()) const

• virtual DtConstVector lastSetLocation() const
Return the value last passed to setLocation, regardless of whether we have an approximator

• virtual void setVelocity(DtConstVector Velocity, DtTime timeNow = DtGetSimTime())
Set/Get the velocity

• virtual DtConstVector velocity(DtTime timeNow = DtGetSimTime()) const

• virtual DtConstVector lastSetVelocity() const
Return the value last passed to setVelocity, regardless of whether we have an approximator

• virtual void setAcceleration(DtConstVector Acceleration, DtTime timeNow = DtGetSimTime())
Set/Get the acceleration

• virtual DtConstVector acceleration(DtTime timeNow = DtGetSimTime()) const

• virtual DtConstVector lastSetAcceleration() const
Return the value last passed to setAcceleration, regardless of whether we have an approximator

• virtual void setOrientation(const [DtTaitBryan](#)& orient, DtTime timeNow = DtGetSimTime())
Set/Get the orientation

• virtual const [DtTaitBryan](#)& orientation(DtTime timeNow = DtGetSimTime()) const

• virtual const [DtTaitBryan](#)& lastSetOrientation() const
Return the value last passed to setOrientation, regardless of whether we have an approximator

• virtual void setRotationalVelocity(DtConstVector BodyRates, DtTime timeNow = DtGetSimTime())
Set/Get the rotational velocity

• virtual DtConstVector rotationalVelocity(DtTime timeNow = DtGetSimTime()) const

• virtual DtConstVector lastSetRotationalVelocity() const
Return the value last passed to setRotationalVelocity, regardless of whether we have an approximator

• virtual DtConstDcm bodyToGeoc(DtTime timeNow = DtGetSimTime()) const
Get the body to geocentric coordinate matrix

• virtual void setAlgorithm([DtDeadReckonTypes](#) alg)
Set/Get the dead reckoning algorithm

• virtual [DtDeadReckonTypes](#) algorithm() const

- virtual void setEntityId(const [DtEntityIdentifier](#)& id)
 Set/Get the entity's identifier
- virtual const [DtEntityIdentifier](#)& entityId() const
- virtual void setEntityType(const [DtEntityType](#)& type)
 Set/Get the entity type
- virtual const [DtEntityType](#)& entityType() const
- virtual void setFrozen([DtBoolean](#) onOff)
 Set/Get whether frozen or not
- virtual [DtBoolean](#) frozen() const
- virtual void setApproximator([DtApproximator](#)* approx)
 Get and set the approximator used to provide approximated position and orientation information for the entity. If your approximator is a DtDeadReckoner, use useDeadReckoner instead of setApproximator. If it's a smoother, use useSmoother. Caller of setApproximator has the responsibility to delete approx after the StateRep no longer needs it.
- virtual [DtApproximator](#)* approximator() const
- virtual void useDeadReckoner([DtDeadReckoner](#)* dr)
 Tells the ESR to use dr as its approximator, and to notify it when DR algorithm changes. Caller is responsible for deleting dr, but only after the StateRep no longer needs it.
- virtual void useDeadReckoner()
 Create and use a new (private) dead-reckoner as our approximator
- virtual [DtDeadReckoner](#)* usingDeadReckoner()
 If the ESR is using a dead reckoner, returns it. Else returns NULL.
- virtual void useSmoother([DtSmoother](#)* sm)
 Tells the ESR to use sm as its approximator, and to notify it when DR algorithm changes. Caller is responsible for deleting sm, but only after the StateRep no longer needs it.
- virtual void useSmoother()
 Create and use a new (private) smoother as our approximator
- virtual [DtSmoother](#)* usingSmoother()
 If the ESR is using a smoother, returns it. Else returns NULL.
- virtual [DtThresholder](#)* thresholder()
 Returns the SR's thresholder, which contains thresholds needed for encoding
- virtual const [DtThresholder](#)& thresholder() const
 Const version
- virtual void printData() const
 Print the state data
- [DtBaseEntityStateRepository](#)* self() const
 Cast off const-ness for trivial internal change. Non-virtual since can't override on return type.

[DtApproximator](#)* myApproximator

holds what's currently being used as our approximator

[DtDeadReckoner](#)* myDeadReckoner

holds what was most recently passed to useDeadReckoner

[DtSmoother](#)* mySmoother

holds what was most recently passed to useSmoother

[DtDeadReckoner](#)* myDfltDeadReckoner

Created and deleted by this ESR if needed

[DtSmoother](#)* myDfltSmoother

[DtVector](#) myLocation

[DtVector](#) myVelocity

[DtVector](#) myAcceleration

[DtTaitBryan](#) myOrientation

[DtVector](#) myRotationalVelocity

[DtDeadReckonTypes](#) myAlgorithm

[DtDcm](#) myBodyToGeoc

[DtBoolean](#) myBodyToGeocDirty

[DtTimeStampType](#) myTimeStampType

[DtEntityIdentifier](#) myEntityId

[DtEntityType](#) myEntityType

[DtBoolean](#) myFrozen

[DtThresholder](#) myThresholder

virtual void setTimeForSets(DtTime time)

virtual DtTime timeForSets() const

DtTime myTimeForSets

inline const [DtEntityIdentifier](#)& entityId() const

inline const [DtEntityType](#)& entityType() const

inline DtConstVector lastSetAcceleration() const

inline DtConstVector lastSetLocation() const

inline const [DtTaitBryan](#)& lastSetOrientation() const

inline DtConstVector lastSetRotationalVelocity() const

- inline DtConstVector lastSetVelocity() const
- inline void setEntityId(const [DtEntityIdentifier](#)& id)
- inline void setEntityType(const [DtEntityType](#)& type)
- inline void setTimeStampType([DtTimeStampType](#) type)
- inline [DtThreshold](#)* thresholder()
- inline const [DtThreshold](#)& thresholder() const
- inline [DtTimeStampType](#) timeStampType() const

Direct child classes:

[DtEntityStateRepository](#)

[DtAggregateStateRepository](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file staterep.h:

class DtStateRepository

class DtStateRepository: DtStateRepository is an abstract base class for maintaining state for an object

Inheritance:

DtStateRepository

Public Methods

- virtual **~DtStateRepository** ()
destructor
- DtStateRepository**& **operator=** (const [DtStateRepository](#)& orig)
assignment operator
- virtual [DtStateRepository](#)* **clone** () const
return an empty object of the same type as this
- virtual void **printData** () const
Print state data
- DtStateRepository** ()
default constructor

Protected Methods

- DtStateRepository** (const [DtStateRepository](#)& orig)
copy constructor

Protected

- Constructors are protected since this is an abstract base class*
-

Documentation

class DtStateRepository:

DtStateRepository is an abstract base class for maintaining state for an object. These are used to decode data into the repository from the network and encode data from the repository to the network.

- virtual **~DtStateRepository** ()

destructor

- `DtStateRepository& operator=(const DtStateRepository& orig)`

assignment operator

- `virtual DtStateRepository* clone() const`

return an empty object of the same type as this

- `virtual void printData() const`

Print state data



Constructors are protected since this is an abstract base class

- `DtStateRepository()`

default constructor

- `DtStateRepository(const DtStateRepository& orig)`

copy constructor

Direct child classes:

[DtEmitterBeamRepository](#)

[DtEmbeddedSystemRepository](#)

[DtBaseEntityStateRepository](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file emittbeamsr.h:

class DtEmitterBeamRepository: public DtStateRepository

class DtEmitterBeamRepository: Instances of DtEmitterBeamRepository are

Inheritance:

DtEmitterBeamRepository - DtStateRepository

Public Methods

	DtEmitterBeamRepository () <i>default constructor</i>
 virtual	~DtEmitterBeamRepository () <i>destructor</i>
	DtEmitterBeamRepository (const DtEmitterBeamRepository & orig) <i>copy constructor</i>
 DtEmitterBeamRepository &	operator= (const DtEmitterBeamRepository & orig) <i>assignment operator</i>
 virtual DtStateRepository *	clone () const <i>return an empty object of the same type as this</i>
 static DtEmitterBeamRepository *	create ()
 virtual void	setAzimuthCenter (DtFloat32 center) <i>Set/Get Azimuth Center</i>
 virtual DtFloat32	azimuthCenter () const
 virtual void	setAzimuthSweep (DtFloat32 sweep) <i>Set/Get Azimuth Sweep</i>
 virtual DtFloat32	azimuthSweep () const
 virtual void	setBeamFunction (DtBeamFunc func) <i>Set/Get Beam Function</i>
 virtual DtBeamFunc	beamFunction () const
 virtual void	setBeamId (DtU8 id) <i>Set/Get Beam Id</i>
 virtual DtU8	beamId () const
 virtual void	setBeamParameterIndex (DtU16 index) <i>Set/Get Beam Parameter Index</i>
 virtual DtU16	beamParameterIndex () const

virtual void	setElevationCenter (DtFloat32 center) <i>Set/Get Elevation Center</i>
virtual DtFloat32	elevationCenter () const
virtual void	setElevationSweep (DtFloat32 sweep) <i>Set/Get Elevation Sweep</i>
virtual DtFloat32	elevationSweep () const
virtual void	setEmittingSystemId (const DtGlobalObjectDesignator& id) <i>Set/Get Emitting System Id</i>
virtual const DtGlobalObjectDesignator&	emittingSystemId () const
virtual void	setERP (DtFloat32 erp) <i>Set/Get ERP</i>
virtual DtFloat32	ERP () const
virtual void	setEntityId (const DtEntityIdentifier & id) <i>Set/Get Entity Id - not used after RPR 0</i>
virtual const DtEntityIdentifier &	entityId () const
virtual void	setEventId (const DtEventId & id) <i>Set/Get Event Id - not used until RPR 0</i>
virtual const DtEventId &	eventId () const
virtual void	setFrequency (DtFloat32 freq) <i>Set/Get Frequency</i>
virtual DtFloat32	frequency () const
virtual void	setFrequencyRange (DtFloat32 freq) <i>Set/Get Frequency Range</i>
virtual DtFloat32	frequencyRange () const
virtual void	setPRF (DtFloat32 prf) <i>Set/Get PRF</i>
virtual DtFloat32	PRF () const
virtual void	setPulseWidth (DtFloat32 width) <i>Set/Get Pulse Width</i>
virtual DtFloat32	pulseWidth () const
virtual void	setSweepSync (DtFloat32 sync) <i>Set/Get Sweep Sync</i>
virtual DtFloat32	sweepSync () const
virtual void	setBeamType (DtBeamType flag) <i>Set/Get beam type - track, jam or neither</i>
virtual DtBeamType	beamType () const
virtual void	setJammingModeSequence (DtU32 sequence) <i>Set/Get Jamming Mode Sequence</i>
virtual DtU32	jammingModeSequence () const
virtual const DtTrackJamDesignatorList &	trackJamTargets () const <i>Get set of track jam targets</i>
virtual DtTrackJamDesignatorList *	trackJamTargets ()
virtual void	setHighDensity (DtBoolean val) <i>Set/Get the high density track/jam flag</i>
virtual DtBoolean	highDensity () const

virtual void

printData () const

Public



The next two accessors should only be called if the beam type is DtTrackBeam or DtJamBeam

Protected Fields



DtFloat32

myAzimuthCenter

Common to emitter beams and track jam beams



DtFloat32

myAzimuthSweep



[DtBeamFunc](#)

myBeamFunction



DtU8

myBeamId



DtU16

myBeamParameterIndex



DtFloat32

myElevationCenter



DtFloat32

myElevationSweep



DtGlobalObjectDesignator

myEmittingSystemId



DtFloat32

myERP



[DtEntityIdentifier](#)

myEntityId



[DtEventId](#)

myEventId



DtFloat32

myFrequency



DtFloat32

myFrequencyRange



DtFloat32

myPRF



DtFloat32

myPulseWidth



DtFloat32

mySweepSync



DtU32

myJammingModeSequence

For Track Jam Beams



[DtTrackJamDesignatorList](#)

myTrackJamTargets



[DtBeamType](#)

myBeamType



[DtBoolean](#)

myHighDensity

Protected Methods



virtual void **printTrackJam** () const

Prints track/jam target information

Inherited from [DtStateRepository](#):

Documentation

class DtEmitterBeamRepository: Instances of DtEmitterBeamRepository are

- `DtEmitterBeamRepository()`
default constructor
- `virtual ~DtEmitterBeamRepository()`
destructor
- `DtEmitterBeamRepository(const DtEmitterBeamRepository& orig)`
copy constructor
- `DtEmitterBeamRepository& operator=( const DtEmitterBeamRepository& orig)`
assignment operator
- `virtual DtStateRepository* clone() const`
return an empty object of the same type as this
- `static DtEmitterBeamRepository* create()`
- `virtual void setAzimuthCenter(DtFloat32 center)`
Set/Get Azimuth Center
- `virtual DtFloat32 azimuthCenter() const`
- `virtual void setAzimuthSweep(DtFloat32 sweep)`
Set/Get Azimuth Sweep
- `virtual DtFloat32 azimuthSweep() const`
- `virtual void setBeamFunction(DtBeamFunc func)`
Set/Get Beam Function
- `virtual DtBeamFunc beamFunction() const`
- `virtual void setBeamId(DtU8 id)`
Set/Get Beam Id
- `virtual DtU8 beamId() const`
- `virtual void setBeamParameterIndex(DtU16 index)`
Set/Get Beam Parameter Index
- `virtual DtU16 beamParameterIndex() const`
- `virtual void setElevationCenter(DtFloat32 center)`
Set/Get Elevation Center
- `virtual DtFloat32 elevationCenter() const`

- `virtual void setElevationSweep(DtFloat32 sweep)`
Set/Get Elevation Sweep
- `virtual DtFloat32 elevationSweep() const`
- `virtual void setEmittingSystemId(const DtGlobalObjectDesignator& id)`
Set/Get Emitting System Id
- `virtual const DtGlobalObjectDesignator& emittingSystemId() const`
- `virtual void setERP(DtFloat32 erp)`
Set/Get ERP
- `virtual DtFloat32 ERP() const`
- `virtual void setEntityId(const DtEntityIdentifier& id)`
Set/Get Entity Id - not used after RPR 0
- `virtual const DtEntityIdentifier& entityId() const`
- `virtual void setEventId(const DtEventId& id)`
Set/Get Event Id - not used until RPR 0
- `virtual const DtEventId& eventId() const`
- `virtual void setFrequency(DtFloat32 freq)`
Set/Get Frequency
- `virtual DtFloat32 frequency() const`
- `virtual void setFrequencyRange(DtFloat32 freq)`
Set/Get Frequency Range
- `virtual DtFloat32 frequencyRange() const`
- `virtual void setPRF(DtFloat32 prf)`
Set/Get PRF
- `virtual DtFloat32 PRF() const`
- `virtual void setPulseWidth(DtFloat32 width)`
Set/Get Pulse Width
- `virtual DtFloat32 pulseWidth() const`
- `virtual void setSweepSync(DtFloat32 sync)`
Set/Get Sweep Sync
- `virtual DtFloat32 sweepSync() const`
- `virtual void setBeamType(DtBeamType flag)`
Set/Get beam type - track, jam or neither
- `virtual DtBeamType beamType() const`
-

The next two accessors should only be called if the beam type is DtTrackBeam or DtJamBeam

- virtual void setJammingModeSequence(DtU32 sequence)
Set/Get Jamming Mode Sequence
- virtual DtU32 jammingModeSequence() const
- virtual const [DtTrackJamDesignatorList](#)& trackJamTargets() const
Get set of track jam targets. Returns pointer to internal object. Do not delete. It will be deleted when the SR is deleted.
- virtual [DtTrackJamDesignatorList](#)* trackJamTargets()
- virtual void setHighDensity([DtBoolean](#) val)
Set/Get the high density track/jam flag
- virtual [DtBoolean](#) highDensity() const
- virtual void printData() const
- virtual void printTrackJam() const
Prints track/jam target information
- DtFloat32 myAzimuthCenter
Common to emitter beams and track jam beams
- DtFloat32 myAzimuthSweep
- [DtBeamFunc](#) myBeamFunction
- DtU8 myBeamId
- DtU16 myBeamParameterIndex
- DtFloat32 myElevationCenter
- DtFloat32 myElevationSweep
- DtGlobalObjectDesignator myEmittingSystemId
- DtFloat32 myERP
- [DtEntityIdentifier](#) myEntityId
- [DtEventId](#) myEventId
- DtFloat32 myFrequency
- DtFloat32 myFrequencyRange
- DtFloat32 myPRF
- DtFloat32 myPulseWidth
- DtFloat32 mySweepSync
- DtU32 myJammingModeSequence
For Track Jam Beams
- [DtTrackJamDesignatorList](#) myTrackJamTargets

 [DtBeamType](#) myBeamType

 [DtBoolean](#) myHighDensity

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtBeamFunc

-  DtBeamFuncOther
 -  DtBeamFuncSearch
 -  DtBeamFuncHeightFinder
 -  DtBeamFuncAcquisition
 -  DtBeamFuncTracking
 -  DtBeamFuncAcquisitionAndTracking
 -  DtBeamFuncCommandGuidance
 -  DtBeamFuncIllumination
 -  DtBeamFuncRangeOnlyRadar
 -  DtBeamFuncMissileBeacon
 -  DtBeamFuncMissileFuze
 -  DtBeamFuncActiveRadarMissileSeeker
 -  BeamFuncJammer
-

Documentation

-  DtBeamFuncOther
-  DtBeamFuncSearch
-  DtBeamFuncHeightFinder
-  DtBeamFuncAcquisition
-  DtBeamFuncTracking
-  DtBeamFuncAcquisitionAndTracking
-  DtBeamFuncCommandGuidance
-  DtBeamFuncIllumination
-  DtBeamFuncRangeOnlyRadar
-  DtBeamFuncMissileBeacon

- DtBeamFuncMissileFuze
- DtBeamFuncActiveRadarMissileSeeker
- BeamFuncJammer

[Alphabetic index](#) [Hierarchy of classes](#)

- DtEventId([DtNetEventId](#) netId)
- DtEventId()
- void init(int site, int host, int eventNum)
- void setFromNet([DtNetEventId](#) netId)
- operator DtNetEventId () const
- int site() const
- int host() const
- int eventNum() const
- void setSite(int site)
- void setHost(int host)
- void setEventNum(int eventNum)
- char* string() const
- int operator==(const [DtEventId](#) &otherGuy) const
- int operator!=(const [DtEventId](#) &otherGuy) const
- static const [DtEventId](#)& nullId()
- DtSimulationAddress DtSimulator
Public for backwards compatibility
- DtU16 DtEvent

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file netstructs.h:

class DtNetEventId

Public Fields

 [DtNetSimulationAddress](#) DtSimulator

 DtNetU16 DtEvent

Public Methods

 [DtEventId](#) operator = ([DtEventId](#))

 [DtNetEventId](#) operator = ([DtNetEventId](#))

Documentation

 [DtNetSimulationAddress](#) DtSimulator

 DtNetU16 DtEvent

 [DtEventId](#) operator = ([DtEventId](#))

 [DtNetEventId](#) operator = ([DtNetEventId](#))

This class has no child classes.

In file emitttrbeamsr.h:

enum DtBeamType

-
-  DtRegularBeam
 -  DtRadarBeam
 -  DtTrackBeam
 -  DtJamBeam
-

Documentation

-  DtRegularBeam
-  DtRadarBeam
-  DtTrackBeam
-  DtJamBeam

In file tkjmdeslst.h:

class DtTrackJamDesignatorList

Public Methods

	DtTrackJamDesignatorList () <i>default constructor</i>
 virtual	~DtTrackJamDesignatorList () <i>destructor</i>
	DtTrackJamDesignatorList (const DtTrackJamDesignatorList & orig) <i>copy constructor</i>
 DtTrackJamDesignatorList &	operator= (const DtTrackJamDesignatorList & orig) <i>assignment operator</i>
 int	operator== (const DtTrackJamDesignatorList & other) const <i>equivalence operators</i>
 int	operator!= (const DtTrackJamDesignatorList & other) const
 virtual void	setNumObjects (int num) <i>Set/Get number of objects</i>
 virtual int	numObjects () const
 virtual DtBoolean	setObject (int index, const DtTrackJamDesignator & des) <i>Set index'th object designator</i>
 virtual const DtTrackJamDesignator &	object (int index, DtBoolean * valid = NULL) const <i>Get index'th object designator</i>
 virtual void	print () const <i>Print data</i>

Protected Fields

 int	myNum
 DtTrackJamDesignator *	myObjs

Documentation

	DtTrackJamDesignatorList () default constructor
 virtual	~DtTrackJamDesignatorList () destructor
	DtTrackJamDesignatorList (const DtTrackJamDesignatorList & orig)

copy constructor

`DtTrackJamDesignatorList& operator=( const DtTrackJamDesignatorList& orig)`

assignment operator

`int operator==(const DtTrackJamDesignatorList& other) const`

equivalence operators

`int operator!=(const DtTrackJamDesignatorList& other) const`

`virtual void setNumObjects(int num)`

Set/Get number of objects. Setting to a number lower than the current number will remove trailing objects from the list.

`virtual int numObjects() const`

`virtual DtBoolean setObject(int index, const DtTrackJamDesignator& des)`

Set index'th object designator

`virtual const DtTrackJamDesignator& object( int index, DtBoolean* valid = NULL) const`

Get index'th object designator. Set valid to DtFalse if bad index, DtTrue otherwise.

`virtual void print() const`

Print data

`int myNum`

`DtTrackJamDesignator* myObjs`

This class has no child classes.

[*Alphabetic index*](#) [*Hierarchy of classes*](#)

In file trackjamdes.h:

class DtTrackJamDesignator: public DtGlobalObjectDesignator

class DtTrackJamDesignator: Instances of DtTrackJamDesignator are

Inheritance:

DtTrackJamDesignator

Public Methods

	DtTrackJamDesignator () <i>default constructor</i>
	DtTrackJamDesignator (const DtEntityIdentifier & entId, int emitterId, int beamId) <i>default constructor</i>
 virtual	~DtTrackJamDesignator () <i>destructor</i>
	DtTrackJamDesignator (const DtTrackJamDesignator & orig) <i>copy constructor</i>
 DtTrackJamDesignator &	operator= (const DtTrackJamDesignator & orig) <i>assignment operator</i>
 virtual void	setEmitterId (int id) <i>set/get the emitter Id</i>
 virtual int	emitterId () const
 virtual void	setBeamId (int id) <i>set/get the beam Id</i>
 virtual int	beamId () const
 virtual DtBoolean	operator== (const DtTrackJamDesignator & des) const <i>comparison operators</i>
 virtual DtBoolean	operator!= (const DtTrackJamDesignator & des) const
 virtual const char*	string () const <i>returns a string representation</i>

Protected Fields

 int	myEmitterId
 int	myBeamId

Documentation

class DtTrackJamDesignator: Instances of DtTrackJamDesignator are

-  `DtTrackJamDesignator()`
default constructor
-  `DtTrackJamDesignator( const DtEntityIdentifier& entId, int emitterId, int beamId)`
default constructor
-  `virtual ~DtTrackJamDesignator()`
destructor
-  `DtTrackJamDesignator(const DtTrackJamDesignator& orig)`
copy constructor
-  `DtTrackJamDesignator& operator=(const DtTrackJamDesignator& orig)`
assignment operator
-  `virtual void setEmitterId(int id)`
set/get the emitter Id
-  `virtual int emitterId() const`
-  `virtual void setBeamId(int id)`
set/get the beam Id
-  `virtual int beamId() const`
-  `virtual DtBoolean operator==(const DtTrackJamDesignator& des) const`
comparison operators
-  `virtual DtBoolean operator!=(const DtTrackJamDesignator& des) const`
-  `virtual const char* string() const`
returns a string representation
-  `int myEmitterId`
-  `int myBeamId`
-  `DtString myStringRep`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file embedsystemsr.h:

class [DtEmbeddedSystemRepository](#): public [DtStateRepository](#)

class DtEmbeddedSystemRepository: Instances of DtEmbeddedSystemRepository are

Inheritance:

DtEmbeddedSystemRepository - [DtStateRepository](#)

Public Methods

	DtEmbeddedSystemRepository () <i>default constructor</i>
	DtEmbeddedSystemRepository (const DtEmbeddedSystemRepository & orig) <i>copy constructor</i>
 virtual	~DtEmbeddedSystemRepository () <i>destructor</i>
 DtEmbeddedSystemRepository &	operator= (const DtEmbeddedSystemRepository & orig) <i>assignment operator</i>
 virtual DtStateRepository *	clone () const <i>return an empty object of the same type as this</i>
 static DtEmbeddedSystemRepository *	create ()
 virtual void	setEntityId (const DtEntityIdentifier & id) <i>Set/Get the entity identifier</i>
 virtual const DtEntityIdentifier &	entityId () const
 virtual void	setHostId (const DtGlobalObjectDesignator& id) <i>Set/Get the host identifier</i>
 virtual const DtGlobalObjectDesignator&	hostId () const
 virtual void	setRelativePosition (DtConstVector position) <i>Set/Get the relative position</i>
 virtual DtConstVector	relativePosition () const
 virtual void	printData () const

Protected Fields

DtEntityIdentifier	myEntityId <i>embedded system class attributes</i>
DtGlobalObjectDesignator	myHostId
DtVector	myRelativePosition

Inherited from [DtStateRepository](#):

Protected



Documentation

class DtEmbeddedSystemRepository: Instances of DtEmbeddedSystemRepository are

- DtEmbeddedSystemRepository()**
default constructor
- DtEmbeddedSystemRepository(const [DtEmbeddedSystemRepository](#)& orig)**
copy constructor
- virtual ~DtEmbeddedSystemRepository()**
destructor
- [DtEmbeddedSystemRepository](#)& operator=(const [DtEmbeddedSystemRepository](#)& orig)**
assignment operator
- virtual [DtStateRepository](#)* clone() const**
return an empty object of the same type as this
- static [DtEmbeddedSystemRepository](#)* create()**
- virtual void setEntityId(const [DtEntityIdentifier](#)& id)**
Set/Get the entity identifier
- virtual const [DtEntityIdentifier](#)& entityId() const**
- virtual void setHostId(const DtGlobalObjectDesignator& id)**
Set/Get the host identifier
- virtual const DtGlobalObjectDesignator& hostId() const**
- virtual void setRelativePosition(DtConstVector position)**
Set/Get the relative position
- virtual DtConstVector relativePosition() const**

virtual void printData() const

[DtEntityIdentifier](#) myEntityId

embedded system class attributes

DtGlobalObjectDesignator myHostId

[DtVector](#) myRelativePosition

Direct child classes:

[DtRadioTransmitterRepository](#)

[DtRadioReceiverRepository](#)

[DtEmitterSystemRepository](#)

[DtDesignatorRepository](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file radioxmitsr.h:

class DtRadioTransmitterRepository: public DtEmbeddedSystemRepository

class DtRadioTransmitterRepository: Instances of DtRadioTransmitterRepository are

Inheritance:

DtRadioTransmitterRepository - DtEmbeddedSystemRepository - DtStateRepository

Public Methods

	DtRadioTransmitterRepository () <i>default constructor</i>
virtual	~DtRadioTransmitterRepository () <i>destructor</i>
	DtRadioTransmitterRepository (const DtRadioTransmitterRepository & orig) <i>copy constructor</i>
DtRadioTransmitterRepository &	operator= (const DtRadioTransmitterRepository & orig) <i>assignment operator</i>
virtual DtStateRepository *	clone () const <i>return an empty object of the same type as this</i>
virtual void	setAntennaPattern (const DtAntennaPatternRecord & rec) <i>Set/Get the antenna pattern</i>
virtual const DtAntennaPatternRecord &	antennaPattern () const
virtual void	setAntennaPatternType (DtAntennaPatternType pattern) <i>Set/Get the antenna pattern type</i>
virtual DtAntennaPatternType	antennaPatternType () const
virtual void	setCryptoKeyId (DtU16 key) <i>Set/Get the crypto key id</i>
virtual DtU16	cryptoKeyId () const
virtual void	setCryptoMode (DtCryptoMode mode) <i>Set/Get the crypto key id</i>
virtual DtCryptoMode	cryptoMode () const
virtual void	setCryptoSystem (DtRadioCryptoSystem system) <i>Set/Get the crypto system</i>
virtual DtRadioCryptoSystem	cryptoSystem () const

- virtual void
- virtual [DtDetailedModulationType](#)
- virtual void
- virtual void
- virtual unsigned long
- virtual void
- virtual DtFloat32
- virtual void
- virtual [DtBoolean](#)
- virtual void
- virtual [DtRadioInputSource](#)
- virtual void
- virtual const void*
- virtual DtU8
- virtual void
- virtual [DtRadioMajorModulation](#)
- virtual void
- virtual [DtRadioSystem](#)
- virtual void
- virtual DtFloat32
- virtual void
- virtual [DtBoolean](#)
- virtual void
- virtual unsigned int
- virtual void
- virtual const [DtRadioEntityType](#)&
- virtual void
- virtual [DtBoolean](#)
- virtual void

setDetailedModulationType ([DtDetailedModulationType](#) modulation)
Set/Get the detailed modulation

detailedModulationType () const

setFrequency (unsigned long upper, unsigned long lower)
Set the frequency, either as upper and lower word, or as full 64 bit integer

setFrequency (unsigned long freq)

frequency ([DtLongWordPortion](#) upperLower = DtLowerWord) const
Get either the upper or lower word of the 64 bit frequency

setFrequencyBandwidth (DtFloat32 bandwidth)
Set/Get the frequency bandwidth

frequencyBandwidth () const

setFrequencyHopInUse ([DtBoolean](#) hopInUse)
Set/Get the frequency hop in use

frequencyHopInUse () const

setInputSource ([DtRadioInputSource](#) source)
Set/Get the input source

inputSource () const

setModulationParams (const void *networkRepresentation, DtU8 size)
Set/Get the modulation parameters' network representation

modulationParamsAddr () const

modulationParamsLength () const

setMajorModulationType ([DtRadioMajorModulation](#) modulation)
Set/Get the major modulation type

majorModulationType () const

setModulationSystemType ([DtRadioSystem](#) system)
Set/Get the modulation system

modulationSystemType () const

setPower (DtFloat32 power)
Set/Get the power

power () const

setPsuedoNoiseSpectrumInUse ([DtBoolean](#) inUse)
Set/Get the Psuedo Noise Spectrum In Use

psuedoNoiseSpectrumInUse () const

setRadioid (unsigned int number)
Set/Get the Transmitter radio number

radioid () const

setRadioEntityType (const [DtRadioEntityType](#)& type)
Set/Get the radio entity type

radioEntityType () const

setTimeHopInUse ([DtBoolean](#) inUse)
Set/Get the Time Hop In Use

timeHopInUse () const

setTransmitState ([DtRadioTransmitState](#) state)
Set/Get the transmit state

virtual DtRadioTransmitState	transmitState () const
virtual void	setWorldAntennaLoc (DtConstVector location) <i>Set/Get the world location</i>
virtual DtConstVector	worldAntennaLoc () const
virtual void	setRelativeAntennaLoc (DtConstVector location) <i>Set/Get the relative location (wrapper around EmbedSysSR member func, for compatibility with DIS transmitter PDU)</i>
virtual DtConstVector	relativeAntennaLoc () const
virtual void	printData () const
static DtRadioTransmitterRepository *	create ()

Protected Fields

DtAntennaPatternType	myAntennaPatternType
DtU16	myCryptoKeyId
DtCryptoMode	myCryptoMode
DtRadioCryptoSystem	myCryptoSystem
DtDetailedModulationType	myDetailedModulation
unsigned long	myFreqHi
unsigned long	myFreqLo
DtFloat32	myFrequencyBandwidth
DtBoolean	myFrequencyHopInUse
DtRadioInputSource	myInputSource
char*	myModulationParamsAddr
DtU8	myModulationParamsLength
DtAntennaPatternRecord *	myAntPat
DtRadioMajorModulation	myMajorModulationType
DtRadioSystem	myModulationSystem
DtFloat32	myPower
DtBoolean	myPsuedoNoiseSpectrumInUse
DtU16	myTransmitterRadioNumber
DtRadioEntityType	myRadioEntityType
DtBoolean	myTimeHopInUse
DtRadioTransmitState	myTransmitState
DtVector	myWorldLocation

Inherited from [DtEmbeddedSystemRepository](#):

Public Methods

- virtual void **setEntityId**(const [DtEntityIdentifier](#)& id)
- virtual const [DtEntityIdentifier](#)& **entityId**() const
- virtual void **setHostId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **hostId**() const
- virtual void **setRelativePosition**(DtConstVector position)
- virtual DtConstVector **relativePosition**() const

Protected Fields

- [DtEntityIdentifier](#) myEntityId
 - DtGlobalObjectDesignator myHostId
 - [DtVector](#) myRelativePosition
-

Inherited from [DtStateRepository](#):

Protected



Documentation

class DtRadioTransmitterRepository: Instances of DtRadioTransmitterRepository are

- [DtRadioTransmitterRepository](#) ()
default constructor
- virtual [~DtRadioTransmitterRepository](#) ()
destructor
- [DtRadioTransmitterRepository](#) (const [DtRadioTransmitterRepository](#)& orig)
copy constructor
- [DtRadioTransmitterRepository](#)& operator=(const [DtRadioTransmitterRepository](#)& orig)
assignment operator
- virtual [DtStateRepository](#)* clone() const
return an empty object of the same type as this

- `virtual void setAntennaPattern(const DtAntennaPatternRecord& rec)`
Set/Get the antenna pattern
- `virtual const DtAntennaPatternRecord& antennaPattern() const`
- `virtual void setAntennaPatternType(DtAntennaPatternType pattern)`
Set/Get the antenna pattern type
- `virtual DtAntennaPatternType antennaPatternType() const`
- `virtual void setCryptoKeyId(DtU16 key)`
Set/Get the crypto key id
- `virtual DtU16 cryptoKeyId() const`
- `virtual void setCryptoMode(DtCryptoMode mode)`
Set/Get the crypto key id
- `virtual DtCryptoMode cryptoMode() const`
- `virtual void setCryptoSystem(DtRadioCryptoSystem system)`
Set/Get the crypto system
- `virtual DtRadioCryptoSystem cryptoSystem() const`
- `virtual void setDetailedModulationType(DtDetailedModulationType modulation)`
Set/Get the detailed modulation
- `virtual DtDetailedModulationType detailedModulationType() const`
- `virtual void setFrequency(unsigned long upper, unsigned long lower)`
Set the frequency, either as upper and lower word, or as full 64 bit integer
- `virtual void setFrequency(unsigned long freq)`
- `virtual unsigned long frequency( DtLongWordPortion upperLower = DtLowerWord) const`
Get either the upper or lower word of the 64 bit frequency
- `virtual void setFrequencyBandwidth(DtFloat32 bandwidth)`
Set/Get the frequency bandwidth
- `virtual DtFloat32 frequencyBandwidth() const`
- `virtual void setFrequencyHopInUse(DtBoolean hopInUse)`
Set/Get the frequency hop in use
- `virtual DtBoolean frequencyHopInUse() const`
- `virtual void setInputSource(DtRadioInputSource source)`
Set/Get the input source
- `virtual DtRadioInputSource inputSource() const`
- `virtual void setModulationParams( const void *networkRepresentation, DtU8 size)`
Set/Get the modulation parameters' network representation

- `virtual const void* modulationParamsAddr() const`
- `virtual DtU8 modulationParamsLength() const`
- `virtual void setMajorModulationType(DtRadioMajorModulation modulation)`
Set/Get the major modulation type
- `virtual DtRadioMajorModulation majorModulationType() const`
- `virtual void setModulationSystemType(DtRadioSystem system)`
Set/Get the modulation system
- `virtual DtRadioSystem modulationSystemType() const`
- `virtual void setPower(DtFloat32 power)`
Set/Get the power
- `virtual DtFloat32 power() const`
- `virtual void setPsuedoNoiseSpectrumInUse(DtBoolean inUse)`
Set/Get the Psuedo Noise Spectrum In Use
- `virtual DtBoolean psuedoNoiseSpectrumInUse() const`
- `virtual void setRadioId(unsigned int number)`
Set/Get the Transmitter radio number
- `virtual unsigned int radioId() const`
- `virtual void setRadioEntityType(const DtRadioEntityType& type)`
Set/Get the radio entity type
- `virtual const DtRadioEntityType& radioEntityType() const`
- `virtual void setTimeHopInUse(DtBoolean inUse)`
Set/Get the Time Hop In Use
- `virtual DtBoolean timeHopInUse() const`
- `virtual void setTransmitState(DtRadioTransmitState state)`
Set/Get the transmit state
- `virtual DtRadioTransmitState transmitState() const`
- `virtual void setWorldAntennaLoc(DtConstVector location)`
Set/Get the world location
- `virtual DtConstVector worldAntennaLoc() const`
- `virtual void setRelativeAntennaLoc(DtConstVector location)`
Set/Get the relative location (wrapper around EmbedSysSR member func, for compatibility with DIS transmitter PDU)
- `virtual DtConstVector relativeAntennaLoc() const`
- `virtual void printData() const`

- static [DtRadioTransmitterRepository](#)* create()
- [DtAntennaPatternType](#) myAntennaPatternType
- DtU16 myCryptoKeyId
- [DtCryptoMode](#) myCryptoMode
- [DtRadioCryptoSystem](#) myCryptoSystem
- [DtDetailedModulationType](#) myDetailedModulation
- unsigned long myFreqHi
- unsigned long myFreqLo
- DtFloat32 myFrequencyBandwidth
- [DtBoolean](#) myFrequencyHopInUse
- [DtRadioInputSource](#) myInputSource
- char* myModulationParamsAddr
- DtU8 myModulationParamsLength
- [DtAntennaPatternRecord](#)* myAntPat
- [DtRadioMajorModulation](#) myMajorModulationType
- [DtRadioSystem](#) myModulationSystem
- DtFloat32 myPower
- [DtBoolean](#) myPsuedoNoiseSpectrumInUse
- DtU16 myTransmitterRadioNumber
- [DtRadioEntityType](#) myRadioEntityType
- [DtBoolean](#) myTimeHopInUse
- [DtRadioTransmitState](#) myTransmitState
- [DtVector](#) myWorldLocation

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtCryptoMode

 DtBasebandEncryption

 DtDiphaseEncryption

Documentation

 DtBasebandEncryption

 DtDiphaseEncryption

In file disenums.h:

enum DtDetailedModulationType

 DtDetailedModulationDummy

Documentation

 DtDetailedModulationDummy

[Alphabetic index](#) [Hierarchy of classes](#)

In file radioxmitsr.h:

enum DtLongWordPortion

 DtUpperWord

 DtLowerWord

Documentation

 DtUpperWord

 DtLowerWord

[Alphabetic index](#) [Hierarchy of classes](#)

In file radiorecvrsr.h:

class [DtRadioReceiverRepository](#): public [DtEmbeddedSystemRepository](#)

class DtRadioReceiverRepository: Instances of DtRadioReceiverRepository are

Inheritance:

DtRadioReceiverRepository - [DtEmbeddedSystemRepository](#) - [DtStateRepository](#)

Public Methods

	DtRadioReceiverRepository () <i>default constructor</i>
virtual	~DtRadioReceiverRepository () <i>destructor</i>
	DtRadioReceiverRepository (const DtRadioReceiverRepository & orig) <i>copy constructor</i>
DtRadioReceiverRepository &	operator= (const DtRadioReceiverRepository & orig) <i>assignment operator</i>
virtual DtStateRepository *	clone () const <i>return an empty object of the same type as this</i>
static DtRadioReceiverRepository *	create ()
virtual void	setRadioid (unsigned int number) <i>Set/Get the Receiver radio number</i>
virtual unsigned int	radioid () const
virtual void	setReceivedPower (DtFloat32 power) <i>Set/Get the received power</i>
virtual DtFloat32	receivedPower () const
virtual void	setTransmitRadioid (const DtGlobalObjectDesignator& id) <i>Set/Get the received transmitter</i>
virtual const DtGlobalObjectDesignator&	transmitRadioid () const
virtual void	setReceiverState (DtRadioReceiverState state) <i>Set/Get the receiver state</i>
virtual DtRadioReceiverState	receiverState () const
virtual void	printData () const

Protected Fields

 <code>int</code>	<code>myReceiverRadioNumber</code>
 <code>DtFloat32</code>	<code>myReceivedPower</code>
 <code>DtGlobalObjectDesignator</code>	<code>myTransmitId</code>
 <code>DtRadioReceiverState</code>	<code>myReceiverState</code>

Inherited from [`DtEmbeddedSystemRepository`](#):

Public Methods

-  virtual void **setEntityId**(const [`DtEntityIdentifier`](#)& id)
-  virtual const [`DtEntityIdentifier`](#)& **entityId**() const
-  virtual void **setHostId**(const DtGlobalObjectDesignator& id)
-  virtual const DtGlobalObjectDesignator& **hostId**() const
-  virtual void **setRelativePosition**(DtConstVector position)
-  virtual DtConstVector **relativePosition**() const

Protected Fields

-  [`DtEntityIdentifier`](#) `myEntityId`
 -  DtGlobalObjectDesignator `myHostId`
 -  [`DtVector`](#) `myRelativePosition`
-

Inherited from [`DtStateRepository`](#):

Protected



Documentation

class DtRadioReceiverRepository: Instances of DtRadioReceiverRepository are

-  `DtRadioReceiverRepository()`
default constructor
-  `virtual ~DtRadioReceiverRepository()`

destructor

`DtRadioReceiverRepository(const DtRadioReceiverRepository& orig)`

copy constructor

`DtRadioReceiverRepository& operator=(const DtRadioReceiverRepository& orig)`

assignment operator

`virtual DtStateRepository* clone() const`

return an empty object of the same type as this

`static DtRadioReceiverRepository* create()`

`virtual void setRadioId(unsigned int number)`

Set/Get the Receiver radio number

`virtual unsigned int radioId() const`

`virtual void setReceivedPower(DtFloat32 power)`

Set/Get the received power

`virtual DtFloat32 receivedPower() const`

`virtual void setTransmitRadioId(const DtGlobalObjectDesignator& id)`

Set/Get the received transmitter

`virtual const DtGlobalObjectDesignator& transmitRadioId() const`

`virtual void setReceiverState(DtRadioReceiverState state)`

Set/Get the receiver state

`virtual DtRadioReceiverState receiverState() const`

`virtual void printData() const`

`int myReceiverRadioNumber`

`DtFloat32 myReceivedPower`

`DtGlobalObjectDesignator myTransmitId`

`DtRadioReceiverState myReceiverState`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file emittersysr.h:

class [DtEmitterSystemRepository](#): public [DtEmbeddedSystemRepository](#)

class DtEmitterSystemRepository: Instances of DtEmitterSystemRepository are

Inheritance:

DtEmitterSystemRepository - [DtEmbeddedSystemRepository](#) - [DtStateRepository](#)

Public Methods

	DtEmitterSystemRepository () <i>default constructor</i>
virtual	~DtEmitterSystemRepository () <i>destructor</i>
	DtEmitterSystemRepository (const DtEmitterSystemRepository& orig) <i>copy constructor</i>
DtEmitterSystemRepository &	operator= (const DtEmitterSystemRepository& orig) <i>assignment operator</i>
virtual DtStateRepository *	clone () const <i>return an empty object of the same type as this</i>
static DtEmitterSystemRepository *	create ()
virtual void	setEmitterFunction (DtEmissionFunc function) <i>Set/Get the emitter function</i>
virtual DtEmissionFunc	emitterFunction () const
virtual void	setEmitterName (DtEmitterSysName name) <i>Set/Get the emitter name</i>
virtual DtEmitterSysName	emitterName () const
virtual void	setEmitterNumber (DtU8 number) <i>Set/Get the emitter number</i>
virtual DtU8	emitterNumber () const
virtual void	setEventId (const DtEventId& id) <i>Set/Get Event Id - not used until RPR 0</i>
virtual const DtEventId &	eventId () const
virtual void	printData () const

- virtual [DtEmitterBeamRepository](#)* **beamSRById** (DtU8 beamId, [DtEmitterSysRetCode](#)* retcode = NULL)
Return a pointer to the first BeamSR with specified beamID
- virtual [DtEmitterBeamRepository](#)* **addBeam** (DtU8 beamId = 0, [DtBeamType](#) type = DtRegularBeam, [DtEmitterSysRetCode](#)* retcode = NULL)
add/remove a new emitter beam within this system if add a beam with duplicate ID, retcode will be set to DtDuplicateBeam, and beamSRById will return the first beamSR with the given ID
- virtual [DtEmitterSysRetCode](#) **removeBeam** (DtU8 beamId)
- virtual int **numBeams** () const
Returns number of beams
- virtual [DtList](#)* **beamList** ()
Returns DtList of DtEmitterBeamStateRepositories
- [DtBoolean](#) **isMember** ([DtEmitterBeamRepository](#)* bsr) const
Returns DtTrue if the give BSR is in this ESR's list of BSRs

Protected Fields

- [DtEventId](#) **myEventId**
- [DtEmissionFunc](#) **myEmitterFunction**
- [DtEmitterSysName](#) **myEmitterName**
- DtU8 **myEmitterNumber**
- [DtList](#) **myBeamList**
list of beams in this system

Protected Methods

- virtual void **purgeBeamList** ()

Inherited from [DtEmbeddedSystemRepository](#):

Public Methods

- virtual void **setEntityId**(const [DtEntityIdentifier](#)& id)
- virtual const [DtEntityIdentifier](#)& **entityId**() const
- virtual void **setHostId**(const DtGlobalObjectDesignator& id)
- virtual const DtGlobalObjectDesignator& **hostId**() const
- virtual void **setRelativePosition**(DtConstVector position)
- virtual DtConstVector **relativePosition**() const

Protected Fields

- [DtEntityIdentifier](#) myEntityId
 - DtGlobalObjectDesignator myHostId
 - [DtVector](#) myRelativePosition
-

Inherited from [DtStateRepository](#):

Protected



Documentation

class DtEmitterSystemRepository: Instances of DtEmitterSystemRepository are

- `DtEmitterSystemRepository()`
default constructor
- `virtual ~DtEmitterSystemRepository()`
destructor
- `DtEmitterSystemRepository(const DtEmitterSystemRepository& orig)`
copy constructor
- `DtEmitterSystemRepository& operator=(const DtEmitterSystemRepository& orig)`
assignment operator
- `virtual DtStateRepository* clone() const`
return an empty object of the same type as this
- `static DtEmitterSystemRepository* create()`
- `virtual void setEmitterFunction(DtEmissionFunc function)`
Set/Get the emitter function
- `virtual DtEmissionFunc emitterFunction() const`
- `virtual void setEmitterName(DtEmitterSysName name)`
Set/Get the emitter name
- `virtual DtEmitterSysName emitterName() const`
- `virtual void setEmitterNumber(DtU8 number)`
Set/Get the emitter number
- `virtual DtU8 emitterNumber() const`

- `virtual void setEventId(const DtEventId& id)`
Set/Get Event Id - not used until RPR 0
- `virtual const DtEventId& eventId() const`
- `virtual void printData() const`
- `virtual DtEmitterBeamRepository* beamSRById( DtU8 beamId, DtEmitterSysRetCode* retcode = NULL)`
Return a pointer to the first BeamSR with specified beamID
- `virtual DtEmitterBeamRepository* addBeam( DtU8 beamId = 0, DtBeamType type = DtRegularBeam, DtEmitterSysRetCode *retcode = NULL)`
add/remove a new emitter beam within this system if add a beam with duplicate ID, retcode will be set to DtDuplicateBeam, and beamSRById will return the first beamSR with the given ID
- `virtual DtEmitterSysRetCode removeBeam(DtU8 beamId)`
- `virtual int numBeams() const`
Returns number of beams
- `virtual DtList* beamList()`
Returns DtList of DtEmitterBeamStateRepositories
- `DtBoolean isMember(DtEmitterBeamRepository* bsr) const`
Returns DtTrue if the give BSR is in this ESR's list of BSRs
- `virtual void purgeBeamList()`
- `DtEventId myEventId`
- `DtEmissionFunc myEmitterFunction`
- `DtEmitterSysName myEmitterName`
- `DtU8 myEmitterNumber`
- `DtList myBeamList`
list of beams in this system

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file emittersysr.h:

enum DtEmitterSysRetCode

error codes for looking up individual beams

-  **DtEmitterSuccess**
 -  **DtBadBeamId**
 -  **DtDuplicateBeamId**
-

Documentation

error codes for looking up individual beams

-  **DtEmitterSuccess**
-  **DtBadBeamId**
-  **DtDuplicateBeamId**

In file `designatorsr.h`:

class DtDesignatorRepository: public DtEmbeddedSystemRepository

class DtDesignatorRepository: Instances of DtDesignatorRepository are

Inheritance:

DtDesignatorRepository - DtEmbeddedSystemRepository - DtStateRepository

Public Methods

	DtDesignatorRepository () <i>default constructor</i>
virtual	~DtDesignatorRepository () <i>destructor</i>
	DtDesignatorRepository (const DtDesignatorRepository & orig) <i>copy constructor</i>
DtDesignatorRepository &	operator= (const DtDesignatorRepository & orig) <i>assignment operator</i>
virtual DtStateRepository *	clone () const <i>return an empty object of the same type as this</i>
static DtDesignatorRepository *	create ()
virtual void	printData () const
virtual void	setCodeName (const DtDesignatorCodeName & name) <i>Set/Get the code name</i>
virtual const DtDesignatorCodeName &	codeName () const
virtual void	setDesignatedObject (const DtGlobalObjectDesignator& name) <i>Set/Get the designated object name</i>
virtual const DtGlobalObjectDesignator&	designatedObject () const
virtual void	setDesignatorCode (const DtDesignatorCode & code) <i>Set/Get the designator code</i>
virtual const DtDesignatorCode &	designatorCode () const
virtual void	setDesignatorId (const DtGlobalObjectDesignator& id) <i>Set/Get the designator object id</i>
virtual const DtGlobalObjectDesignator&	designatorId () const

virtual void	setDesignatorPower (DtFloat32 power) <i>Set/Get the designator power</i>
virtual DtFloat32	designatorPower () const
virtual void	setDesignatorSpotLocation (DtConstVector location) <i>Set/Get the designator spot location</i>
virtual DtConstVector	designatorSpotLocation () const
virtual void	setDesignatorWavelength (DtFloat32 wavelength) <i>Set/Get the designator wavelength</i>
virtual DtFloat32	designatorWavelength () const
virtual void	setDRAAlgorithm (DtDeadReckonTypes algorithm) <i>Set/Get the dead reckoning algorithm</i>
virtual DtDeadReckonTypes	dRAAlgorithm () const
virtual void	setRelativeSpotLocation (DtConstVector location) <i>Set/Get the relative spot location</i>
virtual DtConstVector	relativeSpotLocation () const
virtual void	setSpotLinearAcceleration (DtConstVector acceleration) <i>Set/Get the spot linear acceleration</i>
virtual DtConstVector	spotLinearAcceleration () const

Protected Fields

DtDesignatorCodeName	myCodeName
DtGlobalObjectDesignator	myDesignatedObject
DtGlobalObjectDesignator	myDesignatorId
DtDesignatorCode	myDesignatorCode
DtFloat32	myDesignatorPower
DtVector	myDesignatorSpotLocation
DtFloat32	myDesignatorWavelength
DtDeadReckonTypes	myDRAAlgorithm
DtVector	myRelativeSpotLocation
DtVector	mySpotLinearAcceleration

Inherited from [DtEmbeddedSystemRepository](#):

Public Methods

virtual void	setEntityId (const DtEntityIdentifier & id)
virtual const DtEntityIdentifier &	entityId () const
virtual void	setHostId (const DtGlobalObjectDesignator& id)

- virtual const DtGlobalObjectDesignator& **hostId**() const
- virtual void **setRelativePosition**(DtConstVector position)
- virtual DtConstVector **relativePosition**() const

Protected Fields

- [DtEntityIdentifier](#) myEntityId
- DtGlobalObjectDesignator myHostId
- [DtVector](#) myRelativePosition

Inherited from [DtStateRepository](#):

Protected

Documentation

class DtDesignatorRepository: Instances of DtDesignatorRepository are

- DtDesignatorRepository()**
default constructor
- virtual ~DtDesignatorRepository()**
destructor
- DtDesignatorRepository(const [DtDesignatorRepository](#)& orig)**
copy constructor
- [DtDesignatorRepository](#)& operator=(const [DtDesignatorRepository](#)& orig)**
assignment operator
- virtual [DtStateRepository](#)* clone() const**
return an empty object of the same type as this
- static [DtDesignatorRepository](#)* create()**
- virtual void printData() const**
- virtual void setCodeName(const [DtDesignatorCodeName](#)& name)**
Set/Get the code name
- virtual const [DtDesignatorCodeName](#)& codeName() const**
- virtual void setDesignatedObject(const DtGlobalObjectDesignator& name)**

Set/Get the designated object name

virtual const DtGlobalObjectDesignator& designatedObject() const

virtual void setDesignatorCode(const DtDesignatorCode& code)

Set/Get the designator code

virtual const DtDesignatorCode& designatorCode() const

virtual void setDesignatorId(const DtGlobalObjectDesignator& id)

Set/Get the designator object id. Not communicated in RPR FOM 0.7.

virtual const DtGlobalObjectDesignator& designatorId() const

virtual void setDesignatorPower(DtFloat32 power)

Set/Get the designator power

virtual DtFloat32 designatorPower() const

virtual void setDesignatorSpotLocation(DtConstVector location)

Set/Get the designator spot location

virtual DtConstVector designatorSpotLocation() const

virtual void setDesignatorWavelength(DtFloat32 wavelength)

Set/Get the designator wavelength

virtual DtFloat32 designatorWavelength() const

virtual void setDRAlgorithm(DtDeadReckonTypes algorithm)

Set/Get the dead reckoning algorithm

virtual DtDeadReckonTypes dRAlgorithm() const

virtual void setRelativeSpotLocation(DtConstVector location)

Set/Get the relative spot location

virtual DtConstVector relativeSpotLocation() const

virtual void setSpotLinearAcceleration(DtConstVector acceleration)

Set/Get the spot linear acceleration

virtual DtConstVector spotLinearAcceleration() const

DtDesignatorCodeName myCodeName

DtGlobalObjectDesignator myDesignatedObject

DtGlobalObjectDesignator myDesignatorId

DtDesignatorCode myDesignatorCode

DtFloat32 myDesignatorPower

DtVector myDesignatorSpotLocation

DtFloat32 myDesignatorWavelength

- [DtDeadReckonTypes](#) myDRAlgorithm
 - [DtVector](#) myRelativeSpotLocation
 - [DtVector](#) mySpotLinearAcceleration
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file approximator.h:

class DtApproximator

class DtApproximator: DtApproximator is an abstract base class

Inheritance:

DtApproximator

Public Methods

virtual	~DtApproximator () <i>destructor</i>
DtApproximator &	operator= (const DtApproximator & orig) <i>assignment operator</i>
virtual DtApproximator *	clone () const <i>Create a copy of this object</i>
virtual void	setLocation (DtConstVector loc, DtTime t = DtGetSimTime()) <i>Set absolute location</i>
virtual DtConstVector	approxLocation (DtTime t = DtGetSimTime()) const <i>Get the approximated location at the specified time</i>
virtual void	setVelocity (DtConstVector vel, DtTime t = DtGetSimTime()) <i>Set absolute velocity</i>
virtual DtConstVector	approxVelocity (DtTime t = DtGetSimTime()) const <i>Get the approximated velocity at the specified time</i>
virtual void	setOrientation (const DtTaitBryan & ori, DtTime t = DtGetSimTime()) <i>Set absolute orientation</i>
virtual const DtTaitBryan &	approxEuler (DtTime t = DtGetSimTime()) const <i>Get approximated orientation (euler angles) at the specified time</i>
virtual void	setAcceleration (DtConstVector acc, DtTime t = DtGetSimTime()) <i>Set/Get absolute acceleration</i>
virtual DtConstVector	acceleration (DtTime t = DtGetSimTime()) const
virtual void	setRotationalVelocity (DtConstVector vel, DtTime t = DtGetSimTime()) <i>Set/Get absolute rotational velocity</i>
virtual DtConstVector	rotationalVelocity (DtTime t = DtGetSimTime()) const
virtual DtConstDcm	approxBodyToGeoc (DtTime t = DtGetSimTime()) const <i>Get the approximated data body to geocentric coordinate matrix</i>
	DtApproximator () <i>default constructor</i>

Protected Methods

- **DtApproximator** (const [DtApproximator](#)& orig)
copy constructor

Protected

- *Constructors are protected since this is an abstract base class*
-

Documentation

class DtApproximator:

DtApproximator is an abstract base class. It is used to provide approximated position and orientation information for entities. Position and orientation information is approximated over time from the last set absolute (non-approximated) information.

- **virtual ~DtApproximator()**
destructor
- **[DtApproximator](#)& operator=(const [DtApproximator](#)& orig)**
assignment operator
- **virtual [DtApproximator](#)* clone() const**
Create a copy of this object. Memory is allocated and it is the responsibility of the caller to free the memory via delete.
- **virtual void setLocation(DtConstVector loc, DtTime t = DtGetSimTime())**
Set absolute location
- **virtual DtConstVector approxLocation(DtTime t = DtGetSimTime()) const**
Get the approximated location at the specified time
- **virtual void setVelocity(DtConstVector vel, DtTime t = DtGetSimTime())**
Set absolute velocity
- **virtual DtConstVector approxVelocity(DtTime t = DtGetSimTime()) const**
Get the approximated velocity at the specified time
- **virtual void setOrientation(const [DtTaitBryan](#)& ori, DtTime t = DtGetSimTime())**
Set absolute orientation
- **virtual const [DtTaitBryan](#)& approxEuler(DtTime t = DtGetSimTime()) const**
Get approximated orientation (euler angles) at the specified time
- **virtual void setAcceleration(DtConstVector acc, DtTime t = DtGetSimTime())**
Set/Get absolute acceleration
- **virtual DtConstVector acceleration(DtTime t = DtGetSimTime()) const**

- `virtual void setRotationalVelocity(DtConstVector vel, DtTime t = DtGetSimTime())`
Set/Get absolute rotational velocity
 - `virtual DtConstVector rotationalVelocity( DtTime t = DtGetSimTime()) const`
 - `virtual DtConstDcm approxBodyToGeoc(DtTime t = DtGetSimTime()) const`
Get the approximated data body to geocentric coordinate matrix
 - Constructors are protected since this is an abstract base class
 - `DtApproximator()`
default constructor
 - `DtApproximator(const DtApproximator& orig)`
copy constructor
-

Direct child classes:

[DtDeadReckoner](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file deadreckoner.h:

class DtDeadReckoner: public DtApproximator

class DtDeadReckoner: Instances of DtDeadReckoner are used to provide approximated position and orientation information for entities using various dead reckoning algorithms

Inheritance:

DtDeadReckoner - DtApproximator

Public Methods

	DtDeadReckoner () <i>default constructor</i>
virtual	~DtDeadReckoner () <i>destructor</i>
	DtDeadReckoner (const DtDeadReckoner & orig) <i>copy constructor</i>
DtDeadReckoner &	operator= (const DtDeadReckoner & orig) <i>assignment operator</i>
virtual DtApproximator *	clone () const <i>Create a copy of this object</i>
virtual void	setLocation (DtConstVector loc, DtTime curTime = DtGetSimTime()) <i>Set absolute location</i>
virtual DtConstVector	approxLocation (DtTime curTime = DtGetSimTime()) const <i>Get the approximated (i</i>
virtual DtConstVector	deadReckonedLocation (DtTime curTime = DtGetSimTime()) const <i>Get the dead-reckoned location at the specified time</i>
virtual void	setVelocity (DtConstVector vel, DtTime curTime = DtGetSimTime()) <i>Set absolute velocity</i>
virtual DtConstVector	approxVelocity (DtTime curTime = DtGetSimTime()) const <i>Get the approximated (i</i>
virtual DtConstVector	deadReckonedVelocity (DtTime curTime = DtGetSimTime()) const <i>Get the dead-reckoned velocity at the specified time</i>
virtual void	setOrientation (const DtTaitBryan & ori, DtTime curTime = DtGetSimTime()) <i>Set absolute orientation</i>
virtual const DtTaitBryan &	approxEuler (DtTime curTime = DtGetSimTime()) const <i>Get the approximated (i</i>
virtual const DtTaitBryan &	deadReckonedEuler (DtTime curTime = DtGetSimTime()) const <i>Get dead-reckoned orientation (euler angles) at the specified time</i>
virtual void	setAcceleration (DtConstVector acc, DtTime curTime = DtGetSimTime()) <i>Set/Get absolute acceleration</i>
virtual DtConstVector	acceleration (DtTime curTime = DtGetSimTime()) const

virtual void	setRotationalVelocity (DtConstVector vel, DtTime curTime = DtGetSimTime()) <i>Set/Get absolute rotational velocity</i>
virtual DtConstVector	rotationalVelocity (DtTime curTime = DtGetSimTime()) const
virtual DtConstDcm	approxBodyToGeoc (DtTime curTime = DtGetSimTime()) const <i>Get the approximated (i</i>
virtual DtConstDcm	deadReckonedBodyToGeoc (DtTime curTime = DtGetSimTime()) const <i>Get the dead-reckoned body to geocentric coordinate matrix</i>
virtual void	setAlgorithm (DtDeadReckonTypes alg) <i>Set/Get the dead reckoning algorithm used for generating the approximated values from the absolute values</i>
virtual DtDeadReckonTypes	algorithm () const
static void	deadReckonPosition (DtTime dt, DtDeadReckonTypes algorithm, DtConstVector oldPos, DtConstVector oldVel, DtConstVector oldAccel, DtVectorRef newVel, DtVectorRef newPos) <i>Calculate the dead reckon position information</i>
static void	deadReckonOrientation (DtTime dt, DtDeadReckonTypes algorithm, DtConstDcm old_b2w, const DtDirectedRotRate & drr, DtDcmRef new_b2w) <i>Calculate the orientation matrix</i>
inline DtConstVector	acceleration (DtTime curTime) const
inline DtDeadReckonTypes	algorithm () const
inline DtConstDcm	approxBodyToGeoc (DtTime curTime) const
inline const DtTaitBryan &	approxEuler (DtTime curTime) const
inline DtConstVector	approxLocation (DtTime curTime) const
inline DtConstVector	approxVelocity (DtTime curTime) const
inline DtConstDcm	deadReckonedBodyToGeoc (DtTime curTime) const
inline const DtTaitBryan &	deadReckonedEuler (DtTime curTime) const
inline DtConstVector	deadReckonedLocation (DtTime curTime) const
inline DtConstVector	deadReckonedVelocity (DtTime curTime) const
inline DtConstVector	rotationalVelocity (DtTime curTime) const
inline DtDeadReckoner *	self () const
inline void	setAlgorithm (DtDeadReckonTypes alg)

Protected Fields

DtVector	myLocAtTimeOfValidity
DtVector	myVelAtTimeOfValidity
DtTime	mySimTimePosDataValid
DtDcm	myBodyToGeocAtTimeOfValidity
DtTime	mySimTimeOrientDataValid
DtVector	myDRLoc

DtVector	myDRVel
DtTime	myDRPosTime
DtDcm	myDRBodyToGeoc
DtTime	myDRBodyToGeocTime
DtTaitBryan	myDREuler
DtTime	myDREulerTime
DtDirectedRotRate	myDirectedRotRate
DtDeadReckonTypes	myAlgorithm
DtVector	myAccel
DtBodyRates	myRotVel

Protected Methods

virtual void	deadReckonLoc (DtTime curTime) const <i>If the last dead reckoned position is not valid for the specified time, recalculate it and same the validity time</i>
virtual void	deadReckonBodyToGeoc (DtTime curTime) const <i>If the last dead reckoned body to geocentri coordinate matrix is not valid for the specified time, recalculate it and same the validity time</i>
virtual void	deadReckonEuler (DtTime curTime) const <i>If the last dead reckoned orientation (euler angles) is not valid for the specified time, recalculate it and same the validity time</i>
DtDeadReckoner*	self () const <i>Get rid of constness on the "this" pointer in order to access non-const functions from const member functions (such as simple accessors)</i>
static int	isRotational (DtDeadReckonTypes algorithm) <i>Returns 1 if the dead reckoning algorithm is a rotational one, 0 otherwise</i>

Inherited from [DtApproximator](#):

Protected



Documentation

class DtDeadReckoner:

Instances of DtDeadReckoner are used to provide approximated position and orientation information for entities using various dead reckoning algorithms. Position and orientation information is approximated over time from the last set absolute (non-approximated) information. Functions that start with deadReckoned return dead-reckoned values. Functions that start with approx return approximated values - which default to the dead-reckoned values, but this might be overridden in classes derived from DtDeadReckoner.

- DtDeadReckoner()
 default constructor
- virtual ~DtDeadReckoner()
 destructor
- DtDeadReckoner(const DtDeadReckoner& orig)
 copy constructor
- DtDeadReckoner& operator=(const DtDeadReckoner& orig)
 assignment operator
- virtual DtApproximator* clone() const
 Create a copy of this object. Memory is allocated and it is the responsibility of the caller to free the memory via delete.
- virtual void setLocation(DtConstVector loc, DtTime curTime = DtGetSimTime())
 Set absolute location
- virtual DtConstVector approxLocation(DtTime curTime = DtGetSimTime()) const
 Get the approximated (i.e. dead-reckoned for base DtDeadReckoner) location at the specified time.
- virtual DtConstVector deadReckonedLocation(DtTime curTime = DtGetSimTime()) const
 Get the dead-reckoned location at the specified time
- virtual void setVelocity(DtConstVector vel, DtTime curTime = DtGetSimTime())
 Set absolute velocity
- virtual DtConstVector approxVelocity(DtTime curTime = DtGetSimTime()) const
 Get the approximated (i.e. dead-reckoned for base DtDeadReckoner) velocity at the specified time.
- virtual DtConstVector deadReckonedVelocity(DtTime curTime = DtGetSimTime()) const
 Get the dead-reckoned velocity at the specified time
- virtual void setOrientation(const DtTaitBryan& ori, DtTime curTime = DtGetSimTime())
 Set absolute orientation
- virtual const DtTaitBryan& approxEuler(DtTime curTime = DtGetSimTime()) const
 Get the approximated (i.e. dead-reckoned for base DtDeadReckoner) orientation at the specified time.
- virtual const DtTaitBryan& deadReckonedEuler(DtTime curTime = DtGetSimTime()) const
 Get dead-reckoned orientation (euler angles) at the specified time
- virtual void setAcceleration(DtConstVector acc, DtTime curTime = DtGetSimTime())
 Set/Get absolute acceleration
- virtual DtConstVector acceleration(DtTime curTime = DtGetSimTime()) const
- virtual void setRotationalVelocity(DtConstVector vel, DtTime curTime = DtGetSimTime())
 Set/Get absolute rotational velocity
- virtual DtConstVector rotationalVelocity(DtTime curTime = DtGetSimTime()) const
- virtual DtConstDcm approxBodyToGeoc(DtTime curTime = DtGetSimTime()) const

Get the approximated (i.e. dead-reckoned for base DtDeadReckoner) body to geocentric coordinate matrix.

virtual DtConstDcm deadReckonedBodyToGeoc(DtTime curTime = DtGetSimTime()) const

Get the dead-reckoned body to geocentric coordinate matrix

virtual void setAlgorithm(DtDeadReckonTypes alg)

Set/Get the dead reckoning algorithm used for generating the approximated values from the absolute values

virtual DtDeadReckonTypes algorithm() const

static void deadReckonPosition(DtTime dt, DtDeadReckonTypes algorithm, DtConstVector oldPos, DtConstVector oldVel, DtConstVector oldAccel, DtVectorRef newVel, DtVectorRef newPos)

Calculate the dead reckon position information

static void deadReckonOrientation(DtTime dt, DtDeadReckonTypes algorithm, DtConstDcm old_b2w, const DtDirectedRotRate& drr, DtDcmRef new_b2w)

Calculate the orientation matrix

virtual void deadReckonLoc(DtTime curTime) const

If the last dead reckoned position is not valid for the specified time, recalculate it and same the validity time

virtual void deadReckonBodyToGeoc(DtTime curTime) const

If the last dead reckoned body to geocentri coordinate matrix is not valid for the specified time, recalculate it and same the validity time

virtual void deadReckonEuler(DtTime curTime) const

If the last dead reckoned orientation (euler angles) is not valid for the specified time, recalculate it and same the validity time

DtDeadReckoner* self() const

Get rid of constness on the "this" pointer in order to access non-const functions from const member functions (such as simple accessors). This should be used sparingly since it bypasses the const protection. Note: This is essentially like an operator and is non-virtual since can't override based on return type. Allows derived classes to also define this member.

static int isRotational(DtDeadReckonTypes algorithm)

Returns 1 if the dead reckoning algorithm is a rotational one, 0 otherwise

DtVector myLocAtTimeOfValidity

DtVector myVelAtTimeOfValidity

DtTime mySimTimePosDataValid

DtDcm myBodyToGeocAtTimeOfValidity

DtTime mySimTimeOrientDataValid

DtVector myDRLoc

DtVector myDRVel

DtTime myDRPosTime

DtDcm myDRBodyToGeoc

`DtTime myDRBodyToGeocTime`
`DtTaitBryan myDREuler`
`DtTime myDREulerTime`
`DtDirectedRotRate myDirectedRotRate`
`DtDeadReckonTypes myAlgorithm`
`DtVector myAccel`
`DtBodyRates myRotVel`
`inline DtConstVector acceleration(DtTime curTime) const`
`inline DtDeadReckonTypes algorithm() const`
`inline DtConstDcm approxBodyToGeoc(DtTime curTime) const`
`inline const DtTaitBryan& approxEuler(DtTime curTime) const`
`inline DtConstVector approxLocation(DtTime curTime) const`
`inline DtConstVector approxVelocity(DtTime curTime) const`
`inline DtConstDcm deadReckonedBodyToGeoc(DtTime curTime) const`
`inline const DtTaitBryan& deadReckonedEuler( DtTime curTime) const`
`inline DtConstVector deadReckonedLocation(DtTime curTime) const`
`inline DtConstVector deadReckonedVelocity(DtTime curTime) const`
`inline DtConstVector rotationalVelocity(DtTime curTime) const`
`inline DtDeadReckoner* self() const`
`inline void setAlgorithm(DtDeadReckonTypes alg)`

Direct child classes:

[DtSmoother](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file directrotate.h:

class DtDirectedRotRate

rotation rate given by single rate and axis of rotation

Public Fields

- double **thetaDot**
- double **n [3]**
radians per sec
-
- axis unit vector*

Public Methods

- DtDirectedRotRate ()**
 - DtDirectedRotRate** (DtConstVector pqr)
 - void **setFromVec** (DtConstVector pqr)
 - DtDirectedRotRate ()**
Inline functions
 - DtDirectedRotRate** (DtConstVector pqr)
 - inline void **setFromVec** (DtConstVector pqr)
-

Documentation

rotation rate given by single rate and axis of rotation

- DtDirectedRotRate ()**
- DtDirectedRotRate** (DtConstVector pqr)
- void **setFromVec** (DtConstVector pqr)
- double **thetaDot**
- double **n [3]**
radians per sec
-
- axis unit vector*
-
- inline DtDirectedRotRate ()**
Inline functions

- `inline DtDirectedRotRate(DtConstVector pqr)`
- `inline void setFromVec(DtConstVector pqr)`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file dcm.h:

class DtDcm

Public Methods

	DtDcm ()
	DtDcm (double a, double b, double c, double d, double e, double f, double g, double h, double i)
	DtDcm (DtConstVector a, DtConstVector b, DtConstVector c)
	DtDcm (const DtDcm &mat)
 DtDcm &	operator= (const DtDcm &mat)
	operator DtVector* ()
	operator const DtVector* () const
 DtVector &	operator[] (int i)
 const DtVector &	operator[] (int i) const
 void	print () const <i>Prints the data in the matrix</i>
 static DtConstDcm	zero ()
 static DtConstDcm	identity ()
 inline	DtDcm ()
 inline	DtDcm (double a, double b, double c, double d, double e, double f, double g, double h, double i)
 inline	DtDcm (DtConstVector a, DtConstVector b, DtConstVector c)
 inline	DtDcm (const DtDcm &mat)
 inline	operator DtVector* ()
 inline	operator const DtVector* () const
 inline DtDcm &	operator= (const DtDcm &mat)
 inline DtVector &	operator[] (int i) <i>Inline functions</i>
 inline const DtVector &	operator[] (int i) const

Protected Fields

 [DtVector](#) rep [3]

Documentation

- `DtDcm()`
- `DtDcm(double a, double b, double c, double d, double e, double f, double g, double h, double i)`
- `DtDcm(DtConstVector a, DtConstVector b, DtConstVector c)`
- `DtDcm(const DtDcm &mat)`
- `DtDcm& operator=(const DtDcm &mat)`
- `operator DtVector* ()`
- `operator const DtVector* () const`
- `DtVector& operator[](int i)`
- `const DtVector& operator[](int i) const`
- `void print() const`
Prints the data in the matrix
- `static DtConstDcm zero()`
- `static DtConstDcm identity()`
- `DtVector rep[3]`
- `inline DtDcm()`
- `inline DtDcm(double a, double b, double c, double d, double e, double f, double g, double h, double i)`
- `inline DtDcm(DtConstVector a, DtConstVector b, DtConstVector c)`
- `inline DtDcm(const DtDcm &mat)`
- `inline operator DtVector* ()`
- `inline operator const DtVector* () const`
- `inline DtDcm& operator=(const DtDcm &mat)`
- `inline DtVector& operator[](int i)`
Inline functions
- `inline const DtVector& operator[](int i) const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file smoother.h:

class DtSmoother: public DtDeadReckoner

class DtSmoother: Instances of DtSmoother are used to provide approximated position and orientation information for entities

Inheritance:

DtSmoother - DtDeadReckoner - DtApproximator

Public Fields

• static DtTime **theDfltSmoothPeriod**

Public Methods

•	DtSmoother () <i>constructor</i>
• virtual	~DtSmoother () <i>destructor</i>
•	DtSmoother (const DtSmoother & orig) <i>copy constructor</i>
• DtSmoother &	operator= (const DtSmoother & orig) <i>assignment operator</i>
• virtual DtApproximator *	clone () const <i>Create a copy of this object</i>
• virtual void	setLocation (DtConstVector loc, DtTime t = DtGetSimTime()) <i>Set absolute location</i>
• virtual DtConstVector	approxLocation (DtTime t = DtGetSimTime()) const <i>Get the approximated location at the specified time</i>
• virtual void	setVelocity (DtConstVector vel, DtTime t = DtGetSimTime()) <i>Set absolute velocity</i>
• virtual DtConstVector	approxVelocity (DtTime t = DtGetSimTime()) const <i>Get the approximated velocity at the specified time</i>
• virtual void	setOrientation (const DtTaitBryan & ori, DtTime t = DtGetSimTime()) <i>Set absolute orientation</i>
• virtual const DtTaitBryan &	approxEuler (DtTime t = DtGetSimTime()) const <i>Get approximated orientation (euler angles) at the specified time</i>
• virtual void	setAcceleration (DtConstVector acc, DtTime t = DtGetSimTime()) <i>Set/Get absolute acceleration</i>

virtual DtConstVector	acceleration (DtTime t = DtGetSimTime()) const
virtual void	setRotationalVelocity (DtConstVector vel, DtTime t = DtGetSimTime()) <i>Set/Get absolute rotational velocity</i>
virtual DtConstVector	rotationalVelocity (DtTime t = DtGetSimTime()) const
virtual DtConstDcm	approxBodyToGeoc (DtTime t = DtGetSimTime()) const <i>Get the approximated data body to geocentric coordinate matrix</i>
virtual void	setSmoothEnable (DtBoolean on) <i>Set/Get the individual smoothing switch</i>
virtual DtBoolean	smoothEnable () const
virtual void	setSmoothPeriod (DtTime smoothPeriod) <i>Set/Get smoothing period - the amount of time over which smoothing takes place</i>
virtual DtTime	smoothPeriod () const
static void	setDefaultSmoothPeriod (DtTime smoothPeriod) <i>Set/Get the default smoothing period</i>
static DtTime	defaultSmoothPeriod ()
static void	setMasterSmoothEnable (DtBoolean onOff) <i>Set/Get the value of the "master" smoothing-enabled switch</i>
static DtBoolean	masterSmoothEnable ()
inline DtSmoother *	self () const

Protected Fields

- [DtSmoothPosWithDr](#) mySpos
- [DtSmoothOriWithDr](#) mySorient

Protected Methods

- virtual void **feedSmoothing** ()
provide inputs to the actual smoothing algorithm
- [DtSmoother](#)* **self** () const
Get rid of constness on the "this" pointer in order to access non-const functions from const member functions (such as simple accessors)

Inherited from [DtDeadReckoner](#):

Public Methods

- virtual DtConstVector **deadReckonedLocation**(DtTime curTime = DtGetSimTime()) const
- virtual DtConstVector **deadReckonedVelocity**(DtTime curTime = DtGetSimTime()) const
- virtual const [DtTaitBryan](#)& **deadReckonedEuler**(DtTime curTime = DtGetSimTime()) const
- virtual DtConstDcm **deadReckonedBodyToGeoc**(DtTime curTime = DtGetSimTime()) const

- virtual void **setAlgorithm**([DtDeadReckonTypes](#) alg)
- virtual [DtDeadReckonTypes](#) **algorithm**() const
- static void **deadReckonPosition**(DtTime dt, [DtDeadReckonTypes](#) algorithm, DtConstVector oldPos, DtConstVector oldVel, DtConstVector oldAccel, DtVectorRef newVel, DtVectorRef newPos)
- static void **deadReckonOrientation**(DtTime dt, [DtDeadReckonTypes](#) algorithm, DtConstDcm old_b2w, const [DtDirectedRotRate](#)& drr, DtDcmRef new_b2w)
- inline [DtDeadReckonTypes](#) **algorithm**() const
- inline DtConstDcm **deadReckonedBodyToGeoc**(DtTime curTime) const
- inline const [DtTaitBryan](#)& **deadReckonedEuler**(DtTime curTime) const
- inline DtConstVector **deadReckonedLocation**(DtTime curTime) const
- inline DtConstVector **deadReckonedVelocity**(DtTime curTime) const
- inline void **setAlgorithm**([DtDeadReckonTypes](#) alg)

Protected Fields

- [DtVector](#) **myLocAtTimeOfValidity**
- [DtVector](#) **myVelAtTimeOfValidity**
- DtTime **mySimTimePosDataValid**
- [DtDcm](#) **myBodyToGeocAtTimeOfValidity**
- DtTime **mySimTimeOrientDataValid**
- [DtVector](#) **myDRLoc**
- [DtVector](#) **myDRVel**
- DtTime **myDRPosTime**
- [DtDcm](#) **myDRBodyToGeoc**
- DtTime **myDRBodyToGeocTime**
- [DtTaitBryan](#) **myDREuler**
- DtTime **myDREulerTime**
- [DtDirectedRotRate](#) **myDirectedRotRate**
- [DtDeadReckonTypes](#) **myAlgorithm**
- [DtVector](#) **myAccel**
- DtBodyRates **myRotVel**

Protected Methods

- virtual void **deadReckonLoc**(DtTime curTime) const
 - virtual void **deadReckonBodyToGeoc**(DtTime curTime) const
 - virtual void **deadReckonEuler**(DtTime curTime) const
 - static int **isRotational**([DtDeadReckonTypes](#) algorithm)
-

Inherited from [DtApproximator](#):

Protected



Documentation

class DtSmoother:

Instances of DtSmoother are used to provide approximated position and orientation information for entities. Position and orientation information is approximated over time from the last set absolute (non-approximated) information. When a new position is set, the current position moves smoothly to reflect the new information, rather than jumping to the new absolute position.

- DtSmoother()**
constructor
- virtual ~DtSmoother()**
destructor
- DtSmoother(const [DtSmoother](#)& orig)**
copy constructor
- [DtSmoother](#)& operator=(const [DtSmoother](#)& orig)**
assignment operator
- virtual [DtApproximator](#)* clone() const**
Create a copy of this object. Memory is allocated and it is the responsibility of the caller to free the memory via delete.
- virtual void setLocation(DtConstVector loc, DtTime t = DtGetSimTime())**
Set absolute location
- virtual DtConstVector approxLocation(DtTime t = DtGetSimTime()) const**
Get the approximated location at the specified time
- virtual void setVelocity(DtConstVector vel, DtTime t = DtGetSimTime())**
Set absolute velocity

- `virtual DtConstVector approxVelocity( DtTime t = DtGetSimTime()) const`
 Get the approximated velocity at the specified time
- `virtual void setOrientation(const DtTaitBryan& ori, DtTime t = DtGetSimTime())`
 Set absolute orientation
- `virtual const DtTaitBryan& approxEuler( DtTime t = DtGetSimTime()) const`
 Get approximated orientation (euler angles) at the specified time
- `virtual void setAcceleration(DtConstVector acc, DtTime t = DtGetSimTime())`
 Set/Get absolute acceleration
- `virtual DtConstVector acceleration( DtTime t = DtGetSimTime()) const`
- `virtual void setRotationalVelocity(DtConstVector vel, DtTime t = DtGetSimTime())`
 Set/Get absolute rotational velocity
- `virtual DtConstVector rotationalVelocity( DtTime t = DtGetSimTime()) const`
- `virtual DtConstDcm approxBodyToGeoc( DtTime t = DtGetSimTime()) const`
 Get the approximated data body to geocentric coordinate matrix
- `virtual void setSmoothEnable(DtBoolean on)`
 Set/Get the individual smoothing switch. In order for smoothing to be enabled, the "master" smoothing-enabled switch must be on, AND the individual smoothing switch (accessed through these functions), must be on. Individual smoothing switch defaults to on.
- `virtual DtBoolean smoothEnable() const`
- `virtual void setSmoothPeriod(DtTime smoothPeriod)`
 Set/Get smoothing period - the amount of time over which smoothing takes place. Pass a value of -1 to indicate use of the default smoothing period.
- `virtual DtTime smoothPeriod() const`
- `static void setDfltSmoothPeriod(DtTime smoothPeriod)`
 Set/Get the default smoothing period
- `static DtTime dfltSmoothPeriod()`
- `static void setMasterSmoothEnable(DtBoolean onOff)`
 Set/Get the value of the "master" smoothing-enabled switch. Master smoothing switch defaults to on.
- `static DtBoolean masterSmoothEnable()`
- `static DtTime theDfltSmoothPeriod`
- `virtual void feedSmoothing()`
 provide inputs to the actual smoothing algorithm
- `DtSmoother* self() const`
 Get rid of constness on the "this" pointer in order to access non-const functions from const member functions (such as simple accessors). This should be used sparingly since it bypasses the const protection. Note: This is essentially like an operator and is non-virtual since can't override based on return type. Allows derived classes to also define this member.

- [DtSmoothPosWithDr](#) mySpos
- [DtSmoothOriWithDr](#) mySorient
- inline [DtSmooother](#)* self() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file smoothposwdr.h:

class DtSmoothPosWithDr: public DtSmoothPos

class DtSmoothPosWithDr: Instances of DtSmoothPosWithDr are

Inheritance:

DtSmoothPosWithDr - DtSmoothPos

Public Methods

- **DtSmoothPosWithDr** ([DtDeadReckoner](#)* dr, DtTime smoothT)
default constructor
- virtual **~DtSmoothPosWithDr** ()
destructor

Protected Methods

- **DtSmoothPosWithDr** (const [DtSmoothPosWithDr](#)& orig)
copy constructor
 - [DtSmoothPosWithDr](#)& **operator=** (const [DtSmoothPosWithDr](#)& orig)
assignment operator
 - virtual void **dfltAction** (DtTime simTime, void* usr, DtVectorRef vs, DtVectorRef ps)
The default action is to dead-reckon
-

Inherited from DtSmoothPos:

Public Methods

- void **input**(DtConstVector pos, DtConstVector vel, DtConstVector acc, DtTime simTime, int freeze=0)
- void **tick**(DtTime simTime)
- DtConstVector **position**() const
- DtConstVector **velocity**() const
- virtual void **setSmoothEnable**(int on)
- virtual int **smoothEnable**() const

- virtual void **setSmoothPeriod**(DtTime t)
- virtual DtTime **smoothPeriod**() const
- static void **setDfltSmoothPeriod**(DtTime t)
- static DtTime **dfltSmoothPeriod**()
- static void **setMasterSmoothEnable**([DtBoolean](#) onOff)
- static [DtBoolean](#) **masterSmoothEnable**()
- inline DtTime **smoothPeriod**() const

Protected Fields

- int **frozen**
- DtTime **t0**
- DtTime **t1**
- DtTime **t2**
- DtTime **lastT**
- DtTime **myLocalTau**
- DtTime **myLocalHtau**
- DtTime **myLocalTauInv**
- DtTime* **myTau**
- DtTime* **myHtau**
- DtTime* **myTauInv**
- [DtVector](#) **ps**
- [DtVector](#) **vs**
- [DtVector](#) **p0**
- [DtVector](#) **v0**
- [DtVector](#) **dp**
- [DtVector](#) **dv**
- [DtVector](#) **p1**
- [DtVector](#) **v1**
- [DtVector](#) **a1**
- [DtVector](#) **a2**
- [DtVector](#) **ap**

- `int myEnable`
- `int needT1data`
- `int needT2data`
- `void* usr`
- `static DtTime theDfltTau`
- `static DtTime theDfltHtau`
- `static DtTime theDfltTaulnv`
- `static DtBoolean theMasterSmoothEnable`

Protected Methods

- `void init(void *u, DtTime t)`
- `void setT1data()`

Documentation

class DtSmoothPosWithDr: Instances of DtSmoothPosWithDr are

- `DtSmoothPosWithDr(DtDeadReckoner* dr, DtTime smoothT)`
default constructor
- `virtual ~DtSmoothPosWithDr()`
destructor
- `DtSmoothPosWithDr(const DtSmoothPosWithDr& orig)`
copy constructor
- `DtSmoothPosWithDr& operator=(const DtSmoothPosWithDr& orig)`
assignment operator
- `virtual void dfltAction(DtTime simTime, void* usr, DtVectorRef vs, DtVectorRef ps)`
The default action is to dead-reckon

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file smoothutil.h:

class DtSmoothPos

----- POSITION -----

Inheritance:

DtSmoothPos

Public Methods

	DtSmoothPos (void *usr, DtTime smoothT, DtConstVector pos, DtConstVector vel, DtConstVector acc)
	<i>Constructors</i>
	DtSmoothPos (void *usr, DtTime smoothT) <i>WARNING: not fully constructed until input() & tick() are called</i>
	~DtSmoothPos ()
virtual	input (DtConstVector pos, DtConstVector vel, DtConstVector acc, DtTime simTime, int freeze=0)
void	tick (DtTime simTime)
DtConstVector	position () const
DtConstVector	velocity () const
virtual void	setSmoothEnable (int on) <i>Set/Get the individual smoothing switch</i>
virtual int	smoothEnable () const
virtual void	setSmoothPeriod (DtTime t) <i>Set/Get smoothPeriod</i>
virtual DtTime	smoothPeriod () const
static void	setDefaultSmoothPeriod (DtTime t) <i>Set/Get default smooth period</i>
static DtTime	defaultSmoothPeriod ()
static void	setMasterSmoothEnable (DtBoolean onOff) <i>Set/Get the value of the "master" smoothing-enabled switch</i>
static DtBoolean	masterSmoothEnable ()
inline	DtSmoothPos (const DtSmoothPos &)
inline DtTime	smoothPeriod () const

Protected Fields

int	frozen
DtTime	t0
DtTime	t1
DtTime	t2
DtTime	lastT
DtTime	myLocalTau
DtTime	myLocalHtau
DtTime	myLocalTauInv
DtTime*	myTau
DtTime*	myHtau
DtTime*	myTauInv
DtVector	ps
DtVector	vs
DtVector	p0
DtVector	v0
DtVector	dp
DtVector	dv
DtVector	p1
DtVector	v1
DtVector	a1
DtVector	a2
DtVector	ap
int	myEnable
int	needT1data
int	needT2data
void*	usr
static DtTime	theDfltTau
static DtTime	theDfltHtau
static DtTime	theDfltTauInv
static DtBoolean	theMasterSmoothEnable

Protected Methods

DtSmoothPos (const [DtSmoothPos](#)&)

virtual void **dfltAction** (DtTime simTime, void *usr, DtVectorRef vs, DtVectorRef ps)

void **init** (void *u, DtTime t)

void **setT1data** ()

Documentation

----- POSITION -----

DtSmoothPos(void *usr, DtTime smoothT, DtConstVector pos, DtConstVector vel, DtConstVector acc)

Constructors. smoothT is the initial smoothPeriod. Use a value of -1 to indicate use of the default period.

DtSmoothPos(void *usr, DtTime smoothT)

WARNING: not fully constructed until input() & tick() are called

virtual ~DtSmoothPos()

void input(DtConstVector pos, DtConstVector vel, DtConstVector acc, DtTime simTime, int freeze=0)

void tick(DtTime simTime)

DtConstVector position() const

DtConstVector velocity() const

virtual void setSmoothEnable(int on)

Set/Get the individual smoothing switch. In order for smoothing to be enabled, the "master" smoothing-enabled switch must be on, AND the individual smoothing switch (accessed through these functions), must be on.

virtual int smoothEnable() const

virtual void setSmoothPeriod(DtTime t)

Set/Get smoothPeriod. If set to -1, smooth period reverts to the current default smooth period.

virtual DtTime smoothPeriod() const

static void setDfltSmoothPeriod(DtTime t)

Set/Get default smooth period

static DtTime dfltSmoothPeriod()

static void setMasterSmoothEnable([DtBoolean](#) onOff)

Set/Get the value of the "master" smoothing-enabled switch

static [DtBoolean](#) masterSmoothEnable()

DtSmoothPos(const [DtSmoothPos](#)&)

virtual void dfltAction(DtTime simTime, void *usr, DtVectorRef vs, DtVectorRef ps)

```
void init(void *u, DtTime t)

void setT1data()

int frozen

DtTime t0

DtTime t1

DtTime t2

DtTime lastT

DtTime myLocalTau

DtTime myLocalHtau

DtTime myLocalTauInv

DtTime* myTau

DtTime* myHtau

DtTime* myTauInv

DtVector ps

DtVector vs

DtVector p0

DtVector v0

DtVector dp

DtVector dv

DtVector p1

DtVector v1

DtVector a1

DtVector a2

DtVector ap

int myEnable

int needT1data

int needT2data

void* usr

static DtTime theDfltTau

static DtTime theDfltHtau

static DtTime theDfltTauInv
```

- static [DtBoolean](#) theMasterSmoothEnable
- inline DtSmoothPos(const [DtSmoothPos](#)&)
- inline DtTime smoothPeriod() const

Direct child classes:

[DtSmoothPosWithDr](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file smoothoriwdr.h:

class DtSmoothOriWithDr: public DtSmoothOrient

class DtSmoothOriWithDr: Instances of DtSmoothOriWithDr are

Inheritance:

DtSmoothOriWithDr - DtSmoothOrient

Public Methods

- **DtSmoothOriWithDr** ([DtDeadReckoner](#)* dr, DtTime smoothT)
default constructor
- virtual **~DtSmoothOriWithDr** ()
destructor

Protected Methods

- **DtSmoothOriWithDr** (const [DtSmoothOriWithDr](#)& orig)
copy constructor
 - [DtSmoothOriWithDr](#)& **operator=** (const [DtSmoothOriWithDr](#)& orig)
assignment operator
 - virtual void **dfltAction** (DtTime simTime, void *usr, DtDcmRef os)
The default action is to dead-reckon
-

Inherited from DtSmoothOrient:

Public Methods

- void **input**(DtConstDcm orient, const [DtDirectedRotRate](#) &dr, DtTime simTime, int freeze=0)
- void **tick**(DtTime simTime)
- DtConstDcm **orientation**() const
- virtual void **setSmoothEnable**(int on)
- virtual int **smoothEnable**() const
- virtual void **setSmoothPeriod**(DtTime t)

- virtual DtTime **smoothPeriod()** const
- static void **setDfltSmoothPeriod**(DtTime t)
- static DtTime **dfltSmoothPeriod()**
- static void **setMasterSmoothEnable**([DtBoolean](#) onOff)
- static [DtBoolean](#) **masterSmoothEnable()**
- inline DtTime **smoothPeriod()** const

Protected Fields

- int **frozen**
- DtTime **t0**
- DtTime **t2**
- DtTime **lastT**
- DtTime* **myTau**
- DtTime **myLocalTau**
- [DtDcm](#) **os**
- [DtDcm](#) **o0**
- [DtVector](#) **ws**
- int **myEnable**
- void* **usr**
- static DtTime **theDfltTau**
- static [DtBoolean](#) **theMasterSmoothEnable**

Protected Methods

- void **init**(void *u, DtTime t)
- void **setT1data()**

Documentation

class DtSmoothOriWithDr: Instances of DtSmoothOriWithDr are

- DtSmoothOriWithDr([DtDeadReckoner](#)* dr, DtTime smoothT)
default constructor
- virtual ~DtSmoothOriWithDr()

destructor

- `DtSmoothOriWithDr(const DtSmoothOriWithDr& orig)`

copy constructor

- `DtSmoothOriWithDr& operator=(const DtSmoothOriWithDr& orig)`

assignment operator

- `virtual void dfltAction(DtTime simTime, void *usr, DtDcmRef os)`

The default action is to dead-reckon

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file smoothutil.h:

class DtSmoothOrient

----- ORIENTATION -----

Inheritance:

DtSmoothOrient

Public Methods

- **DtSmoothOrient** (void *usr, DtTime smoothT, DtConstDcm orient, const [DtDirectedRotRate](#) &drr)
Constructors
- **DtSmoothOrient** (void *usr, DtTime smoothT)
WARNING: not fully constructed until input() & tick() are called
- void **input** (DtConstDcm orient, const [DtDirectedRotRate](#) &drr, DtTime simTime, int freeze=0)
- void **tick** (DtTime simTime)
- DtConstDcm **orientation** () const
- virtual void **setSmoothEnable** (int on)
Set/Get the individual smoothing switch
- virtual int **smoothEnable** () const
- virtual void **setSmoothPeriod** (DtTime t)
Set/Get smoothPeriod
- virtual DtTime **smoothPeriod** () const
- static void **setDefaultSmoothPeriod** (DtTime t)
Set/Get default smooth period
- static DtTime **defaultSmoothPeriod** ()
- static void **setMasterSmoothEnable** ([DtBoolean](#) onOff)
Set/Get the value of the "master" smoothing-enabled switch
- static [DtBoolean](#) **masterSmoothEnable** ()
- inline DtTime **smoothPeriod** () const

Protected Fields

- int **frozen**
- DtTime **t0**
- DtTime **t2**

- `DtTime` `lastT`
- `DtTime*` `myTau`
- `DtTime` `myLocalTau`
- [`DtDcm`](#) `os`
- [`DtDcm`](#) `o0`
- [`DtVector`](#) `ws`
- `int` `myEnable`
- `void*` `usr`
- `static DtTime` `theDfltTau`
- `static DtBoolean` `theMasterSmoothEnable`

Protected Methods

- virtual void **`dfltAction`** (`DtTime simTime`, `void *usr`, `DtDcmRef os`)
- void **`init`** (`void *u`, `DtTime t`)
- void **`setT1data`** ()

Documentation

----- ORIENTATION -----

- `DtSmoothOrient(void *usr, DtTime smoothT, DtConstDcm orient, const DtDirectedRotRate &drr)`
Constructors. `smoothT` is the initial `smoothPeriod`. Use a value of -1 to indicate use of the default period.
- `DtSmoothOrient(void *usr, DtTime smoothT)`
WARNING: not fully constructed until `input()` & `tick()` are called
- `void input(DtConstDcm orient, const DtDirectedRotRate &drr, DtTime simTime, int freeze=0)`
- `void tick(DtTime simTime)`
- `DtConstDcm orientation() const`
- `virtual void setSmoothEnable(int on)`
Set/Get the individual smoothing switch. In order for smoothing to be enabled, the "master" smoothing-enabled switch must be on, AND the individual smoothing switch (accessed through these functions), must be on.
- `virtual int smoothEnable() const`
- `virtual void setSmoothPeriod(DtTime t)`
Set/Get `smoothPeriod`. If set to -1, smooth period reverts to the current default smooth period.

- `virtual DtTime smoothPeriod() const`
- `static void setDfltSmoothPeriod(DtTime t)`
Set/Get default smooth period
- `static DtTime dfltSmoothPeriod()`
- `static void setMasterSmoothEnable(DtBoolean onOff)`
Set/Get the value of the "master" smoothing-enabled switch
- `static DtBoolean masterSmoothEnable()`
- `virtual void dfltAction(DtTime simTime, void *usr, DtDcmRef os)`
- `void init(void *u, DtTime t)`
- `void setTldata()`
- `int frozen`
- `DtTime t0`
- `DtTime t2`
- `DtTime lastT`
- `DtTime* myTau`
- `DtTime myLocalTau`
- `DtDcm os`
- `DtDcm o0`
- `DtVector ws`
- `int myEnable`
- `void* usr`
- `static DtTime theDfltTau`
- `static DtBoolean theMasterSmoothEnable`
- `inline DtTime smoothPeriod() const`

Direct child classes:

[DtSmoothOriWithDr](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file threshold.h:

class DtThresholder

class DtThresholder: Instances of DtThresholder store thresholding information used to determine what attributes of an entity need to be updated

Public Methods

	DtThresholder () <i>default constructor</i>
virtual	~DtThresholder () <i>destructor</i>
	DtThresholder (const DtThresholder & orig) <i>copy constructor</i>
DtThresholder &	operator= (const DtThresholder & orig) <i>assignment operator</i>
virtual void	setRotationThreshold (DtReal thresh) <i>Set/Get rotation threshold in radians</i>
virtual double	rotationThreshold () const
virtual double	rotationTraceThreshold () const <i>Returns the "RotationTrace" threshold</i>
virtual void	setTranslationThreshold (DtReal thresh) <i>Set/Get translation threshold in meters</i>
virtual double	translationThreshold () const
virtual void	setAggDimensionThreshold (DtReal thresh) <i>Set/Get aggregate dimension threshold</i>
virtual double	aggDimensionThreshold () const
virtual DtArtPartThresholder *	artPartThresholder () <i>Returns a pointer to an object that maintains thresholds for articulated parts</i>
virtual const DtArtPartThresholder &	artPartThresholder () const
static void	setDefaultTranslationThreshold (DtReal thresh) <i>Set/Get default translation threshold in radians</i>
static double	defaultTranslationThreshold ()
static void	setDefaultRotationThreshold (DtReal thresh) <i>Set/Get default rotation threshold in radians</i>
static double	defaultRotationThreshold ()
static double	defaultRotationTraceThreshold () <i>Returns the "RotationTrace" threshold</i>
static void	setDefaultAggDimensionThreshold (DtReal thresh) <i>Set/Get aggregate dimension threshold</i>
static double	defaultAggDimensionThreshold ()
inline DtArtPartThresholder *	artPartThresholder ()

- inline const [DtArtPartThresholder](#)& **artPartThresholder** () const
- inline double **dfltAggDimensionThreshold** ()
- inline double **dfltRotationThreshold** ()
- inline double **dfltRotationTraceThreshold** ()
- inline double **dfltTranslationThreshold** ()
- inline void **setDfltAggDimensionThreshold** (DtReal thresh)
- inline void **setDfltRotationThreshold** (DtReal thresh)
- inline void **setDfltTranslationThreshold** (DtReal thresh)

Protected Fields

- double **myTransThresh**
- double **myRotThresh**
- double **myTraceRotThresh**
- double **myAggDimThresh**
- [DtBoolean](#) **myUsingDfltTrans**
- [DtBoolean](#) **myUsingDfltRot**
- [DtBoolean](#) **myUsingDfltAggDim**
- [DtArtPartThresholder](#) **myArtPartThresh**
- static double **theDfltTransThresh**
- static double **theDfltRotThresh**
- static double **theDfltTraceRotThresh**
- static double **theDfltAggDimThresh**

Documentation

class DtThresholder: Instances of DtThresholder store thresholding information used to determine what attributes of an entity need to be updated

- DtThresholder** ()
default constructor
- virtual ~DtThresholder** ()
destructor
- DtThresholder** (const [DtThresholder](#)& orig)
copy constructor
- [DtThresholder](#)& **operator=** (const [DtThresholder](#)& orig)

assignment operator

virtual void setRotationThreshold(DtReal thresh)

Set/Get rotation threshold in radians. Negative number means use default.

virtual double rotationThreshold() const

virtual double rotationTraceThreshold() const

Returns the "RotationTrace" threshold

virtual void setTranslationThreshold(DtReal thresh)

Set/Get translation threshold in meters. Negative number means use default.

virtual double translationThreshold() const

virtual void setAggDimensionThreshold(DtReal thresh)

Set/Get aggregate dimension threshold. Thresh is a number between 0 and 1, indicating the percentage change that an aggregate dimension can undergo before causing an update to be generated. Negative number means use default. Applies only to aggregate entities.

virtual double aggDimensionThreshold() const

virtual [DtArtPartThresholder](#)* artPartThresholder()

Returns a pointer to an object that maintains thresholds for articulated parts

virtual const [DtArtPartThresholder](#)& artPartThresholder() const

static void setDfltTranslationThreshold(DtReal thresh)

Set/Get default translation threshold in radians.

static double dfltTranslationThreshold()

static void setDfltRotationThreshold(DtReal thresh)

Set/Get default rotation threshold in radians.

static double dfltRotationThreshold()

static double dfltRotationTraceThreshold()

Returns the "RotationTrace" threshold

static void setDfltAggDimensionThreshold(DtReal thresh)

Set/Get aggregate dimension threshold. Thresh is a number between 0 and 1, indicating the percentage change that an aggregate dimension can undergo before causing an update to be generated. Applies only to aggregate entities.

static double dfltAggDimensionThreshold()

double myTransThresh

double myRotThresh

double myTraceRotThresh

double myAggDimThresh

[DtBoolean](#) myUsingDfltTrans

[DtBoolean](#) myUsingDfltRot

```
DtBoolean myUsingDfltAggDim
DtArtPartThresholder myArtPartThresh
static double theDfltTransThresh
static double theDfltRotThresh
static double theDfltTraceRotThresh
static double theDfltAggDimThresh
inline DtArtPartThresholder* artPartThresholder()
inline const DtArtPartThresholder& artPartThresholder() const
inline double dfltAggDimensionThreshold()
inline double dfltRotationThreshold()
inline double dfltRotationTraceThreshold()
inline double dfltTranslationThreshold()
inline void setDfltAggDimensionThreshold(DtReal thresh)
inline void setDfltRotationThreshold(DtReal thresh)
inline void setDfltTranslationThreshold(DtReal thresh)
```

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file artpartthresh.h:

class DtArtPartThresholder

class DtArtPartThresholder: DtArtPartThresholder is used to store art-part-related thresholds, and to perform some art part thresholding

Public Classes

- enum DtArtPartThresholdType
 - DtArtPartDfltThreshMode
 - DtArtPartIndividualThresh
 - DtArtPartAggregateThresh

Public Methods

-
-
-
- [DtArtPartThresholder](#)&
- virtual void
- virtual double
- virtual double
- virtual void
- virtual double
- virtual void
- virtual double
- virtual void
- virtual [DtArtPartThresholdType](#)
- virtual [DtBoolean](#)
- DtArtPartThresholder** ()
Constructor
- ~DtArtPartThresholder** ()
Destructor
- DtArtPartThresholder** (const [DtArtPartThresholder](#)& orig)
copy constructor
- operator=** (const [DtArtPartThresholder](#)& orig)
assignment operator
- setRotationThreshold** (DtReal thresh)
Set/Get rotation threshold in radians
- rotationThreshold** () const
- rotationTraceThreshold** () const
Returns the "RotationTrace" threshold
- setTranslationThreshold** (DtReal thresh)
Set/Get translation threshold in meters
- translationThreshold** () const
- setFractionalThreshold** (DtReal thresh)
Set/Get fractional threshold (fraction between 0 and 1)
- fractionalThreshold** () const
- setThreshMode** ([DtArtPartThresholdType](#) mode)
Set/Get the threshhold mode
- threshMode** () const
- artPartsExceedAggThreshold** (const [DtArtPartList](#)& localList, const [DtArtPartList](#)& remoteList) const
Get whether or not an aggregate threshold (such as position, translation, rotation, etc

virtual [DtBoolean](#)

virtual float

static void

static [DtArtPartThresholdType](#)

static void

static DtFloat32

static void

static double

static void

static double

inline DtFloat32

inline double

inline [DtArtPartThresholder::DtArtPartThresholdType](#)

inline double

inline void

inline void

inline void

inline void

artPartsExceedIndivThreshold (const [DtArtPartList](#)& localList, const [DtArtPartList](#)& remoteList) const

Get whether or not an individual parameter threshold has been exceeded

artPartThreshold (int paramType) const

Get the thresholding value (functional, translational, or rotational) for the parameter type

setDfltThreshMode ([DtArtPartThresholdType](#) mode)

Set/Get the threshold mode

dfltThreshMode ()

setDfltFractionalThreshold (DtFloat32 thresh)

Set/Get fractional threshold value

dfltFractionalThreshold ()

setDfltRotationThreshold (DtReal r)

Set/Get rotational threshold value

dfltRotationThreshold ()

setDfltTranslationThreshold (DtReal t)

Set/Get translational threshold value

dfltTranslationThreshold ()

dfltFractionalThreshold ()

dfltRotationThreshold ()

dfltThreshMode ()

dfltTranslationThreshold ()

setDfltFractionalThreshold (DtFloat32 thresh)

setDfltRotationThreshold (DtReal thresh)

setDfltThreshMode ([DtArtPartThresholdType](#) mode)

setDfltTranslationThreshold (DtReal t)

Protected Fields

[DtArtPartThresholdType](#)

float

double

double

double

[DtBoolean](#)

[DtBoolean](#)

[DtBoolean](#)

[DtBoolean](#)

static [DtArtPartThresholdType](#) **theDfltThreshMode**

myThreshMode

myFracThresh

myTransThresh

myRotThresh

myTraceRotThresh

myUsingDfltMode

myUsingDfltTrans

myUsingDfltRot

myUsingDfltFrac

static float	theDfltFracThresh
static double	theDfltTransThresh
static double	theDfltRotThresh
static double	theDfltTraceRotThresh

Documentation

class DtArtPartThresholder: DtArtPartThresholder is used to store art-part-related thresholds, and to perform some art part thresholding

- enum DtArtPartThresholdType
- DtArtPartDfltThreshMode
- DtArtPartIndividualThresh
- DtArtPartAggregateThresh
- DtArtPartThresholder()
- Constructor
- ~DtArtPartThresholder()
- Destructor
- DtArtPartThresholder(const [DtArtPartThresholder](#)& orig)
- copy constructor
- [DtArtPartThresholder](#)& operator=(const [DtArtPartThresholder](#)& orig)
- assignment operator
- virtual void setRotationThreshold(DtReal thresh)
- Set/Get rotation threshold in radians. Negative number means use default.
- virtual double rotationThreshold() const
- virtual double rotationTraceThreshold() const
- Returns the "RotationTrace" threshold
- virtual void setTranslationThreshold(DtReal thresh)
- Set/Get translation threshold in meters. Negative number means use default.
- virtual double translationThreshold() const
- virtual void setFractionalThreshold(DtReal thresh)
- Set/Get fractional threshold (fraction between 0 and 1). Negative number means use default.
- virtual double fractionalThreshold() const
- virtual void setThreshMode([DtArtPartThresholdType](#) mode)
- Set/Get the theshold mode. Use DtArtPartDfltThreshMode to indicate use of the default.
- virtual [DtArtPartThresholdType](#) threshMode() const

virtual [DtBoolean](#) artPartsExceedAggThreshold(const [DtArtPartList](#)& localList, const [DtArtPartList](#)& remoteList) const

Get whether or not an aggregate threshold (such as position, translation, rotation, etc.) has been exceeded.

virtual [DtBoolean](#) artPartsExceedIndivThreshold(const [DtArtPartList](#)& localList, const [DtArtPartList](#)& remoteList) const

Get whether or not an individual parameter threshold has been exceeded

virtual float artPartThreshold(int paramType) const

Get the thresholding value (functional, translational, or rotational) for the parameter type

static void setDfltThreshMode([DtArtPartThresholdType](#) mode)

Set/Get the threshold mode

static [DtArtPartThresholdType](#) dfltThreshMode()

static void setDfltFractionalThreshold(DtFloat32 thresh)

Set/Get fractional threshold value

static DtFloat32 dfltFractionalThreshold()

static void setDfltRotationThreshold(DtReal r)

Set/Get rotational threshold value

static double dfltRotationThreshold()

static void setDfltTranslationThreshold(DtReal t)

Set/Get translational threshold value

static double dfltTranslationThreshold()

[DtArtPartThresholdType](#) myThreshMode

float myFracThresh

double myTransThresh

double myRotThresh

double myTraceRotThresh

[DtBoolean](#) myUsingDfltMode

[DtBoolean](#) myUsingDfltTrans

[DtBoolean](#) myUsingDfltRot

[DtBoolean](#) myUsingDfltFrac

static [DtArtPartThresholdType](#) theDfltThreshMode

static float theDfltFracThresh

static double theDfltTransThresh

static double theDfltRotThresh

```
static double theDfltTraceRotThresh  
inline DtFloat32 dfltFractionalThreshold()  
inline double dfltRotationThreshold()  
inline DtArtPartThresholder::DtArtPartThresholdType dfltThreshMode()  
inline double dfltTranslationThreshold()  
inline void setDfltFractionalThreshold( DtFloat32 thresh)  
inline void setDfltRotationThreshold(DtReal thresh)  
inline void setDfltThreshMode( DtArtPartThresholdType mode)  
inline void setDfltTranslationThreshold(DtReal t)
```

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitysr.h:

class DtEntityStateRepository: public DtBaseEntityStateRepository

class DtEntityStateRepository: Instances of DtEntityStateRepository are used to maintain entity state

Inheritance:

DtEntityStateRepository - DtBaseEntityStateRepository -
DtStateRepository

Public Methods

	DtEntityStateRepository () <i>default constructor</i>
virtual	~DtEntityStateRepository () <i>destructor</i>
	DtEntityStateRepository (const DtEntityStateRepository & orig) <i>copy constructor</i>
DtEntityStateRepository &	operator= (const DtEntityStateRepository & orig) <i>assignment operator</i>
virtual DtStateRepository *	clone () const <i>return an empty object of the same type as this</i>
virtual void	setForceId (DtForceType force) <i>Set/Get the force type</i>
virtual DtForceType	forceId () const
virtual void	setGuise (const DtEntityType & type) <i>Set/Get the alternate entity type or guise</i>
virtual const DtEntityType &	guise () const
virtual void	setFrozen (DtBoolean onOff) <i>Set/Get whether frozen or not</i>
virtual DtBoolean	frozen () const
virtual void	setMarkingCharSet (int charSet) <i>Set/Get the character set used for the marking</i>
virtual int	markingCharSet () const
virtual void	setMarkingText (const char* text) <i>Set/Get the marking text</i>
virtual const char*	markingText () const

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtDamageState](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtHatchState](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtLifeformState](#)

virtual void

virtual [DtLightState](#)

virtual void

virtual [DtDensityType](#)

virtual void

virtual [DtBoolean](#)

virtual void

virtual [DtBoolean](#)

setPowerPlantOn ([DtBoolean](#) onOff)

Set/Get whether entity's power plant is on or not

powerPlantOn () const

setDamageState ([DtDamageState](#) s)

Set/Get the damage state

damageState () const

setEngineSmokeOn ([DtBoolean](#) onOff)

Set/Get whether engine smoke is on or not

engineSmokeOn () const

setFlamesPresent ([DtBoolean](#) onOff)

Set/Get whether flames are present or not

flamesPresent () const

setHasFuelSupplyCap ([DtBoolean](#) onOff)

Set/Get whether entity has fuel supplying capability

hasFuelSupplyCap () const

setHasRecoveryCap ([DtBoolean](#) onOff)

Set/Get whether entity has recovery capability

hasRecoveryCap () const

setHasRepairCap ([DtBoolean](#) onOff)

Set/Get whether entity has repair capability

hasRepairCap () const

setHatchState ([DtHatchState](#) s)

Set/Get the entity's hatch state

hatchState () const

setImmobilized ([DtBoolean](#) onOff)

Set/Get whether the entity is immobilized or not

immobilized () const

setLifeformState ([DtLifeformState](#) s)

Set/Get entity life form state

lifeformState () const

setLightState ([DtLightState](#) s)

Set/Get light state of entity

lightState () const

setDensity ([DtDensityType](#) dens)

Set/Get light state of entity

density () const

setRampDeployed ([DtBoolean](#) onOff)

Set/Get whether entity has ramp deployed or not

rampDeployed () const

setSmokePlumePresent ([DtBoolean](#) onOff)

Set/Get whether a smoke plume is present or not

smokePlumePresent () const

virtual void	setTentDeployed (DtBoolean onOff) <i>Set/Get whether a tent is deployed or not</i>
virtual DtBoolean	tentDeployed () const
virtual void	setTrailState (DtTrailState s) <i>Set/Get trailing effect state</i>
virtual DtTrailState	trailState () const
virtual void	setCamouflageType (DtCamouflageType type) <i>Set/Get camouflage type</i>
virtual DtCamouflageType	camouflageType () const
virtual void	setFirePowerDisabled (DtBoolean onOff) <i>Set/Get whether entity is fire power disabled or not</i>
virtual DtBoolean	firePowerDisabled () const
virtual void	setIsConcealed (DtBoolean onOff) <i>Set/Get whether entity is concealed or not</i>
virtual DtBoolean	isConcealed () const
virtual void	setAfterburnerOn (DtBoolean onOff) <i>Set/Get whether entity has after burner on or not</i>
virtual DtBoolean	afterburnerOn () const
virtual void	setHasAmmunitionSupplyCap (DtBoolean onOff) <i>Set/Get whether entity has ammunition supply capability or not</i>
virtual DtBoolean	hasAmmunitionSupplyCap () const
virtual void	setLauncherRaised (DtBoolean onOff) <i>Set/Get whether entity has launcher raised or not</i>
virtual DtBoolean	launcherRaised () const
virtual void	setLauncherFlashPresent (DtBoolean onOff) <i>Set/Get whether launcher flash is present or not</i>
virtual DtBoolean	launcherFlashPresent () const
virtual void	setPrimaryWeaponState (DtWeaponState s) <i>Set/Get primary weapon deployment state</i>
virtual DtWeaponState	primaryWeaponState () const
virtual void	setSecondaryWeaponState (DtWeaponState s) <i>Set/Get secondary weapon deployment state</i>
virtual DtWeaponState	secondaryWeaponState () const
virtual void	initArtParts (const DtArtPartSpec* specList) <i>initialize articulated parts</i>
virtual const DtArtPartList *	artPartList () const <i>get the articulated parts list</i>
virtual DtArtPartList *	artPartList ()
virtual const DtAttachedPartList *	attPartList () const <i>get the attached parts list</i>
virtual DtAttachedPartList *	attPartList ()
virtual void	setAppearance (DtU32 app) <i>Set/Get the appearance field</i>

virtual DtU32	appearance () const
virtual void	setAppearanceAttribute (const DtAppearAttribute attr, const DtAppearAttrVal val) <i>Set/Get an appearance attribute</i>
virtual DtAppearAttrVal	appearanceAttribute (const DtAppearAttribute attr) const
virtual void	setCapabilities (DtU32 cap) <i>Set/Get the capabilities field</i>
virtual DtU32	capabilities () const
virtual void	printData () const <i>Print the state data</i>
static DtEntityStateRepository *	create ()
inline DtU32	appearance () const
inline const DtArtPartList *	artPartList () const
inline DtArtPartList *	artPartList ()
inline const DtAttachedPartList *	attPartList () const
inline DtAttachedPartList *	attPartList ()
inline DtU32	capabilities () const
inline DtForceType	forceld () const
inline const DtEntityType &	guise () const
inline void	internalSetMarkingText (const char* text)
inline int	markingCharSet () const
inline const char*	markingText () const
inline DtEntityStateRepository *	self () const
inline void	setAppearance (DtU32 app)
inline void	setCapabilities (DtU32 cap)
inline void	setForceld (DtForceType force)
inline void	setGuise (const DtEntityType & type)
inline void	setMarkingCharSet (int charSet)
inline void	setMarkingText (const char* text)

Protected Fields

DtArtPartList	myArtPartList
DtAttachedPartList	myAttPartList
DtForceType	myForceld
DtEntityType	myGuise
DtU32	myAppearance

int	myMarkingCharSet
char*	myMarkingText <i>myMarkingText is an array of theMaxMarkingSize</i>
DtU32	myCapabilities
static const int	theMaxMarkingSize

Protected Methods

DtEntityStateRepository *	self () const <i>Cast off const-ness for trivial internal change</i>
void	internalSetMarkingText (const char* text) <i>Set the marking text</i>

Inherited from [DtBaseEntityStateRepository](#):

Public Methods

- void **setTimeStampType**([DtTimeStampType](#) type)
- [DtTimeStampType](#) **timeStampType**() const
- virtual void **setLocation**(DtConstVector location, DtTime timeNow = DtGetSimTime())
- virtual DtConstVector **location**(DtTime timeNow = DtGetSimTime()) const
- virtual DtConstVector **lastSetLocation**() const
- virtual void **setVelocity**(DtConstVector Velocity, DtTime timeNow = DtGetSimTime())
- virtual DtConstVector **velocity**(DtTime timeNow = DtGetSimTime()) const
- virtual DtConstVector **lastSetVelocity**() const
- virtual void **setAcceleration**(DtConstVector Acceleration, DtTime timeNow = DtGetSimTime())
- virtual DtConstVector **acceleration**(DtTime timeNow = DtGetSimTime()) const
- virtual DtConstVector **lastSetAcceleration**() const
- virtual void **setOrientation**(const [DtTaitBryan](#)& orient, DtTime timeNow = DtGetSimTime())
- virtual const [DtTaitBryan](#)& **orientation**(DtTime timeNow = DtGetSimTime()) const
- virtual const [DtTaitBryan](#)& **lastSetOrientation**() const
- virtual void **setRotationalVelocity**(DtConstVector BodyRates, DtTime timeNow = DtGetSimTime())
- virtual DtConstVector **rotationalVelocity**(DtTime timeNow = DtGetSimTime()) const
- virtual DtConstVector **lastSetRotationalVelocity**() const
- virtual DtConstDcm **bodyToGeoc**(DtTime timeNow = DtGetSimTime()) const

- virtual void **setAlgorithm**([DtDeadReckonTypes](#) alg)
- virtual [DtDeadReckonTypes](#) **algorithm**() const
- virtual void **setEntityId**(const [DtEntityIdentifier](#)& id)
- virtual const [DtEntityIdentifier](#)& **entityId**() const
- virtual void **setEntityType**(const [DtEntityType](#)& type)
- virtual const [DtEntityType](#)& **entityType**() const
- virtual void **setApproximator**([DtApproximator](#)* approx)
- virtual [DtApproximator](#)* **approximator**() const
- virtual void **useDeadReckoner**([DtDeadReckoner](#)* dr)
- virtual void **useDeadReckoner**()
- virtual [DtDeadReckoner](#)* **usingDeadReckoner**()
- virtual void **useSmoother**([DtSmoother](#)* sm)
- virtual void **useSmoother**()
- virtual [DtSmoother](#)* **usingSmoother**()
- virtual [DtThresholder](#)* **thresholder**()
- virtual const [DtThresholder](#)& **thresholder**() const
- virtual void **setTimeForSets**(DtTime time)
- inline const [DtEntityIdentifier](#)& **entityId**() const
- inline const [DtEntityType](#)& **entityType**() const
- inline DtConstVector **lastSetAcceleration**() const
- inline DtConstVector **lastSetLocation**() const
- inline const [DtTaitBryan](#)& **lastSetOrientation**() const
- inline DtConstVector **lastSetRotationalVelocity**() const
- inline DtConstVector **lastSetVelocity**() const
- inline void **setEntityId**(const [DtEntityIdentifier](#)& id)
- inline void **setEntityType**(const [DtEntityType](#)& type)
- inline void **setTimeStampType**([DtTimeStampType](#) type)
- inline [DtThresholder](#)* **thresholder**()
- inline const [DtThresholder](#)& **thresholder**() const
- inline [DtTimeStampType](#) **timeStampType**() const

Protected Fields

- [DtApproximator](#)* myApproximator
- [DtDeadReckoner](#)* myDeadReckoner
- [DtSmoother](#)* mySmoother
- [DtDeadReckoner](#)* myDfltDeadReckoner
- [DtSmoother](#)* myDfltSmoother
- [DtVector](#) myLocation
- [DtVector](#) myVelocity
- [DtVector](#) myAcceleration
- [DtTaitBryan](#) myOrientation
- [DtVector](#) myRotationalVelocity
- [DtDeadReckonTypes](#) myAlgorithm
- [DtDcm](#) myBodyToGeoc
- [DtBoolean](#) myBodyToGeocDirty
- [DtTimeStampType](#) myTimeStampType
- [DtEntityIdentifier](#) myEntityId
- [DtEntityType](#) myEntityType
- [DtBoolean](#) myFrozen
- [DtThresholder](#) myThresholder
- DtTime myTimeForSets

Protected Methods

- virtual DtTime **timeForSets()** const
-

Inherited from [DtStateRepository](#):

Protected



Documentation

class DtEntityStateRepository:

Instances of DtEntityStateRepository are used to maintain entity state. It provides a fat interface which covers various types of entities. As a result, individual fields may not apply depending on the type of entity.

- `DtEntityStateRepository()`
default constructor
- `virtual ~DtEntityStateRepository()`
destructor
- `DtEntityStateRepository(const DtEntityStateRepository& orig)`
copy constructor
- `DtEntityStateRepository& operator=(const DtEntityStateRepository& orig)`
assignment operator
- `virtual DtStateRepository* clone() const`
return an empty object of the same type as this
- `virtual void setForceId(DtForceType force)`
Set/Get the force type
- `virtual DtForceType forceId() const`
- `virtual void setGuise(const DtEntityType& type)`
Set/Get the alternate entity type or guise
- `virtual const DtEntityType& guise() const`
- `virtual void setFrozen(DtBoolean onOff)`
Set/Get whether frozen or not
- `virtual DtBoolean frozen() const`
- `virtual void setMarkingCharSet(int charSet)`
Set/Get the character set used for the marking. Defaults to 1.
- `virtual int markingCharSet() const`
- `virtual void setMarkingText(const char* text)`
Set/Get the marking text
- `virtual const char* markingText() const`
- `virtual void setPowerPlantOn(DtBoolean onOff)`
Set/Get whether entity's power plant is on or not
- `virtual DtBoolean powerPlantOn() const`
- `virtual void setDamageState(DtDamageState s)`
Set/Get the damage state

- virtual [DtDamageState](#) damageState() const
- virtual void setEngineSmokeOn([DtBoolean](#) onOff)
Set/Get whether engine smoke is on or not
- virtual [DtBoolean](#) engineSmokeOn() const
- virtual void setFlamesPresent([DtBoolean](#) onOff)
Set/Get whether flames are present or not
- virtual [DtBoolean](#) flamesPresent() const
- virtual void setHasFuelSupplyCap([DtBoolean](#) onOff)
Set/Get whether entity has fuel supplying capability
- virtual [DtBoolean](#) hasFuelSupplyCap() const
- virtual void setHasRecoveryCap([DtBoolean](#) onOff)
Set/Get whether entity has recovery capability
- virtual [DtBoolean](#) hasRecoveryCap() const
- virtual void setHasRepairCap([DtBoolean](#) onOff)
Set/Get whether entity has repair capability
- virtual [DtBoolean](#) hasRepairCap() const
- virtual void setHatchState([DtHatchState](#) s)
Set/Get the entity's hatch state
- virtual [DtHatchState](#) hatchState() const
- virtual void setImmobilized([DtBoolean](#) onOff)
Set/Get whether the entity is immobilized or not
- virtual [DtBoolean](#) immobilized() const
- virtual void setLifeformState([DtLifeformState](#) s)
Set/Get entity life form state
- virtual [DtLifeformState](#) lifeformState() const
- virtual void setLightState([DtLightState](#) s)
Set/Get light state of entity
- virtual [DtLightState](#) lightState() const
- virtual void setDensity([DtDensityType](#) dens)
Set/Get light state of entity. (Environment entities).
- virtual [DtDensityType](#) density() const
- virtual void setRampDeployed([DtBoolean](#) onOff)
Set/Get whether entity has ramp deployed or not

- virtual [DtBoolean](#) rampDeployed() const
- virtual void setSmokePlumePresent([DtBoolean](#) onOff)
Set/Get whether a smoke plume is present or not
- virtual [DtBoolean](#) smokePlumePresent() const
- virtual void setTentDeployed([DtBoolean](#) onOff)
Set/Get whether a tent is deployed or not
- virtual [DtBoolean](#) tentDeployed() const
- virtual void setTrailState([DtTrailState](#) s)
Set/Get trailing effect state
- virtual [DtTrailState](#) trailState() const
- virtual void setCamouflageType([DtCamouflageType](#) type)
Set/Get camouflage type
- virtual [DtCamouflageType](#) camouflageType() const
- virtual void setFirePowerDisabled([DtBoolean](#) onOff)
Set/Get whether entity is fire power disabled or not
- virtual [DtBoolean](#) firePowerDisabled() const
- virtual void setIsConcealed([DtBoolean](#) onOff)
Set/Get whether entity is concealed or not
- virtual [DtBoolean](#) isConcealed() const
- virtual void setAfterburnerOn([DtBoolean](#) onOff)
Set/Get whether entity has after burner on or not
- virtual [DtBoolean](#) afterburnerOn() const
- virtual void setHasAmmunitionSupplyCap([DtBoolean](#) onOff)
Set/Get whether entity has ammunition supply capability or not
- virtual [DtBoolean](#) hasAmmunitionSupplyCap() const
- virtual void setLauncherRaised([DtBoolean](#) onOff)
Set/Get whether entity has launcher raised or not
- virtual [DtBoolean](#) launcherRaised() const
- virtual void setLauncherFlashPresent([DtBoolean](#) onOff)
Set/Get whether launcher flash is present or not
- virtual [DtBoolean](#) launcherFlashPresent() const
- virtual void setPrimaryWeaponState([DtWeaponState](#) s)
Set/Get primary weapon deployment state

- virtual [DtWeaponState](#) primaryWeaponState() const
- virtual void setSecondaryWeaponState([DtWeaponState](#) s)
Set/Get secondary weapon deployment state
- virtual [DtWeaponState](#) secondaryWeaponState() const
- virtual void initArtParts(const DtArtPartSpec* specList)
initialize articulated parts
- virtual const [DtArtPartList](#)* artPartList() const
get the articulated parts list
- virtual [DtArtPartList](#)* artPartList()
- virtual const [DtAttachedPartList](#)* attPartList() const
get the attached parts list
- virtual [DtAttachedPartList](#)* attPartList()
- virtual void setAppearance(DtU32 app)
Set/Get the appearance field
- virtual DtU32 appearance() const
- virtual void setAppearanceAttribute(const [DtAppearAttribute](#) attr, const [DtAppearAttrVal](#) val)
Set/Get an appearance attribute. attr is the appearance bitfield mask val is the hex number to be or'ed in
- virtual [DtAppearAttrVal](#) appearanceAttribute(const [DtAppearAttribute](#) attr) const
- virtual void setCapabilities(DtU32 cap)
Set/Get the capabilities field
- virtual DtU32 capabilities() const
- virtual void printData() const
Print the state data
- static [DtEntityStateRepository](#)* create()
- [DtEntityStateRepository](#)* self() const
Cast off const-ness for trivial internal change. Non-virtual since can't override on return type.
- void internalSetMarkingText(const char* text)
Set the marking text. Non-virtual since meant to be used by constructor.
- [DtArtPartList](#) myArtPartList
- [DtAttachedPartList](#) myAttPartList
- [DtForceType](#) myForceId
- [DtEntityType](#) myGuisse
- DtU32 myAppearance

```

int myMarkingCharSet
char* myMarkingText
    myMarkingText is an array of theMaxMarkingSize. String length is theMaxMarkingSize - 1 (doesn't include NULL).
DtU32 myCapabilities
static const int theMaxMarkingSize
inline DtU32 appearance() const
inline const DtArtPartList* artPartList() const
inline DtArtPartList* artPartList()
inline const DtAttachedPartList* attPartList() const
inline DtAttachedPartList* attPartList()
inline DtU32 capabilities() const
inline DtForceType forceId() const
inline const DtEntityType& guise() const
inline void internalSetMarkingText(const char* text)
inline int markingCharSet() const
inline const char* markingText() const
inline DtEntityStateRepository* self() const
inline void setAppearance(DtU32 app)
inline void setCapabilities(DtU32 cap)
inline void setForceId(DtForceType force)
inline void setGuise(const DtEntityType& type)
inline void setMarkingCharSet(int charSet)
inline void setMarkingText(const char* text)

```

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtDamageState

This is for the tank ditch and concertina wire breach appearance Areal object specific appearance

-  DtDamageNone
 -  DtDamageSlight
 -  DtDamageModerate
 -  DtDamageDestroyed
-

Documentation

This is for the tank ditch and concertina wire breach appearance Areal object specific appearance

-  DtDamageNone
-  DtDamageSlight
-  DtDamageModerate
-  DtDamageDestroyed

In file disenums.h:

enum DtHatchState

-
-  DtHatchNA
 -  DtHatchClosed
 -  DtHatchPopped
 -  DtHatchPoppedAndPerson
 -  DtHatchOpen
 -  DtHatchOpenAndPerson
-

Documentation

-  DtHatchNA
-  DtHatchClosed
-  DtHatchPopped
-  DtHatchPoppedAndPerson
-  DtHatchOpen
-  DtHatchOpenAndPerson

In file disenums.h:

enum DtLifeformState

-  DtLifeformNA
 -  DtLifeformStanding
 -  DtLifeformWalking
 -  DtLifeformRunning
 -  DtLifeformKneeling
 -  DtLifeformProne
 -  DtLifeformCrawling
 -  DtLifeformSwimming
 -  DtLifeformParachuting
 -  DtLifeformJumping
 -  DtLifeformSitting
 -  DtLifeformSquatting
 -  DtLifeformCrouching
 -  DtLifeformWading
-

Documentation

-  DtLifeformNA
-  DtLifeformStanding
-  DtLifeformWalking
-  DtLifeformRunning
-  DtLifeformKneeling
-  DtLifeformProne
-  DtLifeformCrawling
-  DtLifeformSwimming
-  DtLifeformParachuting

- DtLifeformJumping
- DtLifeformSitting
- DtLifeformSquatting
- DtLifeformCrouching
- DtLifeformWading

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtLightState

-
-  DtLightsNone
 -  DtLightsRunning
 -  DtLightsNavigation
 -  DtLightsFormation
-

Documentation

-  DtLightsNone
-  DtLightsRunning
-  DtLightsNavigation
-  DtLightsFormation

In file disenums.h:

enum DtDensityType

-  DtDensityClear
 -  DtDensityHazy
 -  DtDensityDense
 -  DtDensityVeryDense
 -  DtDensityOpaque
-

Documentation

-  DtDensityClear
-  DtDensityHazy
-  DtDensityDense
-  DtDensityVeryDense
-  DtDensityOpaque

In file disenums.h:

enum DtTrailState

-  DtTrailingEffectsNone
 -  DtTrailingEffectsSmall
 -  DtTrailingEffectsMedium
 -  DtTrailingEffectsLarge
-

Documentation

-  DtTrailingEffectsNone
-  DtTrailingEffectsSmall
-  DtTrailingEffectsMedium
-  DtTrailingEffectsLarge

In file disenums.h:

enum DtCamouflageType

-
-  DtDesertCamouflage
 -  DtWinterCamouflage
 -  DtForestCamouflage
-

Documentation

-  DtDesertCamouflage
-  DtWinterCamouflage
-  DtForestCamouflage

In file disenums.h:

enum DtWeaponState

-
-  DtNoWeapon
 -  DtWeaponStowed
 -  DtWeaponDeployed
 -  DtWeaponInFirePosition
-

Documentation

-  DtNoWeapon
-  DtWeaponStowed
-  DtWeaponDeployed
-  DtWeaponInFirePosition

In file objdeslist.h:

class DtGlobalObjectDesignatorList

Public Methods

	DtGlobalObjectDesignatorList () <i>default constructor</i>
virtual	~DtGlobalObjectDesignatorList () <i>destructor</i>
	DtGlobalObjectDesignatorList (const DtGlobalObjectDesignatorList & orig) <i>copy constructor</i>
DtGlobalObjectDesignatorList &	operator= (const DtGlobalObjectDesignatorList & orig) <i>assignment operator</i>
int	operator== (const DtGlobalObjectDesignatorList & other) const <i>equivlance operators</i>
int	operator!= (const DtGlobalObjectDesignatorList & other) const
virtual void	setNumObjects (int num) <i>Set/Get number of objects</i>
virtual int	numObjects () const
virtual DtBoolean	setObject (int index, const DtGlobalObjectDesignator& des) <i>Set index'th object designator</i>
virtual const DtGlobalObjectDesignator&	object (int index, DtBoolean * valid = NULL) const <i>Get index'th object designator</i>
virtual void	print () const <i>Print data</i>
virtual void	setFromNet (const void* netRep, int length) <i>Set from network representation</i>
virtual const void*	netRep (int* size = NULL) const <i>Returns network representation, fills size with size of netRep</i>
virtual int	netSize () const <i>Returns size of network representation</i>

Protected Fields

int	myNum
DtGlobalObjectDesignator*	myObjs

Documentation

- `DtGlobalObjectDesignatorList()`
default constructor
- `virtual ~DtGlobalObjectDesignatorList()`
destructor
- `DtGlobalObjectDesignatorList(const DtGlobalObjectDesignatorList& orig)`
copy constructor
- `DtGlobalObjectDesignatorList& operator=( const DtGlobalObjectDesignatorList& orig)`
assignment operator
- `int operator==(const DtGlobalObjectDesignatorList& other) const`
equivlance operators
- `int operator!=(const DtGlobalObjectDesignatorList& other) const`
- `virtual void setNumObjects(int num)`
Set/Get number of objects. Setting to a number lower than the current number will remove trailing objects from the list.
- `virtual int numObjects() const`
- `virtual DtBoolean setObject(int index, const DtGlobalObjectDesignator& des)`
Set index'th object designator
- `virtual const DtGlobalObjectDesignator& object( int index, DtBoolean* valid = NULL) const`
Get index'th object designator. Set valid to DtFalse if bad index, DtTrue otherwise.
- `virtual void print() const`
Print data
- `virtual void setFromNet(const void* netRep, int length)`
Set from network representation
- `virtual const void* netRep(int* size = NULL) const`
Returns network representation, fills size with size of netRep. Return value is pointer to internal object with same lifetime as this. Do not delete.
- `virtual int netSize() const`
Returns size of network representation
- `int myNum`
- `DtGlobalObjectDesignator* myObjs`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file daggdec.h:

class DtAggregateStateDecoder

class DtAggregateStateDecoder: Instances of DtAggregateStateDecoder are used to take state information from an aggregate state PDU and place it into an aggregate state repository

Public Methods

	DtAggregateStateDecoder () <i>default constructor</i>
virtual	~DtAggregateStateDecoder () <i>destructor</i>
	DtAggregateStateDecoder (const DtAggregateStateDecoder & orig) <i>copy constructor</i>
DtAggregateStateDecoder &	operator= (const DtAggregateStateDecoder & orig) <i>assignment operator</i>
virtual void	decode (const DtAggregateStatePdu & pdu, DtAggregateStateRepository * stateRep) <i>Decode an aggregate state PDU and update a state repository</i>

Documentation

class DtAggregateStateDecoder: Instances of DtAggregateStateDecoder are used to take state information from an aggregate state PDU and place it into an aggregate state repository

	DtAggregateStateDecoder () default constructor
virtual	~DtAggregateStateDecoder () destructor
	DtAggregateStateDecoder (const DtAggregateStateDecoder & orig) copy constructor
DtAggregateStateDecoder &	operator= (const DtAggregateStateDecoder & orig) assignment operator
virtual void	decode (const DtAggregateStatePdu & pdu, DtAggregateStateRepository * stateRep) Decode an aggregate state PDU and update a state repository

This class has no child classes.

In file dagenc.h:

class DtAggregateStateEncoder

class DtAggregateStateEncoder: Instances of DtAggregateStateEncoder are used to take state information from an aggregate state repository and place it into an aggregate state PDU

Public Methods

- **DtAggregateStateEncoder ()**
default constructor
- virtual **~DtAggregateStateEncoder ()**
destructor
- **DtAggregateStateEncoder (const [DtAggregateStateEncoder](#)& orig)**
copy constructor
- [DtAggregateStateEncoder](#)& **operator= (const [DtAggregateStateEncoder](#)& orig)**
assignment operator
- virtual [DtAggregateStatePdu](#)* **stateMsg (const [DtAggregateStateRepository](#)& stateRep, const [DtAggregateStateRepository](#)& asSeenByRemote)**
create a state message from a local and remote state repository
- virtual void **setSendNeeded ()**
force an state message to be generated

Protected Fields

- [DtAggregateStatePdu](#) **myAsPdu**
- int **forceSendNeeded**

Protected Methods

- virtual [DtBoolean](#) **sendNeeded (const [DtAggregateStateRepository](#)& stateRep, const [DtAggregateStateRepository](#)& asSeenByRemote)**
determine if a state message needs to be sent
 - virtual void **fillOut (const [DtAggregateStateRepository](#)& stateRep, const [DtAggregateStateRepository](#)& asSeenByRemote, [DtAggregateStatePdu](#)* pdu)**
fill out a state message which reflects a state repository
-

Documentation

class DtAggregateStateEncoder: Instances of DtAggregateStateEncoder are used to take state information from an aggregate state repository and place it into an aggregate state PDU

- **DtAggregateStateEncoder ()**

default constructor

• `virtual ~DtAggregateStateEncoder()`

destructor

• `DtAggregateStateEncoder(const DtAggregateStateEncoder& orig)`

copy constructor

• `DtAggregateStateEncoder& operator=(const DtAggregateStateEncoder& orig)`

assignment operator

• `virtual DtAggregateStatePdu* stateMsg( const DtAggregateStateRepository& stateRep, const DtAggregateStateRepository& asSeenByRemote)`

create a state message from a local and remote state repository

• `virtual void setSendNeeded()`

force an state message to be generated

• `virtual DtBoolean sendNeeded( const DtAggregateStateRepository& stateRep, const DtAggregateStateRepository& asSeenByRemote)`

determine if a state message needs to be sent

• `virtual void fillOut( const DtAggregateStateRepository& stateRep, const DtAggregateStateRepository& asSeenByRemote, DtAggregateStatePdu* pdu)`

fill out a state message which reflects a state repository

• `DtAggregateStatePdu myAsPdu`

• `int forceSendNeeded`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file cartview.h:

class DtCartesianView

class DtCartesianView: Instances of DtCartesianView are used to create a Cartesian coordinate view of the geocentric coordinates in an Entity State Repository

Inheritance:

DtCartesianView

Public Methods

•	DtCartesianView (DtBaseEntityStateRepository * ksr) <i>constructor</i>
•	DtCartesianView (DtBaseEntityStateRepository * ksr, DtConstVector originInGeoc, DtConstDcm geocToLocalRot)
• virtual	~DtCartesianView () <i>destructor</i>
•	DtCartesianView (const DtCartesianView & orig) <i>copy constructor</i>
• DtCartesianView &	operator= (const DtCartesianView & orig) <i>assignment operator</i>
• virtual void	setOffset (DtConstVector originInGeoc) <i>set the origin of the local system in geocentric coords</i>
• virtual void	setRotation (DtConstDcm geocToLocalRot) <i>set the rotation matrix that converts from geocentric to local coords</i>
• virtual void	setLocation (DtConstVector Location) <i>Set/Get the location in local coordinates</i>
• virtual DtConstVector	location () const
• virtual void	setVelocity (DtConstVector Velocity) <i>Set/Get the velocity</i>
• virtual DtConstVector	velocity () const
• virtual void	setOrientation (const DtTaitBryan & Orient) <i>Set/Get the orientation</i>
• virtual const DtTaitBryan &	orientation () const
• virtual DtConstDcm	bodyToLocal () const <i>Get the body to local coordinate matrix</i>
• virtual void	setAcceleration (DtConstVector Acceleration) <i>Set/Get the acceleration</i>
• virtual DtConstVector	acceleration () const

- virtual void **setRotationalVelocity** (DtConstVector BodyRates)
Set/Get the rotational velocity
- virtual DtConstVector **rotationalVelocity** () const
- inline [DtCartesianView](#)* **self** () const

Protected Fields

- [DtBaseEntityStateRepository](#)* **myESR**
- [DtCoordTransform](#) **myGeocToLocal**
- [DtCoordTransform](#) **myLocalToGeoc**
- [DtVector](#) **myLocalLoc**
used as return values by inspectors
- [DtVector](#) **myLocalVel**
- [DtVector](#) **myLocalAccel**
- [DtTaitBryan](#) **myLocalOrient**
- [DtDcm](#) **myBodyToLocal**

Protected Methods

- [DtCartesianView](#)* **self** () const
Cast off const-ness for trivial internal change
-

Documentation

class DtCartesianView: Instances of DtCartesianView are used to create a Cartesian coordinate view of the geocentric coordinates in an Entity State Repository

- DtCartesianView**([DtBaseEntityStateRepository](#)* ksr)
constructor
- DtCartesianView**([DtBaseEntityStateRepository](#)* ksr, DtConstVector originInGeoc, DtConstDcm geocToLocalRot)
- virtual ~DtCartesianView**()
destructor
- DtCartesianView**(const [DtCartesianView](#)& orig)
copy constructor
- [DtCartesianView](#)& operator=(const [DtCartesianView](#)& orig)
assignment operator
- virtual void setOffset**(DtConstVector originInGeoc)
set the origin of the local system in geocentric coords

- `virtual void setRotation(DtConstDcm geocToLocalRot)`
set the rotation matrix that converts from geocentric to local coords
- `virtual void setLocation(DtConstVector Location)`
Set/Get the location in local coordinates
- `virtual DtConstVector location() const`
- `virtual void setVelocity(DtConstVector Velocity)`
Set/Get the velocity
- `virtual DtConstVector velocity() const`
- `virtual void setOrientation(const DtTaitBryan& Orient)`
Set/Get the orientation
- `virtual const DtTaitBryan& orientation() const`
- `virtual DtConstDcm bodyToLocal() const`
Get the body to local coordinate matrix. This is just a matrix representation of the euler angles returned by `orientation()`.
- `virtual void setAcceleration(DtConstVector Acceleration)`
Set/Get the acceleration
- `virtual DtConstVector acceleration() const`
- `virtual void setRotationalVelocity(DtConstVector BodyRates)`
Set/Get the rotational velocity
- `virtual DtConstVector rotationalVelocity() const`
- `DtCartesianView* self() const`
Cast off const-ness for trivial internal change. Non-virtual since can't override on return type.
- `DtBaseEntityStateRepository* myESR`
- `DtCoordTransform myGeocToLocal`
- `DtCoordTransform myLocalToGeoc`
- `DtVector myLocalLoc`
used as return values by inspectors. Do not necessarily hold current state.
- `DtVector myLocalVel`
- `DtVector myLocalAccel`
- `DtTaitBryan myLocalOrient`
- `DtDcm myBodyToLocal`
- `inline DtCartesianView* self() const`

Direct child classes:

[DtTopoView](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file libmatrix.h:

class DtCoordTransform

Public Fields

-  [DtVector](#) **offset**
these were protected, but it is often more efficient to set these by passing them to another function
-  [DtDcm](#) **rotation**
"to" origin in base coords

Public Methods

-  **DtCoordTransform** ()
 -  **DtCoordTransform** (DtConstVector toOriginInFromCoord, DtConstDcm from2to)
 -  void **setOffset** (DtConstVector toOriginInFromCoord)
 -  void **setRotation** (DtConstDcm from2to)
 -  void **setByInverse** (const [DtCoordTransform](#) &to2from)
 -  void **getOffset** (DtVectorRef toOriginInFromCoord) const
 -  void **getOrientation** (DtDcmRef from_to) const
 -  void **coordTrans** (DtConstVector from, DtVectorRef to) const
 -  void **vecTrans** (DtConstVector from, DtVectorRef to) const
 -  void **eulerTrans** (const [DtTaitBryan](#) &from, [DtTaitBryan](#) *to) const
-

Documentation

-  **DtCoordTransform** ()
-  **DtCoordTransform** (DtConstVector toOriginInFromCoord, DtConstDcm from2to)
-  void **setOffset** (DtConstVector toOriginInFromCoord)
-  void **setRotation** (DtConstDcm from2to)
-  void **setByInverse** (const [DtCoordTransform](#) &to2from)
-  void **getOffset** (DtVectorRef toOriginInFromCoord) const
-  void **getOrientation** (DtDcmRef from_to) const
-  void **coordTrans** (DtConstVector from, DtVectorRef to) const

- `void vecTrans(DtConstVector from, DtVectorRef to) const`
- `void eulerTrans(const DtTaitBryan &from, DtTaitBryan *to) const`
- [DtVector](#) offset

these were protected, but it is often more efficient to set these by passing them to another function

- [DtDcm](#) rotation

"to" origin in base coords

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file topview.h:

class DtTopoView: public DtCartesianView

class DtTopoView: Instances of DtTopoView are used to create a Topographic coordinate view of the geocentric coordinates in an Entity State Repository

Inheritance:

DtTopoView - DtCartesianView

Public Methods

- **DtTopoView** ([DtBaseEntityStateRepository](#)* esr, double latitude, double longitude)
constructor
 - virtual **~DtTopoView** ()
destructor
 - **DtTopoView** (const [DtTopoView](#)& orig)
copy constructor
 - [DtTopoView](#)& **operator=** (const [DtTopoView](#)& orig)
assignment operator
-

Inherited from DtCartesianView:

Public Methods

- virtual void **setOffset**(DtConstVector originInGeoc)
- virtual void **setRotation**(DtConstDcm geocToLocalRot)
- virtual void **setLocation**(DtConstVector Location)
- virtual DtConstVector **location**() const
- virtual void **setVelocity**(DtConstVector Velocity)
- virtual DtConstVector **velocity**() const
- virtual void **setOrientation**(const [DtTaitBryan](#)& Orient)
- virtual const [DtTaitBryan](#)& **orientation**() const
- virtual DtConstDcm **bodyToLocal**() const

- virtual void **setAcceleration**(DtConstVector Acceleration)
- virtual DtConstVector **acceleration**() const
- virtual void **setRotationalVelocity**(DtConstVector BodyRates)
- virtual DtConstVector **rotationalVelocity**() const
- inline [DtCartesianView](#)* **self**() const

Protected Fields

- [DtBaseEntityStateRepository](#)* myESR
- [DtCoordTransform](#) myGeocToLocal
- [DtCoordTransform](#) myLocalToGeoc
- [DtVector](#) myLocalLoc
- [DtVector](#) myLocalVel
- [DtVector](#) myLocalAccel
- [DtTaitBryan](#) myLocalOrient
- [DtDcm](#) myBodyToLocal

Protected Methods

- [DtCartesianView](#)* **self**() const
-

Documentation

class DtTopoView: Instances of DtTopoView are used to create a Topographic coordinate view of the geocentric coordinates in an Entity State Repository

- DtTopoView**([DtBaseEntityStateRepository](#)* esr, double latitude, double longitude)
constructor
 - virtual ~DtTopoView**()
destructor
 - DtTopoView**(const [DtTopoView](#)& orig)
copy constructor
 - [DtTopoView](#)& operator=(const [DtTopoView](#)& orig)**
assignment operator
-

This class has no child classes.

In file libmatrix.h:

class DtCoordTranslation

Public Methods

- **DtCoordTranslation** ()
- **DtCoordTranslation** (DtConstVector toOriginInFromCoord)
- void **setOffset** (DtConstVector toOriginInFromCoord)
- void **setByInverse** ([DtCoordTranslation](#) &to2from)
- void **getOffset** (DtVectorRef toOriginInFromCoord)
- void **coordTrans** (DtConstVector from, DtVectorRef to)

Private Fields

- [DtVector](#) **offset**
 - *"to" origin in base coords*
-

Documentation

- `DtCoordTranslation()`
 - `DtCoordTranslation(DtConstVector toOriginInFromCoord)`
 - `void setOffset(DtConstVector toOriginInFromCoord)`
 - `void setByInverse(DtCoordTranslation &to2from)`
 - `void getOffset(DtVectorRef toOriginInFromCoord)`
 - `void coordTrans(DtConstVector from, DtVectorRef to)`
 - [DtVector](#) **offset**
 - *"to" origin in base coords*
-

This class has no child classes.

In file datummap.h:

class DtDatumMap

Public Methods

-  **DtDatumMap** (void)
-  void **setData** (const [DtSetDataPdu](#) &data, void *usr)
-  void **fillDataPdu** ([DtDataPdu](#) *dataPdu, void *usr)
-  void **initDataPdu** (const [DtSetDataPdu](#) &setDataPdu, [DtDataPdu](#) *dataPdu)
-  void **initDataPdu** (const [DtDataQueryPdu](#) &queryPdu, [DtDataPdu](#) *dataPdu)
-  void **makeResponseDataPdu** (const [DtSetDataPdu](#) &setDataPdu, [DtDataPdu](#) *dataPdu, void *usr)
-  void **makeResponseDataPdu** (const [DtDataQueryPdu](#) &queryPdu, [DtDataPdu](#) *dataPdu, void *usr)
-  [DtDatumMapEntry](#)* **lookup** ([DtDatumParam](#) id, int verbose=1)
-  void **install** ([DtDatumMapEntry](#) *entry, [DtDatumParam](#) id)
-  void **remove** ([DtDatumParam](#) id, int verbose=1)

Protected Fields

-  [DtHashlist](#) list
-

Documentation

-  **DtDatumMap** (void)
-  void **setData** (const [DtSetDataPdu](#) &data, void *usr)
-  void **fillDataPdu** ([DtDataPdu](#) *dataPdu, void *usr)
-  void **initDataPdu** (const [DtSetDataPdu](#) &setDataPdu, [DtDataPdu](#) *dataPdu)
-  void **initDataPdu** (const [DtDataQueryPdu](#) &queryPdu, [DtDataPdu](#) *dataPdu)
-  void **makeResponseDataPdu** (const [DtSetDataPdu](#) &setDataPdu, [DtDataPdu](#) *dataPdu, void *usr)
-  void **makeResponseDataPdu** (const [DtDataQueryPdu](#) &queryPdu, [DtDataPdu](#) *dataPdu, void *usr)
-  [DtDatumMapEntry](#)* **lookup** ([DtDatumParam](#) id, int verbose=1)

- `void install(DtDatumMapEntry *entry, DtDatumParam id)`
- `void remove(DtDatumParam id, int verbose=1)`
- `DtHashlist list`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file datummap.h:

class DtDatumMapEntry

Inheritance:

DtDatumMapEntry

Public Methods

- **DtDatumMapEntry** ()
 - virtual void **setdata** (const [DtSetDataPdu](#) &setDataPdu, int fixedOrVar, int index, void *usr)
 - virtual void **getdata** ([DtDataPdu](#) *dataPdu, int fixedOrVar, int index, void *usr)
-

Documentation

- **DtDatumMapEntry** ()
 - virtual void **setdata** (const [DtSetDataPdu](#) &setDataPdu, int fixedOrVar, int index, void *usr)
 - virtual void **getdata** ([DtDataPdu](#) *dataPdu, int fixedOrVar, int index, void *usr)
-

Direct child classes:

[DtDmeGlobalFcns](#)

In file datummap.h:

class DtDmeGlobalFcns: public DtDatumMapEntry

Inheritance:

DtDmeGlobalFcns - DtDatumMapEntry

Public Methods

- **DtDmeGlobalFcns** (DtGlobalSetFcn setF, DtGlobalGetFcn getF)
- void **setdata** (const [DtSetDataPdu](#) &setDataPdu, int fixedOrVar, int index, void *usr)
- void **getdata** ([DtDataPdu](#) *dataPdu, int fixedOrVar, int index, void *usr)

Protected Fields

- DtGlobalGetFcn **getFcn**
 - DtGlobalSetFcn **setFcn**
-

Inherited from DtDatumMapEntry:

Documentation

- DtDmeGlobalFcns (DtGlobalSetFcn setF, DtGlobalGetFcn getF)
 - void setdata (const [DtSetDataPdu](#) &setDataPdu, int fixedOrVar, int index, void *usr)
 - void getdata ([DtDataPdu](#) *dataPdu, int fixedOrVar, int index, void *usr)
 - DtGlobalGetFcn getFcn
 - DtGlobalSetFcn setFcn
-

This class has no child classes.

In file dentitypub.h:

class DtEntityPublisher

Public Methods

	DtEntityPublisher (DtExerciseConn * conn, const DtObjectId& id = gFindObjectId()) <i>Constructors</i>
	DtEntityPublisher (const DtEntityType & type, DtExerciseConn * conn, const DtObjectId& id = gFindObjectId())
	DtEntityPublisher (const DtEntityType & type, DtExerciseConn * conn, DtDeadReckonTypes algorithm, DtForceType forceId, const DtEntityType & guise = gGuiseSameAsType(), const DtObjectId& id = gFindObjectId())
	~DtEntityPublisher () <i>Destructors</i>
	tick () <i>This function should be called once per frame for each entity</i>
	entityStateRep () <i>Returns a pointer to the EntityStateRepository - through which you can set or inspect current state information</i>
	esr () <i>Synonym for entityStateRep()</i>
	decoder () <i>Returns a pointer to the Decoder being used to decode PDUs that we send back into the asSeenByRemote ESR</i>
	encoder () <i>Returns a pointer to the Encoder being used to encode current state into an entity state PDU</i>
	id () const <i>Get the entity's identifier</i>
	localId () const <i>Get the entity's localId - same as id for DIS</i>
	globalId () const <i>Get the global object designator - same as id for DIS</i>
	setTimeThreshold (DtTime time) <i>Set/Get time threshold</i>
	timeThreshold () const
	sendFinalPduOnDestruction (DtBoolean onOff) <i>Set flag indicating whether to send a final PDU with the "FinalPdu" appearance bit set, from within the DtEntityPublisher destructor</i>
	forceUpdate () <i>Force a state update to be sent during the next call to tick, even if nothing would normally cause a send</i>
	exerciseConn () <i>Get our exercise connection</i>

- static void
- static DtTime
- static void
- static double
- static void
- static double
- static const DtObjectId&
- static const [DtEntityType](#)&
- inline [DtEntityStateDecoder](#)*
- inline double
- inline DtTime
- inline double
- inline [DtEntityStateEncoder](#)*
- else inline [DtEntityStateRepository](#)*
- inline [DtEntityStateRepository](#)*
- inline [DtExerciseConn](#)*
- return
- inline const DtObjectId&
- inline const DtObjectId&
- inline void
- inline void
- inline void
- inline void
- inline void
- inline DtTime

setDefaultTimeThreshold (DtTime time)
Set/Get default time threshold

dfltTimeThreshold ()

setDefaultRotationThreshold (DtReal r)
These four functions are obsolete, but remain for backwards compatibility

dfltRotationThreshold ()

setDefaultTranslationThreshold (DtReal t)

dfltTranslationThreshold ()

findOwnId ()
Returns a special DtObjectId that is a signal to the DtEntityPublisher constructor that it should obtain its own ObjectId

guiseSameAsType ()
Returns a special DtEntityType that indicates to the DtEntityPublisher constructor that the entity type should also be used as the guise

decoder ()

dfltRotationThreshold ()

dfltTimeThreshold ()

dfltTranslationThreshold ()

encoder ()

entityStateRep ()

esr ()

exerciseConn ()

findOwnId ()

id () const

localId () const

sendFinalPduOnDestruction ([DtBoolean](#) onOff)

setDefaultRotationThreshold (DtReal r)

setDefaultTimeThreshold (DtTime time)

setDefaultTranslationThreshold (DtReal t)

setTimeThreshold (DtTime time)

timeThreshold () const

Protected Fields

- DtObjectId myId
- [DtExerciseConn](#)* myExConn
- [DtEntityStateRepository](#)* myEntityStateRep
- [DtEntityStateRepository](#)* asSeenByRemote

DtEntityStateDecoder *	myEntityDecoder
DtEntityStateEncoder *	myEntityEncoder
DtTime	mySimTimeLastUpdate
DtTime	myRealTimeLastUpdate
DtBoolean	mySendFinalInDtor
DtTime	myTimeThreshold
static DtTime	theDfltTimeThreshold

Protected Methods

- virtual void **send** ([DtEntityStatePdu](#)* pdu) const
Called by tick() to send a PDU
 - void **sendFinalPdu** ()
not virtual since called from destructor
 - void **init** ([DtExerciseConn](#)* exer, const [DtEntityIdentifier](#)& vehId)
not virtual since called from constructor
-

Documentation

- [DtEntityPublisher](#)([DtExerciseConn](#)* conn, const DtObjectId& id = gFindOwnId())
Constructors

- [DtEntityPublisher](#)(const [DtEntityType](#)& type, [DtExerciseConn](#)* conn, const DtObjectId& id = gFindOwnId())

- [DtEntityPublisher](#)(const [DtEntityType](#)& type, [DtExerciseConn](#)* conn, [DtDeadReckonTypes](#) algorithm, [DtForceType](#) forceId, const [DtEntityType](#)& guise = gGuiseSameAsType(), const DtObjectId& id = gFindOwnId())

- virtual ~[DtEntityPublisher](#)()
Destructors

- virtual void tick()
This function should be called once per frame for each entity. It determines if any data needs to be sent to the other exercise participants, and sends that data.

- virtual [DtEntityStateRepository](#)* entityStateRep()
Returns a pointer to the EntityStateRepository - through which you can set or inspect current state information

- virtual [DtEntityStateRepository](#)* esr()
Synonym for entityStateRep()

- virtual [DtEntityStateDecoder](#)* decoder()
Returns a pointer to the Decoder being used to decode PDUs that we send back into the asSeenByRemote ESR.

- virtual [DtEntityStateEncoder](#)* encoder()

Returns a pointer to the Encoder being used to encode current state into an entity state PDU

• `virtual const DtObjectId& id() const`

Get the entity's identifier

• `virtual const DtObjectId& localId() const`

Get the entity's localId - same as id for DIS

• `virtual const DtGlobalObjectDesignator& globalId() const`

Get the global object designator - same as id for DIS

• `virtual void setTimeThreshold(DtTime time)`

Set/Get time threshold

• `virtual DtTime timeThreshold() const`

• `virtual void sendFinalPduOnDestruction(DtBoolean onOff)`

Set flag indicating whether to send a final PDU with the "FinalPdu" appearance bit set, from within the DtEntityPublisher destructor

• `virtual void forceUpdate()`

Force a state update to be sent during the next call to tick, even if nothing would normally cause a send

• `virtual DtExerciseConn* exerciseConn()`

Get our exercise connection

• `static void setDfltTimeThreshold(DtTime time)`

Set/Get default time threshold

• `static DtTime dfltTimeThreshold()`

• `static void setDfltRotationThreshold(DtReal r)`

These four functions are obsolete, but remain for backwards compatibility. Please use the DtThresholder static member functions instead.

• `static double dfltRotationThreshold()`

• `static void setDfltTranslationThreshold(DtReal t)`

• `static double dfltTranslationThreshold()`

• `static const DtObjectId& findOwnId()`

Returns a special DtObjectId that is a signal to the DtEntityPublisher constructor that it should obtain its own ObjectId

• `static const DtEntityType& guiseSameAsType()`

Returns a special DtEntityType that indicates to the DtEntityPublisher constructor that the entity type should also be used as the guise

• `virtual void send(DtEntityStatePdu* pdu) const`

Called by tick() to send a PDU. By overriding this function, you can tweak the pdu just before it is sent.

• `void sendFinalPdu()`

not virtual since called from destructor

• `void init(DtExerciseConn* exer, const DtEntityIdentifier& vehId)`

not virtual since called from constructor

```
DtObjectId myId
DtExerciseConn* myExConn
DtEntityStateRepository* myEntityStateRep
DtEntityStateRepository* asSeenByRemote
DtEntityStateDecoder* myEntityDecoder
DtEntityStateEncoder* myEntityEncoder
DtTime mySimTimeLastUpdate
DtTime myRealTimeLastUpdate
DtBoolean mySendFinalInDtor
DtTime myTimeThreshold
static DtTime theDfltTimeThreshold
inline DtEntityStateDecoder* decoder()
inline double dfltRotationThreshold()
inline DtTime dfltTimeThreshold()
inline double dfltTranslationThreshold()
inline DtEntityStateEncoder* encoder()
else inline DtEntityStateRepository* entityStateRep()
inline DtEntityStateRepository* esr()
inline DtExerciseConn* exerciseConn()
return findOwnId()
inline const DtObjectId& id() const
inline const DtObjectId& localId() const
inline void sendFinalPduOnDestruction(DtBoolean onOff)
inline void setDfltRotationThreshold(DtReal r)
inline void setDfltTimeThreshold(DtTime time)
inline void setDfltTranslationThreshold(DtReal t)
inline void setTimeThreshold(DtTime time)
inline DtTime timeThreshold() const
```

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file dentitydec.h:

class DtEntityStateDecoder

class DtEntityStateDecoder: Instances of DtEntityStateDecoder are objects capable of decoding EntityState PDUs into a DtEntityStateRepository

Public Methods

- **DtEntityStateDecoder** ([DtExerciseConn](#)* exConn)
constructor
- virtual **~DtEntityStateDecoder** ()
destructor
- **DtEntityStateDecoder** (const [DtEntityStateDecoder](#)& orig)
copy constructor
- [DtEntityStateDecoder](#)& **operator=** (const [DtEntityStateDecoder](#)& orig)
assignment operator
- virtual void **decode** (const [DtEntityStatePdu](#)& pdu, [DtEntityStateRepository](#)* stateRep)
decode a PDU into an EntityStateRepository

Protected Fields

- [DtExerciseConn](#)* myExConn

Protected Methods

- virtual void **decodeArtParts** (const [DtEntityStatePdu](#)& pdu, [DtEntityStateRepository](#)* stateRep)
decode the articulated parts portion of a PDU into an EntityStateRepository
 - virtual void **decodeAttachedParts** (const [DtEntityStatePdu](#)& pdu, [DtEntityStateRepository](#)* stateRep)
decode the attached parts portion of a PDU into an EntityStateRepository
-

Documentation

class DtEntityStateDecoder: Instances of DtEntityStateDecoder are objects capable of decoding EntityState PDUs into a DtEntityStateRepository

- **DtEntityStateDecoder** ([DtExerciseConn](#)* exConn)
constructor
- virtual **~DtEntityStateDecoder** ()
destructor
- **DtEntityStateDecoder** (const [DtEntityStateDecoder](#)& orig)

copy constructor

[DtEntityStateDecoder](#)& operator=(const **[DtEntityStateDecoder](#)**& orig)

assignment operator

virtual void decode(const **[DtEntityStatePdu](#)**& pdu, **[DtEntityStateRepository](#)*** stateRep)

decode a PDU into an EntityStateRepository

virtual void decodeArtParts(const **[DtEntityStatePdu](#)**& pdu, **[DtEntityStateRepository](#)*** stateRep)

decode the articulated parts portion of a PDU into an EntityStateRepository

virtual void decodeAttachedParts(const **[DtEntityStatePdu](#)**& pdu, **[DtEntityStateRepository](#)*** stateRep)

decode the attached parts portion of a PDU into an EntityStateRepository

[DtExerciseConn](#)* myExConn

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file dentityenc.h:

class DtEntityStateEncoder

Public Methods

- **DtEntityStateEncoder ()**
Constructor
- virtual [DtEntityStatePdu](#)* **stateMsg** (const [DtEntityStateRepository](#)& stateRep, const [DtEntityStateRepository](#)& asSeenByRemote)
- virtual void **setSendNeeded ()**
Force the generation of an entity state PDU next time stateMsg is called even if one would not otherwise be needed
- inline void **setSendNeeded ()**

Protected Fields

- [DtEntityStatePdu](#) **myEsPdu**
- int **myForceSendNeeded**

Protected Methods

- virtual [DtBoolean](#) **sendNeeded** (const [DtEntityStateRepository](#)& stateRep, const [DtEntityStateRepository](#)& asSeenByRemote)
Returns true if a PDU needs to be sent, (based on thresholds, etc) false otherwise
 - virtual void **fillOut** (const [DtEntityStateRepository](#)& stateRep, const [DtEntityStateRepository](#)& asSeenByRemote, [DtEntityStatePdu](#)* pdu)
Copy current ESR data to the pdu
 - virtual [DtBoolean](#) **artPartsNeedUpdating** (const [DtEntityStateRepository](#)& stateRep, const [DtEntityStateRepository](#)& asSeenByRemote)
Returns true if art part info is out of date, false otherwise
 - virtual [DtBoolean](#) **attPartsNeedUpdating** (const [DtEntityStateRepository](#)& stateRep, const [DtEntityStateRepository](#)& asSeenByRemote)
Returns true if art part info is out of date, false otherwise
 - virtual void **fillOutArtAndAttParts** (const [DtEntityStateRepository](#)& stateRep, const [DtEntityStateRepository](#)& asSeenByRemote, [DtEntityStatePdu](#)* pdu)
Copies current articulated and attached part info from the ESR to the PDU
-

Documentation

- `DtEntityStateEncoder()`
Constructor
- `virtual DtEntityStatePdu* stateMsg(const DtEntityStateRepository& stateRep, const DtEntityStateRepository& asSeenByRemote)`
- `virtual void setSendNeeded()`
Force the generation of an entity state PDU next time stateMsg is called even if one would not otherwise be needed
- `virtual DtBoolean sendNeeded(const DtEntityStateRepository& stateRep, const DtEntityStateRepository& asSeenByRemote)`
Returns true if a PDU needs to be sent, (based on thresholds, etc) false otherwise
- `virtual void fillOut(const DtEntityStateRepository& stateRep, const DtEntityStateRepository& asSeenByRemote, DtEntityStatePdu* pdu)`
Copy current ESR data to the pdu
- `virtual DtBoolean artPartsNeedUpdating( const DtEntityStateRepository& stateRep, const DtEntityStateRepository& asSeenByRemote)`
Returns true if art part info is out of date, false otherwise
- `virtual DtBoolean attPartsNeedUpdating( const DtEntityStateRepository& stateRep, const DtEntityStateRepository& asSeenByRemote)`
Returns true if art part info is out of date, false otherwise
- `virtual void fillOutArtAndAttParts( const DtEntityStateRepository& stateRep, const DtEntityStateRepository& asSeenByRemote, DtEntityStatePdu* pdu)`
Copies current articulated and attached part info from the ESR to the PDU
- `DtEntityStatePdu myEsPdu`
- `int myForceSendNeeded`
- `inline void setSendNeeded()`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file fedreader.h:

class DtFedReader

class DtFedReader: Instances of DtFedReader are used to read a FED file, and fill in a DtFedTarget with the information read

Public Methods

- **DtFedReader ()**
Constructor
- virtual **~DtFedReader ()**
Destructor
- **DtFedReader (const [DtFedReader](#)& orig)**
copy constructor
- [DtFedReader](#)& **operator= (const [DtFedReader](#)& orig)**
assignment operator
- virtual [DtFedReadStatus](#) **readFed (const char* fedFileName, [DtFedTarget](#)* fedTarget)**
Read the FED file fedFileName, filling in intMgr and objMgr

Protected Methods

- virtual int **findString (FILE* fp, const char* str)**
Returns 1 after reading a line containing str Returns 0 if hit EOF without finding such a line
 - virtual char* **readLine (FILE* fp)**
Read a line from the file
 - virtual int **readObjects (FILE* fp, [DtFedTarget](#)* fedTarget)**
Starting from the line after "(objects", read through the ")" that denotes the end of the objects definitions, filling in objMgr as we go
 - virtual int **readObjClass (FILE* fp, [DtFedTarget](#)* fedTarget, char* line, const char* withinClass)**
Read the description of a single object class, filling in objMgr with data read
 - virtual int **readInteractions (FILE* fp, [DtFedTarget](#)* fedTarget)**
Starting from the line after "(interactions", read through the ")" that denotes the end of the interactions definitions, filling in interMgr as we go
 - virtual int **readInterClass (FILE* fp, [DtFedTarget](#)* fedTarget, char* line, const char* withinClass)**
Read the description of a single interaction class, filling in interMgr with data read
 - virtual FILE* **openFile (const char* fedFileName)**
Open the file fedFileName
-

Documentation

class DtFedReader:

Instances of DtFedReader are used to read a FED file, and fill in a DtFedTarget with the information read.

- DtFedReader()**
 Constructor
- virtual ~DtFedReader()**
 Destructor
- DtFedReader(const [DtFedReader](#)& orig)**
 copy constructor
- [DtFedReader](#)& operator=(const [DtFedReader](#)& orig)**
 assignment operator
- virtual [DtFedReadStatus](#) readFed(const char* fedFileName, [DtFedTarget](#)* fedTarget)**
 Read the FED file fedFileName, filling in intMgr and objMgr
- virtual int findString(FILE* fp, const char* str)**
 Returns 1 after reading a line containing str Returns 0 if hit EOF without finding such a line
- virtual char* readLine(FILE* fp)**
 Read a line from the file. Returns NULL if hit EOF. Returns a ptr to static memory that is reused in subsequent calls.
- virtual int readObjects(FILE* fp, [DtFedTarget](#)* fedTarget)**
 Starting from the line after "(objects", read through the ")" that denotes the end of the objects definitions, filling in objMgr as we go. Returns 0 if hit EOF during read, 1 otherwise.
- virtual int readObjClass(FILE* fp, [DtFedTarget](#)* fedTarget, char* line, const char* withinClass)**
 Read the description of a single object class, filling in objMgr with data read. line should be the line that contains the "class" keyword and the class name. withinClass is the name of the class enclosing the class being read, or NULL if we are reading a base class. Returns 0 if hit EOF before finishing reading the class, 1 otherwise. Recursive calls read and process subclasses. Reads through the class-closing "0".
- virtual int readInteractions(FILE* fp, [DtFedTarget](#)* fedTarget)**
 Starting from the line after "(interactions", read through the ")" that denotes the end of the interactions definitions, filling in interMgr as we go. Returns 0 if hit EOF during read, 1 otherwise.
- virtual int readInterClass(FILE* fp, [DtFedTarget](#)* fedTarget, char* line, const char* withinClass)**
 Read the description of a single interaction class, filling in interMgr with data read. line should be the line that contains the "class" keyword and the class name. withinClass is the name of the class enclosing the class being read, or NULL if we are reading a base class. Returns 0 if hit EOF before finishing reading the class, 1 otherwise. Recursive calls read and process subclasses. Reads through the class-closing "0".
- virtual FILE* openFile(const char* fedFileName)**
 Open the file fedFileName

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file fedreader.h:

enum DtFedReadStatus

-
-  DtFedReadOK
 -  DtCouldNotOpenFed
 -  DtErrorReadingFed
-

Documentation

-  DtFedReadOK
-  DtCouldNotOpenFed
-  DtErrorReadingFed

In file fedtarget.h:

class DtFedTarget

class DtFedTarget: DtFedTarget is an abstract class representing some representation of FED file data that can be filled in by a DtFedReader

Public Methods

- **DtFedTarget ()**
Constructor
 - virtual **~DtFedTarget ()**
Destructor
 - **DtFedTarget (const [DtFedTarget](#)& orig)**
copy constructor
 - [DtFedTarget](#)& **operator=** (const [DtFedTarget](#)& orig)
assignment operator
 - virtual void **addInterClass** (const char* name, const char* baseName, [DtTransportType](#) transport, [DtOrderType](#) order)
Define a class
 - virtual void **addObjClass** (const char* name, const char* baseName)
Define a class
 - virtual void **addParameter** (const char* className, const char* paramName)
Define a parameter of a class
 - virtual void **addAttribute** (const char* className, const char* attributeName, [DtTransportType](#) transport, [DtOrderType](#) order)
Define an attribute of a class
-

Documentation

class DtFedTarget: DtFedTarget is an abstract class representing some representation of FED file data that can be filled in by a DtFedReader

- **DtFedTarget ()**
Constructor
- virtual **~DtFedTarget ()**
Destructor
- **DtFedTarget (const [DtFedTarget](#)& orig)**
copy constructor
- [DtFedTarget](#)& **operator=** (const [DtFedTarget](#)& orig)
assignment operator

● virtual void addInterClass(const char* name, const char* baseName, [DtTransportType](#) transport, [DtOrderType](#) order)

Define a class. Used when initializing from FED file.

● virtual void addObjClass(const char* name, const char* baseName)

Define a class. Used when initializing from FED file.

● virtual void addParameter(const char* className, const char* paramName)

Define a parameter of a class. Used when initializing from FED file.

● virtual void addAttribute(const char* className, const char* attributeName, [DtTransportType](#) transport, [DtOrderType](#) order)

Define an attribute of a class. Used when initializing from FED file.

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file fedtarget.h:

enum DtTransportType

 DtBest_Effort

 DtReliable

Documentation

 DtBest_Effort

 DtReliable

In file fedtarget.h:

enum DtOrderType

-
-  DtReceive
 -  DtTimestamp
-

Documentation

-  DtReceive
-  DtTimestamp

In file `elecemitpdu.h`:

class DtFundamentalParameterData

Public Fields

- float **frequency**
- float **frequencyRange**
- float **effectiveRadiatedPower**
- float **pulseRepetitionFrequency**
- float **pulseWidth**
- float **beamAzCenter**
- float **beamAzSweep**
- float **beamElCenter**
- float **beamElSweep**
- float **beamSweepSync**

Public Methods

- void **printData** () const
-

Documentation

- float **frequency**
- float **frequencyRange**
- float **effectiveRadiatedPower**
- float **pulseRepetitionFrequency**
- float **pulseWidth**
- float **beamAzCenter**
- float **beamAzSweep**
- float **beamElCenter**
- float **beamElSweep**

float beamSweepSync

void printData() const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file globidhlst.h:

class DtGlobaldHashlist

class DtGlobaldHashlist: Instances of DtGlobaldHashlist are hashlists, where you can store associations between DtGlobalObjectDesignators and arbitrary pieces of data

Public Methods

- **DtGlobaldHashlist** ([DtBoolean](#) intrusive = DtFalse)
default constructor
- virtual **~DtGlobaldHashlist** ()
destructor
- virtual [DtListItem](#)* **add** (const DtGlobalObjectDesignator& key, void* data)
Add an association between key and data to the list
- virtual void* **remove** (const DtGlobalObjectDesignator& key)
Remove the current association involving key from the list
- virtual void* **lookup** (const DtGlobalObjectDesignator& key) const
Lookup the data associated with key
- virtual void* **operator[]** (const DtGlobalObjectDesignator& key) const
- virtual const [DtList](#)* **list** () const
Get a DtList containing all of the void data that you have listed, regardless of keys*
- inline void* **operator[]** (const DtGlobalObjectDesignator& key) const

Public

- *In the event that an item with key already exists, then add will replace the data and move it to the location in list*

Protected Fields

- [DtList](#)* **myList**
- [DtHashlist](#)* **myHashlist**

Protected Methods

- **DtGlobaldHashlist** (const [DtGlobaldHashlist](#)& orig)
copy constructor
 - [DtGlobaldHashlist](#)& **operator=** (const [DtGlobaldHashlist](#)& orig)
assignment operator
-

Documentation

class DtGlobalIdHashlist: Instances of DtGlobalIdHashlist are hashlists, where you can store associations between DtGlobalObjectDesignators and arbitrary pieces of data

DtGlobalIdHashlist(DtBoolean intrusive = DtFalse)

default constructor. Pass DtTrue only if the data that you are listing are instances of a class derived from DtListedObject.

virtual ~DtGlobalIdHashlist()

destructor

In the event that an item with key already exists, then add will replace the data and move it to the location in list. If your data was allocated, do not add unless you know key is unique, or a memory leak will result.

virtual DtListItem* add(const DtGlobalObjectDesignator& key, void* data)

Add an association between key and data to the list

virtual void* remove(const DtGlobalObjectDesignator& key)

Remove the current association involving key from the list. Returns the data that was associated with key.

virtual void* lookup(const DtGlobalObjectDesignator& key) const

Lookup the data associated with key

virtual void* operator[] (const DtGlobalObjectDesignator& key) const

virtual const DtList* list() const

Get a DtList containing all of the void* data that you have listed, regardless of keys

DtGlobalIdHashlist(const DtGlobalIdHashlist& orig)

copy constructor

DtGlobalIdHashlist& operator=(const DtGlobalIdHashlist& orig)

assignment operator

DtList* myList

DtHashlist* myHashlist

inline void* operator[] (const DtGlobalObjectDesignator& key) const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file hashitem.h:

class DtHashItem

class DtHashItem: Instances of DtHashItem are used internally to a hashlist to associate a pointer to an item in a list with a key

Public Methods

	DtHashItem (const DtBaseHashKey & key, DtListItem * item)
	<i>default constructor</i>
 virtual	~DtHashItem ()
	<i>destructor</i>
	DtHashItem (const DtHashItem & orig)
	<i>copy constructor</i>
 DtHashItem &	operator= (const DtHashItem & orig)
	<i>assignment operator</i>
 virtual const DtBaseHashKey &	hashKey () const
	<i>get the key for the hashed item</i>
 virtual DtListItem *	listItem () const
	<i>get the list item</i>

Protected Fields

 DtBaseHashKey *	myKey
 DtListItem *	myItem

Documentation

class DtHashItem: Instances of DtHashItem are used internally to a hashlist to associate a pointer to an item in a list with a key

	DtHashItem (const DtBaseHashKey & key, DtListItem * item)
	default constructor
 virtual	~DtHashItem ()
	destructor
	DtHashItem (const DtHashItem & orig)
	copy constructor
 DtHashItem &	operator= (const DtHashItem & orig)
	assignment operator

- virtual const [DtBaseHashKey](#)& hashKey() const
get the key for the hashed item
 - virtual [DtListItem](#)* listItem() const
get the list item
 - [DtBaseHashKey](#)* myKey
 - [DtListItem](#)* myItem
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vllist.h:

class DtListedObj

This is a base class from which elements of a DtIntrusiveList must be derived

Inheritance:

DtListedObj

Public Methods

- **DtListedObj ()**
Constructor
- virtual **~DtListedObj ()**
Destructor
- void* **next () const**
Return next (previous) elements on the list, or NULL if hit the end (beginning) of the list
- void* **prev () const**
- void* **wrapNext () const**
Return next (previous) elements on the list, but wrap to the beginning (end) of the list when you hit the end (beginning)
- void* **wrapPrev () const**
- inline void* **next () const**
- inline void* **prev () const**
- inline void* **wrapNext () const**
- inline void* **wrapPrev () const**

Protected Fields

- [DtList](#)* **myList**
 - [DtListItem](#)* **myItem**
-

Documentation

This is a base class from which elements of a DtIntrusiveList must be derived. Previous and next pointers can be obtained directly from the DtListedObj, without going through the DtList to which it has been added.

- `DtListedObj()`
Constructor
- `virtual ~DtListedObj()`
Destructor
- `void* next() const`
Return next (previous) elements on the list, or NULL if hit the end (beginning) of the list
- `void* prev() const`
- `void* wrapNext() const`
Return next (previous) elements on the list, but wrap to the beginning (end) of the list when you hit the end (beginning). Only returns NULL when the list is totally empty.
- `void* wrapPrev() const`
- `DtList* myList`
- `DtListItem* myItem`
- `inline void* next() const`
- `inline void* prev() const`
- `inline void* wrapNext() const`
- `inline void* wrapPrev() const`

Direct child classes:

- [DtReflectedObjectCallbackInfo](#)
- [DtReflectedObject](#)
- [DtBaseReflectedEntity](#)
- [DtBaseReflectedAggregate](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file refobjcbinfo.h:

class DtReflectedObjectCallbackInfo: public DtListedObj

class DtReflectedObjectCallbackInfo: Instances of DtReflectedObjectCallbackInfo are used to contain the function and user data associated with a DtReflectedObject callback,

Inheritance:

DtReflectedObjectCallbackInfo - DtListedObj

Public Methods

	DtReflectedObjectCallbackInfo () <i>default constructor</i>
	DtReflectedObjectCallbackInfo (DtReflectedObjectCallbackFunction cb, void* userData) <i>constructor</i>
 virtual	~DtReflectedObjectCallbackInfo () <i>destructor</i>
	DtReflectedObjectCallbackInfo (const <u>DtReflectedObjectCallbackInfo</u> & orig) <i>copy constructor</i>
 <u>DtReflectedObjectCallbackInfo</u> &	operator= (const <u>DtReflectedObjectCallbackInfo</u> & orig) <i>assignment operator</i>
 int	operator== (const <u>DtReflectedObjectCallbackInfo</u> & orig) const <i>equality operators</i>
 int	operator!= (const <u>DtReflectedObjectCallbackInfo</u> & orig) const
 virtual void	setCallbackFunction (DtReflectedObjectCallbackFunction funct) <i>set/get the callback function</i>
 virtual DtReflectedObjectCallbackFunction	callbackFunction () const
 virtual void	setUserData (void* data) <i>set/get the user data</i>
 virtual void*	userData () const

Protected Fields

- `DtReflectedObjectCallbackFunction` **myCallbackFunction**
 - `void*` **myUserData**
-

Inherited from [DtListedObj](#):

Public Methods

- `void* next() const`
- `void* prev() const`
- `void* wrapNext() const`
- `void* wrapPrev() const`
- `inline void* next() const`
- `inline void* prev() const`
- `inline void* wrapNext() const`
- `inline void* wrapPrev() const`

Protected Fields

- `DtList* myList`
 - `DtListItem* myItem`
-

Documentation

class DtReflectedObjectCallbackInfo: Instances of DtReflectedObjectCallbackInfo are used to contain the function and user data associated with a DtReflectedObject callback,

- `DtReflectedObjectCallbackInfo()`
default constructor
- `DtReflectedObjectCallbackInfo( DtReflectedObjectCallbackFunction cb, void* userData)`
constructor
- `virtual ~DtReflectedObjectCallbackInfo()`
destructor
- `DtReflectedObjectCallbackInfo(const DtReflectedObjectCallbackInfo& orig)`
copy constructor
- `DtReflectedObjectCallbackInfo& operator=( const DtReflectedObjectCallbackInfo& orig)`

assignment operator

- `int operator==(const DtReflectedObjectCallbackInfo& orig) const`

equality operators

- `int operator!=(const DtReflectedObjectCallbackInfo& orig) const`

- `virtual void setCallbackFunction(DtReflectedObjectCallbackFunction funct)`

set/get the callback function

- `virtual DtReflectedObjectCallbackFunction callbackFunction() const`

- `virtual void setUserData(void* data)`

set/get the user data

- `virtual void* userData() const`

- `DtReflectedObjectCallbackFunction myCallbackFunction`

- `void* myUserData`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file dreflectedobj.h:

class DtReflectedObject: public DtListedObj

class DtReflectedObject: Instances of DtReflectedObject are used to represent a simulation entity reflected across a network (i

Inheritance:

DtReflectedObject - DtListedObj

Public Methods

	DtReflectedObject (DtExerciseConn * conn)
	<i>constructor</i>
 virtual	~DtReflectedObject ()
	<i>destructor</i>
 virtual DtStateRepository *	stateRep () const
	<i>Returns a pointer to the StateRepository - through which you can inspect current state information</i>
 virtual DtStateRepository *	sr () const
	<i>Synonym for stateRep()</i>
 virtual DtTime	lastSimTimeUpdated () const
	<i>Returns the value of simulation time at the time an update was last received for the entity</i>
 virtual void	processStateMessage (const DtStateMsg & msg)
	<i>Process an entity state PDU</i>
 virtual void	baseAddPostUpdateCallback (DtReflectedObjectCallbackFunction cb, void* userData)
	<i>Object state callback registration/deregistration</i>
 virtual void	baseRemovePostUpdateCallback (DtReflectedObjectCallbackFunction cb, void* userData)
 virtual DtReflectedObject *	nextObj () const
	<i>Get the next remote object</i>
 virtual DtReflectedObject *	prevObj () const
	<i>Get the previous remote object</i>
 virtual DtReflectedObject *	wrapNextObj () const
	<i>Get the next remote object</i>
 virtual DtReflectedObject *	wrapPrevObj () const
	<i>Get the previous remote object</i>
 virtual const DtBaseHashKey &	hashKey () const
	<i>Get a hashkey for this object</i>
 virtual const DtString &	idString () const
	<i>Get a string identification of this object</i>

Protected Fields

- [DtExerciseConn](#)* myConn
- [DtIntrusiveList](#) myStateUpdateCallbacks
- DtTime myLastSimTimeUpdated

Protected Methods

- **DtReflectedObject** (const [DtReflectedObject](#)& orig)
copy constructor
 - [DtReflectedObject](#)& **operator=** (const [DtReflectedObject](#)& orig)
assignment operator
 - virtual void **processPostUpdateCallbacks** ()
process the post update callbacks
-

Inherited from [DtListedObj](#):

Public Methods

- void* **next**() const
- void* **prev**() const
- void* **wrapNext**() const
- void* **wrapPrev**() const
- inline void* **next**() const
- inline void* **prev**() const
- inline void* **wrapNext**() const
- inline void* **wrapPrev**() const

Protected Fields

- [DtList](#)* myList
 - [DtListItem](#)* myItem
-

Documentation

class DtReflectedObject:

Instances of DtReflectedObject are used to represent a simulation entity reflected across a network (i.e. this is not an full simulation running on this machine).

- DtReflectedObject([DtExerciseConn](#)* conn)
 constructor
- virtual ~DtReflectedObject()
 destructor
- virtual [DtStateRepository](#)* stateRep() const
 Returns a pointer to the StateRepository - through which you can inspect current state information
- virtual [DtStateRepository](#)* sr() const
 Synonym for stateRep()
- virtual DtTime lastSimTimeUpdated() const
 Returns the value of simulation time at the time an update was last received for the entity
- virtual void processStateMessage(const [DtStateMsg](#)& msg)
 Process an entity state PDU
- virtual void baseAddPostUpdateCallback(DtReflectedObjectCallbackFunction cb, void* userData)
 Object state callback registration/deregistration
- virtual void baseRemovePostUpdateCallback(DtReflectedObjectCallbackFunction cb, void* userData)
- virtual [DtReflectedObject](#)* nextObj() const
 Get the next remote object
- virtual [DtReflectedObject](#)* prevObj() const
 Get the previous remote object
- virtual [DtReflectedObject](#)* wrapNextObj() const
 Get the next remote object. Will cycle back to the first remote object.
- virtual [DtReflectedObject](#)* wrapPrevObj() const
 Get the previous remote object. Will cycle back to the last remote object.
- virtual const [DtBaseHashKey](#)& hashKey() const
 Get a hashkey for this object
- virtual const [DtString](#)& idString() const
 Get a string identification of this object
- DtReflectedObject(const [DtReflectedObject](#)& orig)
 copy constructor
- [DtReflectedObject](#)& operator=(const [DtReflectedObject](#)& orig)
 assignment operator
- virtual void processPostUpdateCallbacks()
 process the post update callbacks
- [DtExerciseConn](#)* myConn

● [DtIntrusiveList](#) myStateUpdateCallbacks

● DtTime myLastSimTimeUpdated

Direct child classes:

[DtReflectedEntity](#)

[DtReflectedEmitterSystem](#)

[DtReflectedAggregate](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file `dtreflectedent.h`:

class DtReflectedEntity: public DtReflectedObject

class DtReflectedEntity: Instances of this class provide a protocol independant interface to a reflected entity in a DIS environment

Inheritance:

DtReflectedEntity - DtReflectedObject - DtListedObj

Public Methods

	DtReflectedEntity (DtExerciseConn * conn, const DtEntityIdentifier & id, const DtEntityType & type) <i>Constructor</i>
	~DtReflectedEntity () <i>Destructor</i>
	entityStateRep () const <i>Returns a pointer to the EntityStateRepository - through which you can inspect current state information</i>
	esr () const <i>Synonym for entityStateRep()</i>
	stateRep () const <i>Returns a pointer to the StateRepository - through which you can inspect current state information</i>
	decoder () <i>Returns a pointer to the Decoder being used to decode state messages into our ESR</i>
	entityId () const <i>Returns the entity identifier</i>
	id () const <i>Get the entity's id</i>
	localId () const <i>Get the entity's local id</i>
	globalId () const <i>Get the entity's object's global object designator - for DIS, it's the same as its DtEntityIdentifier</i>
	processStateMessage (const DtStateMsg & msg) <i>Process an entity state PDU</i>
	next () const <i>get the next and previous elements in the reflected entity list</i>
	prev () const

virtual DtReflectedEntity *	wrapNext () const <i>get the next and previous elements, wrap at the ends of the list</i>
virtual DtReflectedEntity *	wrapPrev () const
virtual void	addPostUpdateCallback (DtEntityCallbackFunction cb, void* userData) <i>Entity state callback registration/deregistration</i>
virtual void	removePostUpdateCallback (DtEntityCallbackFunction cb, void* userData)
virtual const DtBaseHashKey &	hashKey () const <i>Get a hashkey for this object</i>
virtual const DtString &	idString () const <i>Get a string identification of this object</i>
virtual void	processEntityState (const DtEntityStatePdu & pdu) <i>Process an entity state PDU</i>
inline const DtObjectId&	localId () const

Protected Fields

DtEntityIdentifier	myId
DtEntityStateRepository *	myEntityStateRep
DtEntityStateDecoder *	myEntityDecoder
DtObjectIdHashKey	myHashKey

Protected Methods

	DtReflectedEntity (const DtReflectedEntity & orig) <i>copy constructor</i>
DtReflectedEntity &	operator= (const DtReflectedEntity & orig) <i>assignment operator</i>

Inherited from [DtReflectedObject](#):

Public Methods

virtual DtStateRepository * sr () const
virtual DtTime lastSimTimeUpdated () const
virtual void baseAddPostUpdateCallback (DtReflectedObjectCallbackFunction cb, void* userData)
virtual void baseRemovePostUpdateCallback (DtReflectedObjectCallbackFunction cb, void* userData)
virtual DtReflectedObject * nextObj () const
virtual DtReflectedObject * prevObj () const
virtual DtReflectedObject * wrapNextObj () const

virtual [DtReflectedObject](#)* wrapPrevObj() const

Protected Fields

[DtExerciseConn](#)* myConn

[DtIntrusiveList](#) myStateUpdateCallbacks

DtTime myLastSimTimeUpdated

Protected Methods

virtual void processPostUpdateCallbacks()

Inherited from [DtListedObj](#):

Protected Fields

[DtList](#)* myList

[DtListItem](#)* myItem

Documentation

class DtReflectedEntity: Instances of this class provide a protocol independant interface to a reflected entity in a DIS environment

[DtReflectedEntity](#)([DtExerciseConn](#)* conn, const [DtEntityIdentifier](#)& id, const [DtEntityType](#)& type)

Constructor

virtual ~DtReflectedEntity()

Destructor

virtual [DtEntityStateRepository](#)* entityStateRep() const

Returns a pointer to the EntityStateRepository - through which you can inspect current state information

virtual [DtEntityStateRepository](#)* esr() const

Synonym for entityStateRep()

virtual [DtStateRepository](#)* stateRep() const

Returns a pointer to the StateRepository - through which you can inspect current state information. Implementation for abstract base class. Calls esr().

virtual [DtEntityStateDecoder](#)* decoder()

Returns a pointer to the Decoder being used to decode state messages into our ESR.

- virtual const [DtEntityIdentifier](#)& entityId() const
 Returns the entity identifier
- virtual const DtObjectId& id() const
 Get the entity's id. (For DIS, DtObjectId is DtEntityIdentifier).
- virtual const DtObjectId& localId() const
 Get the entity's local id. For DIS - same as id().
- virtual const DtGlobalObjectDesignator& globalId() const
 Get the entity's object's global object designator - for DIS, it's the same as its DtEntityIdentifier
- virtual void processStateMessage(const [DtStateMsg](#)& msg)
 Process an entity state PDU. Calls processEntityState().
- virtual [DtReflectedEntity](#)* next() const
 get the next and previous elements in the reflected entity list
- virtual [DtReflectedEntity](#)* prev() const
- virtual [DtReflectedEntity](#)* wrapNext() const
 get the next and previous elements, wrap at the ends of the list
- virtual [DtReflectedEntity](#)* wrapPrev() const
- virtual void addPostUpdateCallback(DtEntityCallbackFunction cb, void* userData)
 Entity state callback registration/deregistration
- virtual void removePostUpdateCallback(DtEntityCallbackFunction cb, void* userData)
- virtual const [DtBaseHashKey](#)& hashKey() const
 Get a hashkey for this object
- virtual const [DtString](#)& idString() const
 Get a string identification of this object
- virtual void processEntityState(const [DtEntityStatePdu](#)& pdu)
 Process an entity state PDU. Also calls down to base class processStateMessage().
- DtReflectedEntity(const [DtReflectedEntity](#)& orig)
 copy constructor
- [DtReflectedEntity](#)& operator=(const [DtReflectedEntity](#)& orig)
 assignment operator
- [DtEntityIdentifier](#) myId
- [DtEntityStateRepository](#)* myEntityStateRep
- [DtEntityStateDecoder](#)* myEntityDecoder
- [DtObjectIdHashKey](#) myHashKey
- inline const DtObjectId& localId() const

This class has no child classes.

[*Alphabetic index*](#) [*Hierarchy of classes*](#)

In file dreflemitrsys.h:

class DtReflectedEmitterSystem: public DtReflectedObject

class DtReflectedEmitterSystem: Instances of DtReflectedEmitterSystem represent EmitterSystem object simulated by remote applications

Inheritance:

DtReflectedEmitterSystem - [DtReflectedObject](#) - [DtListedObj](#)

Public Methods

	DtReflectedEmitterSystem (DtExerciseConn * conn, const DtGlobalObjectDesignator& emittingEntity, DtU8 systemNumber) <i>Constructor</i>
	~DtReflectedEmitterSystem () <i>destructor</i>
virtual	emitterSysRep () const <i>Get the emitter system's state repository</i>
virtual DtEmitterSystemRepository *	esr () const <i>Synonym for emitterSysRep()</i>
virtual DtEmitterSystemRepository *	stateRep () const <i>Returns a pointer to the StateRepository - through which you can inspect current state information</i>
virtual DtStateRepository *	
virtual void	processStateMessage (const DtStateMsg & msg) <i>Process an electronic emissions PDU Implementation for abstract base class</i>
virtual DtReflectedEmitterSystem *	next () const <i>Get the next remote emitterSystem</i>
virtual DtReflectedEmitterSystem *	prev () const <i>Get the previous remote emitterSystem</i>
virtual DtReflectedEmitterSystem *	wrapNext () const <i>Get the next remote emitterSystem</i>
virtual DtReflectedEmitterSystem *	wrapPrev () const <i>Get the previous remote emitterSystem</i>
virtual void	addPostUpdateCallback (DtEmitterSystemCallbackFunction cb, void* userData) <i>emitterSystem state callback registration/deregistration</i>
virtual void	removePostUpdateCallback (DtEmitterSystemCallbackFunction cb, void* userData)
virtual const DtBaseHashKey &	hashKey () const <i>Get a hashkey for this object</i>

virtual const [DtString](#)&

idString () const
Get a string identification of this object

Protected Fields

[DtEmitterSystemRepository](#) **myStateRepository**

[DtSystemHashKey](#) **myHashKey**

[DtString](#) **myIdString**

Protected Methods

DtReflectedEmitterSystem (const [DtReflectedEmitterSystem](#)& orig)
copy constructor

[DtReflectedEmitterSystem](#)& **operator=** (const [DtReflectedEmitterSystem](#)& orig)
assignment operator

Inherited from [DtReflectedObject](#):

Public Methods

virtual [DtStateRepository](#)* **sr**() const

virtual DtTime **lastSimTimeUpdated**() const

virtual void **baseAddPostUpdateCallback**(DtReflectedObjectCallbackFunction cb, void* userData)

virtual void **baseRemovePostUpdateCallback**(DtReflectedObjectCallbackFunction cb, void* userData)

virtual [DtReflectedObject](#)* **nextObj**() const

virtual [DtReflectedObject](#)* **prevObj**() const

virtual [DtReflectedObject](#)* **wrapNextObj**() const

virtual [DtReflectedObject](#)* **wrapPrevObj**() const

Protected Fields

[DtExerciseConn](#)* **myConn**

[DtIntrusiveList](#) **myStateUpdateCallbacks**

DtTime **myLastSimTimeUpdated**

Protected Methods

- virtual void `processPostUpdateCallbacks()`

Inherited from [DtListedObj](#):

Protected Fields

- [DtList](#)* `myList`
- [DtListItem](#)* `myItem`

Documentation

class DtReflectedEmitterSystem:

Instances of DtReflectedEmitterSystem represent EmitterSystem object simulated by remote applications. State data can be viewed through it's state repository.

- `DtReflectedEmitterSystem(DtExerciseConn* conn, const DtGlobalObjectDesignator& emittingEntity, DtU8 systemNumber)`

Constructor

- virtual `~DtReflectedEmitterSystem()`

destructor

- virtual `DtEmitterSystemRepository* emitterSysRep() const`

Get the emitter system's state repository

- virtual `DtEmitterSystemRepository* esr() const`

Synonym for emitterSysRep()

- virtual `DtStateRepository* stateRep() const`

Returns a pointer to the StateRepository - through which you can inspect current state information. Implementation for abstract base class. Calls esr().

- virtual void `processStateMessage(const DtStateMsg& msg)`

Process an electronic emissions PDU Implementation for abstract base class

- virtual `DtReflectedEmitterSystem* next() const`

Get the next remote emitterSystem

- virtual `DtReflectedEmitterSystem* prev() const`

Get the previous remote emitterSystem

- virtual `DtReflectedEmitterSystem* wrapNext() const`

Get the next remote emitterSystem. Will cycle back to the first remote emitterSystem.

- virtual [DtReflectedEmitterSystem](#)* wrapPrev() const
Get the previous remote emitterSystem. Will cycle back to the last remote emitterSystem.
- virtual void addPostUpdateCallback(DtEmitterSystemCallbackFunction cb, void* userData)
emitterSystem state callback registration/deregistration
- virtual void removePostUpdateCallback(DtEmitterSystemCallbackFunction cb, void* userData)
- virtual const [DtBaseHashKey](#)& hashKey() const
Get a hashkey for this object
- virtual const [DtString](#)& idString() const
Get a string identification of this object
- DtReflectedEmitterSystem(const [DtReflectedEmitterSystem](#)& orig)
copy constructor
- [DtReflectedEmitterSystem](#)& operator=(const [DtReflectedEmitterSystem](#)& orig)
assignment operator
- [DtEmitterSystemRepository](#) myStateRepository
- [DtSystemHashKey](#) myHashKey
- [DtString](#) myIdString

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file `dtreflectedagg.h`:

class DtReflectedAggregate: public DtReflectedObject

class DtReflectedAggregate: Instances of this class provide a protocol independant interface to a reflected aggregate in a DIS environment

Inheritance:

DtReflectedAggregate - [DtReflectedObject](#) - [DtListedObj](#)

Public Methods

	DtReflectedAggregate (DtExerciseConn * conn, const DtEntityIdentifier & id, const DtEntityType & type) <i>Constructor</i>
	~DtReflectedAggregate () <i>Destructor</i>
	aggregateStateRep () const <i>Returns a pointer to the AggregateStateRepository - through which you can inspect current state information</i>
	asr () const <i>Synonym for aggregateStateRep()</i>
	stateRep () const <i>Returns a pointer to the StateRepository - through which you can inspect current state information</i>
	decoder () <i>Returns a pointer to the Decoder being used to decode state messages into our ASR</i>
	aggregateID () const <i>Returns the aggregate identifier</i>
	id () const <i>Get the aggregate's id</i>
	globalId () const <i>Get the aggregate's object's global object designator - for DIS, it's the same as its DtEntityIdentifier</i>
	processStateMessage (const DtStateMsg & msg) <i>Process an aggregate state PDU Implementation for abstract base class</i>
	next () const <i>get the next and previous elements in the reflected aggregate list</i>
	prev () const
•	
• virtual	
• virtual	DtAggregateStateRepository *
• virtual	DtAggregateStateRepository *
• virtual	DtStateRepository *
• virtual	DtAggregateStateDecoder *
• virtual const	DtEntityIdentifier &
• virtual const	DtObjectId&
• virtual const	DtGlobalObjectDesignator&
• virtual void	
• virtual	DtReflectedAggregate *
• virtual	DtReflectedAggregate *

virtual DtReflectedAggregate *	wrapNext () const <i>get the next and previous elements, wrap at the ends of the list</i>
virtual DtReflectedAggregate *	wrapPrev () const
virtual void	addPostUpdateCallback (DtAggregateCallbackFunction cb, void* userData) <i>Entity state callback registration/deregistration</i>
virtual void	removePostUpdateCallback (DtAggregateCallbackFunction cb, void* userData)
virtual const DtBaseHashKey &	hashKey () const <i>Get a hashkey for this object</i>
virtual const DtString &	idString () const <i>Get a string identification of this object</i>
virtual void	processAggregateState (const DtAggregateStatePdu & pdu) <i>Process an entity state PDU</i>

Protected Fields

DtEntityIdentifier	myId
DtAggregateStateRepository *	myAggregateStateRep
DtAggregateStateDecoder *	myAggregateDecoder
DtObjectIdHashKey	myHashKey

Protected Methods

	DtReflectedAggregate (const DtReflectedAggregate & orig) <i>copy constructor</i>
DtReflectedAggregate &	operator= (const DtReflectedAggregate & orig) <i>assignment operator</i>

Inherited from [DtReflectedObject](#):

Public Methods

virtual DtStateRepository * sr () const
virtual DtTime lastSimTimeUpdated () const
virtual void baseAddPostUpdateCallback (DtReflectedObjectCallbackFunction cb, void* userData)
virtual void baseRemovePostUpdateCallback (DtReflectedObjectCallbackFunction cb, void* userData)
virtual DtReflectedObject * nextObj () const
virtual DtReflectedObject * prevObj () const
virtual DtReflectedObject * wrapNextObj () const

virtual [DtReflectedObject](#)* wrapPrevObj() const

Protected Fields

[DtExerciseConn](#)* myConn

[DtIntrusiveList](#) myStateUpdateCallbacks

DtTime myLastSimTimeUpdated

Protected Methods

virtual void processPostUpdateCallbacks()

Inherited from [DtListedObj](#):

Protected Fields

[DtList](#)* myList

[DtListItem](#)* myItem

Documentation

class DtReflectedAggregate: Instances of this class provide a protocol independant interface to a reflected aggregate in a DIS environment

[DtReflectedAggregate](#)([DtExerciseConn](#)* conn, const [DtEntityIdentifier](#)& id, const [DtEntityType](#)& type)

Constructor

virtual ~DtReflectedAggregate()

Destructor

virtual [DtAggregateStateRepository](#)* aggregateStateRep() const

Returns a pointer to the AggregateStateRepository - through which you can inspect current state information

virtual [DtAggregateStateRepository](#)* asr() const

Synonym for aggregateStateRep()

virtual [DtStateRepository](#)* stateRep() const

Returns a pointer to the StateRepository - through which you can inspect current state information. Implementation for abstract base class. Calls asr().

virtual [DtAggregateStateDecoder](#)* decoder()

Returns a pointer to the Decoder being used to decode state messages into our ASR.

- virtual const [DtEntityIdentifier](#)& aggregateID() const
 Returns the aggregate identifier
- virtual const DtObjectId& id() const
 Get the aggregate's id. (For DIS, DtObjectId is DtEntityIdentifier).
- virtual const DtGlobalObjectDesignator& globalId() const
 Get the aggregate's object's global object designator - for DIS, it's the same as its DtEntityIdentifier
- virtual void processStateMessage(const [DtStateMsg](#)& msg)
 Process an aggregate state PDU Implementation for abstract base class
- virtual [DtReflectedAggregate](#)* next() const
 get the next and previous elements in the reflected aggregate list
- virtual [DtReflectedAggregate](#)* prev() const
- virtual [DtReflectedAggregate](#)* wrapNext() const
 get the next and previous elements, wrap at the ends of the list
- virtual [DtReflectedAggregate](#)* wrapPrev() const
- virtual void addPostUpdateCallback(DtAggregateCallbackFunction cb, void* userData)
 Entity state callback registration/deregistration
- virtual void removePostUpdateCallback(DtAggregateCallbackFunction cb, void* userData)
- virtual const [DtBaseHashKey](#)& hashKey() const
 Get a hashkey for this object
- virtual const [DtString](#)& idString() const
 Get a string identification of this object
- virtual void processAggregateState(const [DtAggregateStatePdu](#)& pdu)
 Process an entity state PDU. Also calls down to base class processStateMessage().
- DtReflectedAggregate(const [DtReflectedAggregate](#)& orig)
 copy constructor
- [DtReflectedAggregate](#)& operator=(const [DtReflectedAggregate](#)& orig)
 assignment operator
- [DtEntityIdentifier](#) myId
- [DtAggregateStateRepository](#)* myAggregateStateRep
- [DtAggregateStateDecoder](#)* myAggregateDecoder
- [DtObjectIdHashKey](#) myHashKey

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file basereflent.h:

class DtBaseReflectedEntity: public DtListedObj

Inheritance:

DtBaseReflectedEntity - DtListedObj

Public Methods

 virtual	~DtBaseReflectedEntity () <i>destructor</i>
 DtBaseReflectedEntity &	operator= (const DtBaseReflectedEntity & orig) <i>assignment operator</i>
 virtual DtEntityStateRepository *	entityStateRep () const <i>Get the entity's state repository</i>
 virtual const DtObjectId&	id () const <i>Get the entity's identifier</i>
	DtBaseReflectedEntity (const DtObjectId& id) <i>constructor</i>
 inline const DtObjectId&	id () const

Protected Fields

 DtObjectId myObjectId

Protected Methods

 DtBaseReflectedEntity (const DtBaseReflectedEntity & orig) <i>copy constructor</i>
--

Protected

 <i>Constructors are protected since this is an abstract base class</i>
--

Inherited from [DtListedObj](#):

Public Methods

- void* **next**() const
- void* **prev**() const
- void* **wrapNext**() const
- void* **wrapPrev**() const
- inline void* **next**() const
- inline void* **prev**() const
- inline void* **wrapNext**() const
- inline void* **wrapPrev**() const

Protected Fields

- [DtList](#)* myList
 - [DtListItem](#)* myItem
-

Documentation

- virtual ~DtBaseReflectedEntity()
destructor
- [DtBaseReflectedEntity](#)& operator=(const [DtBaseReflectedEntity](#)& orig)
assignment operator
- virtual [DtEntityStateRepository](#)* entityStateRep() const
Get the entity's state repository
- virtual const DtObjectId& id() const
Get the entity's identifier
- Constructors are protected since this is an abstract base class
- DtBaseReflectedEntity(const DtObjectId& id)
constructor
- DtBaseReflectedEntity(const [DtBaseReflectedEntity](#)& orig)
copy constructor
- DtObjectId myObjectId

 `inline const DtObjectId& id() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file basereflagg.h:

class DtBaseReflectedAggregate: public DtListedObj

Inheritance:

DtBaseReflectedAggregate - DtListedObj

Public Methods

 virtual	~DtBaseReflectedAggregate () <i>destructor</i>
 <u>DtBaseReflectedAggregate</u> &	operator= (const <u>DtBaseReflectedAggregate</u> & orig) <i>assignment operator</i>
 virtual <u>DtAggregateStateRepository</u> *	aggregateStateRep () const <i>Get the aggregate's state repository</i>
 virtual const DtObjectId&	id () const <i>Get the aggregate's identifier</i>
	DtBaseReflectedAggregate (const DtObjectId& id) <i>constructor</i>

Protected Fields

 DtObjectId myObjectId

Protected Methods

 DtBaseReflectedAggregate (const <u>DtBaseReflectedAggregate</u> & orig) <i>copy constructor</i>

Protected

 <i>Constructors are protected since this is an abstract base class</i>
--

Inherited from [DtListedObj](#):

Public Methods

- void* **next**() const
- void* **prev**() const
- void* **wrapNext**() const
- void* **wrapPrev**() const
- inline void* **next**() const
- inline void* **prev**() const
- inline void* **wrapNext**() const
- inline void* **wrapPrev**() const

Protected Fields

- [DtList](#)* myList
 - [DtListItem](#)* myItem
-

Documentation

- virtual ~DtBaseReflectedAggregate()
destructor
 - [DtBaseReflectedAggregate](#)& operator=(const [DtBaseReflectedAggregate](#)& orig)
assignment operator
 - virtual [DtAggregateStateRepository](#)* aggregateStateRep() const
Get the aggregate's state repository
 - virtual const DtObjectId& id() const
Get the aggregate's identifier
 - Constructors are protected since this is an abstract base class
 - DtBaseReflectedAggregate(const DtObjectId& id)
constructor
 - DtBaseReflectedAggregate(const [DtBaseReflectedAggregate](#)& orig)
copy constructor
 - DtObjectId myObjectId
-

This class has no child classes.

[*Alphabetic index*](#) [*Hierarchy of classes*](#)

In file netstructs.h:

class DtNet32Vector

Public Methods

- **DtNet32Vector ()**
- **DtNet32Vector (const [DtVector](#) &vec)**
- [DtVector](#) **operator = (const [DtVector](#) &vec)**
- **operator DtVector () const**
- [DtNetFloat32](#)& **operator[] (int i)**
- [DtNetFloat32](#) **operator[] (int i) const**

Private Fields

- [DtNetFloat32](#) rep [3]
-

Documentation

- **DtNet32Vector ()**
 - **DtNet32Vector (const [DtVector](#) &vec)**
 - [DtVector](#) **operator = (const [DtVector](#) &vec)**
 - **operator DtVector () const**
 - [DtNetFloat32](#)& **operator[] (int i)**
 - [DtNetFloat32](#) **operator[] (int i) const**
 - [DtNetFloat32](#) rep [3]
-

This class has no child classes.

In file nettypes.h:

class DtNetFloat32

Network-order 32-bit IEEE-754 floating-point quantity

Public Fields

 [DtNetU32](#) bits

Public Methods

-  **DtNetFloat32** ()
 -  **DtNetFloat32** (DtFloat32)
 -  **DtNetFloat32** (const [DtNetFloat32](#)&)
 -  DtFloat32 **operator =** (DtFloat32)
 -  **operator DtFloat32** () const
-

Documentation

Network-order 32-bit IEEE-754 floating-point quantity

-  [DtNetU32](#) bits
 -  **DtNetFloat32** ()
 -  **DtNetFloat32** (DtFloat32)
 -  **DtNetFloat32** (const [DtNetFloat32](#)&)
 -  DtFloat32 **operator =** (DtFloat32)
 -  **operator DtFloat32** () const
-

This class has no child classes.

In file netstructs.h:

class DtNet64Vector

Public Methods

- **DtNet64Vector ()**
- **DtNet64Vector (const [DtVector](#) &vec)**
- [DtVector](#) **operator = (const [DtVector](#) &vec)**
- **operator DtVector () const**
- [DtNetFloat64](#)& **operator[] (int i)**
- [DtNetFloat64](#) **operator[] (int i) const**

Private Fields

- [DtNetFloat64](#) rep [3]
-

Documentation

- **DtNet64Vector ()**
 - **DtNet64Vector (const [DtVector](#) &vec)**
 - [DtVector](#) **operator = (const [DtVector](#) &vec)**
 - **operator DtVector () const**
 - [DtNetFloat64](#)& **operator[] (int i)**
 - [DtNetFloat64](#) **operator[] (int i) const**
 - [DtNetFloat64](#) rep [3]
-

This class has no child classes.

In file netstructs.h:

class DtNetEntityMarking

Public Fields

- DtNetU8 **DtCharacterSet**
- DtNetU8 **DtText** [DtMaxEntityMarkingLength]

Public

- #define **DtMaxEntityMarkingLength**
-

Documentation

- DtNetU8 DtCharacterSet
 - #define DtMaxEntityMarkingLength
 - DtNetU8 DtText [DtMaxEntityMarkingLength]
-

This class has no child classes.

In file netstructs.h:

class DtNetEulerAngles

Public Fields

- [DtNetFloat32](#) psi
- [DtNetFloat32](#) theta
Counter clockwise about z
- [DtNetFloat32](#) phi
Counter clockwise about y
- Counter clockwise about x

Public Methods

- DtNetEulerAngles ()
 - DtNetEulerAngles (const [DtTaitBryan](#) &tait)
 - [DtTaitBryan](#) operator = (const [DtTaitBryan](#) &tait)
 - operator DtTaitBryan () const
-

Documentation

- DtNetEulerAngles ()
 - DtNetEulerAngles (const [DtTaitBryan](#) &tait)
 - [DtTaitBryan](#) operator = (const [DtTaitBryan](#) &tait)
 - operator DtTaitBryan () const
 - [DtNetFloat32](#) psi
 - [DtNetFloat32](#) theta
Counter clockwise about z
 - [DtNetFloat32](#) phi
Counter clockwise about y
 - Counter clockwise about x
-

This class has no child classes.

[*Alphabetic index*](#) [*Hierarchy of classes*](#)

In file nettypes.h:

class DtNetInt16

Network-order signed 16-bit quantity

Public Fields

 DtNetInt8 **msbyte**

 DtNetInt8 **lsbyte**

Public Methods

 **DtNetInt16** ()

 **DtNetInt16** (DtInt16)

 **DtNetInt16** (const DtNetInt16&)

 DtInt16 **operator =** (DtInt16)

 **operator DtInt16** () const

Documentation

Network-order signed 16-bit quantity

 DtNetInt8 **msbyte**

 DtNetInt8 **lsbyte**

 **DtNetInt16** ()

 **DtNetInt16** (DtInt16)

 **DtNetInt16** (const DtNetInt16&)

 DtInt16 **operator =** (DtInt16)

 **operator DtInt16** () const

This class has no child classes.

In file nettypes.h:

class DtNetInt32

Network-order signed 32-bit quantity

Public Fields

-  **DtNetInt16 msword**
-  **DtNetInt16 lsword**
Most Significant Word

Public Methods

-  **DtNetInt32 ()**
Least Significant Word
 -  **DtNetInt32 (DtInt32)**
 -  **DtNetInt32 (const DtNetInt32&)**
 -  **DtInt32 operator = (DtInt32)**
 -  **operator DtInt32 () const**
-

Documentation

Network-order signed 32-bit quantity

-  **DtNetInt16 msword**
 -  **DtNetInt16 lsword**
Most Significant Word
 -  **DtNetInt32 ()**
Least Significant Word
 -  **DtNetInt32 (DtInt32)**
 -  **DtNetInt32 (const DtNetInt32&)**
 -  **DtInt32 operator = (DtInt32)**
 -  **operator DtInt32 () const**
-

This class has no child classes.

In file nettimestamp.h:

class DtNetTimeStamp

class DtNetTimeStamp: Instances of DtNetTimeStamp are network representations of a DIS time stamp

Public Methods

- **DtNetTimeStamp ()**
default constructor - creates an instance with a time of zero and timestamp type of relative
- **DtNetTimeStamp (DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)**
Creates an instance with the time set to time (specified in seconds) modulo an hour, and the timestamp type set to type
- **~DtNetTimeStamp ()**
destructor
- **DtNetTimeStamp (const [DtNetTimeStamp](#)& orig)**
copy constructor
- **[DtNetTimeStamp](#)& operator= (const [DtNetTimeStamp](#)& orig)**
assignment operator
- **operator DtU32 () const**
Cast to a DtU32 representing the whole timestamp Useful mainly for backwards compatibility for applications that used to cast DtNetTimeStamp to DtU32 before accessing it
- **DtTime time () const**
Get the time contained in the timestamp (in seconds)
- **void setTime (DtTime time)**
Get the time contained in the timestamp to time % an hour
- **void setType ([DtTimeStampType](#) type)**
Set/Get timestamp type
- **[DtTimeStampType](#) type () const**

Protected Fields

- **[DtNetU32](#) myRep**
 - **static double theHourFracPerHour**
 - **static double theHourPerHourFrac**
-

Documentation

class DtNetTimeStamp:

Instances of DtNetTimeStamp are network representations of a DIS time stamp. This class has NO VIRTUAL FUNCTIONS, since it is a DtNetType - whose exact byte layout is relied on by PDU classes, etc.

- **DtNetTimeStamp()**
default constructor - creates an instance with a time of zero and timestamp type of relative
 - **DtNetTimeStamp(DtTime time, [DtTimeStampType](#) type = DtTimeStampRelative)**
Creates an instance with the time set to time (specified in seconds) modulo an hour, and the timestamp type set to type
 - **~DtNetTimeStamp()**
destructor
 - **DtNetTimeStamp(const [DtNetTimeStamp](#)& orig)**
copy constructor
 - **[DtNetTimeStamp](#)& operator=(const [DtNetTimeStamp](#)& orig)**
assignment operator
 - **operator DtU32 () const**
Cast to a DtU32 representing the whole timestamp Useful mainly for backwards compatibility for applications that used to cast DtNetTimeStamp to DtU32 before accessing it
 - **DtTime time() const**
Get the time contained in the timestamp (in seconds). The value returned will be between 0.0 and 3600.0 seconds.
 - **void setTime(DtTime time)**
Get the time contained in the timestamp to time % an hour. time is specified in seconds.
 - **void setType([DtTimeStampType](#) type)**
Set/Get timestamp type
 - **[DtTimeStampType](#) type() const**
 - **[DtNetU32](#) myRep**
 - **static double theHourFracPerHour**
 - **static double theHourPerHourFrac**
-

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file nettypes.h:

class DtNetU16

Network-order unsigned 16-bit quantity

Public Fields

-  **DtNetU8 msbyte**
-  **DtNetU8 lsbyte**
Most Significant Byte

Public Methods

-  **DtNetU16 ()**
Least Significant Byte
 -  **DtNetU16 (DtU16)**
 -  **DtNetU16 (const DtNetU16&)**
 -  **DtU16 operator = (DtU16)**
 -  **operator DtU16 () const**
-

Documentation

Network-order unsigned 16-bit quantity

-  **DtNetU8 msbyte**
 -  **DtNetU8 lsbyte**
Most Significant Byte
 -  **DtNetU16 ()**
Least Significant Byte
 -  **DtNetU16 (DtU16)**
 -  **DtNetU16 (const DtNetU16&)**
 -  **DtU16 operator = (DtU16)**
 -  **operator DtU16 () const**
-

This class has no child classes.

In file nullfedamb.h:

class DtNullFederateAmbassador: public RTI FederateAmbassador

Inheritance:

DtNullFederateAmbassador

Public Methods

- **DtNullFederateAmbassador ()**
- virtual **~DtNullFederateAmbassador ()**
- * virtual void **synchronizationPointRegistrationSucceeded** (const char *label) throw(RTI::FederateInternalError)
////////////////////////////////
- virtual void **synchronizationPointRegistrationFailed** (const char *label) throw(RTI::FederateInternalError)
- virtual void **announceSynchronizationPoint** (const char *label, const char *tag) throw(RTI::FederateInternalError)
- virtual void **federationSynchronized** (const char *label) throw(RTI::FederateInternalError)
- virtual void **initiateFederateSave** (const char *label) throw(RTI::UnableToPerformSave,
RTI::FederateInternalError)
- virtual void **federationSaved** () throw(RTI::FederateInternalError)
- virtual void **federationNotSaved** () throw(RTI::FederateInternalError)
- virtual void **requestFederationRestoreSucceeded** (const char *label) throw(RTI::FederateInternalError)
- virtual void **requestFederationRestoreFailed** (const char *label, const char *reason) throw(
RTI::FederateInternalError)
- virtual void **federationRestoreBegun** () throw(RTI::FederateInternalError)
- virtual void **initiateFederateRestore** (const char *label, RTI::FederateHandle handle) throw(
RTI::SpecifiedSaveLabelDoesNotExist, RTI::CouldNotRestore, RTI::FederateInternalError)
- virtual void **federationRestored** () throw(RTI::FederateInternalError)
- virtual void **federationNotRestored** () throw(RTI::FederateInternalError)
- * virtual void **startRegistrationForObjectClass** (RTI::ObjectClassHandle theClass) throw(
RTI::ObjectClassNotPublished, RTI::FederateInternalError)
////////////////////////////////
- virtual void **stopRegistrationForObjectClass** (RTI::ObjectClassHandle theClass) throw(
RTI::ObjectClassNotPublished, RTI::FederateInternalError)
- virtual void **turnInteractionsOn** (RTI::InteractionClassHandle theHandle) throw(RTI::InteractionClassNotPublished,
RTI::FederateInternalError)

- virtual void **turnInteractionsOff** (RTI::InteractionClassHandle theHandle) throw(RTI::InteractionClassNotPublished, RTI::FederateInternalError)
- * virtual void **discoverObjectInstance** (RTI::ObjectHandle theObject, RTI::ObjectClassHandle theObjectClass, const char* theObjectName) throw(RTI::CouldNotDiscover, RTI::ObjectClassNotKnown, RTI::FederateInternalError)
////////////////////////////////
- virtual void **reflectAttributeValues** (RTI::ObjectHandle theObject, const RTI::AttributeHandleValuePairSet& theAttributes, const RTI::FedTime& theTime, const char *theTag, RTI::EventRetractionHandle theHandle) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::FederateOwnsAttributes, RTI::InvalidFederationTime, RTI::FederateInternalError)
- virtual void **reflectAttributeValues** (RTI::ObjectHandle theObject, const RTI::AttributeHandleValuePairSet& theAttributes, const char *theTag) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::FederateOwnsAttributes, RTI::FederateInternalError)
- virtual void **receiveInteraction** (RTI::InteractionClassHandle theInteraction, const RTI::ParameterHandleValuePairSet& theParameters, const RTI::FedTime& theTime, const char *theTag, RTI::EventRetractionHandle theHandle) throw(RTI::InteractionClassNotKnown, RTI::InteractionParameterNotKnown, RTI::InvalidFederationTime, RTI::FederateInternalError)
4
- virtual void **receiveInteraction** (RTI::InteractionClassHandle theInteraction, const RTI::ParameterHandleValuePairSet& theParameters, const char *theTag) throw(RTI::InteractionClassNotKnown, RTI::InteractionParameterNotKnown, RTI::FederateInternalError)
- virtual void **removeObjectInstance** (RTI::ObjectHandle theObject, const RTI::FedTime& theTime, const char *theTag, RTI::EventRetractionHandle theHandle) throw(RTI::ObjectNotKnown, RTI::InvalidFederationTime, RTI::FederateInternalError)
- virtual void **removeObjectInstance** (RTI::ObjectHandle theObject, const char *theTag) throw(RTI::ObjectNotKnown, RTI::FederateInternalError)
- virtual void **attributesInScope** (RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::FederateInternalError)
- virtual void **attributesOutOfScope** (RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::FederateInternalError)
- virtual void **provideAttributeValueUpdate** (RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeNotOwned, RTI::FederateInternalError)
- virtual void **turnUpdatesOnForObjectInstance** (RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotOwned, RTI::FederateInternalError)
- virtual void **turnUpdatesOffForObjectInstance** (RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotOwned, RTI::FederateInternalError)
- * virtual void **requestAttributeOwnershipAssumption** (RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& offeredAttributes, const char *theTag) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeAlreadyOwned, RTI::AttributeNotPublished, RTI::FederateInternalError)
////////////////////////////////
- virtual void **attributeOwnershipDivestitureNotification** (RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& releasedAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeNotOwned, RTI::AttributeDivestitureWasNotRequested, RTI::FederateInternalError)
- virtual void **attributeOwnershipAcquisitionNotification** (RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& securedAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeAcquisitionWasNotRequested, RTI::AttributeAlreadyOwned, RTI::AttributeNotPublished, RTI::FederateInternalError)
- virtual void **attributeOwnershipUnavailable** (RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeAlreadyOwned, RTI::AttributeAcquisitionWasNotRequested, RTI::FederateInternalError)
- virtual void **requestAttributeOwnershipRelease** (RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& candidateAttributes, const char *theTag) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeNotOwned, RTI::FederateInternalError)

- virtual void **confirmAttributeOwnershipAcquisitionCancellation** (RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeAlreadyOwned, RTI::AttributeAcquisitionWasNotCanceled, RTI::FederateInternalError)
- virtual void **informAttributeOwnership** (RTI::ObjectHandle theObject, RTI::AttributeHandle theAttribute, RTI::FederateHandle theOwner) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::FederateInternalError)
- virtual void **attributelsNotOwned** (RTI::ObjectHandle theObject, RTI::AttributeHandle theAttribute) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::FederateInternalError)
- virtual void **attributeOwnedByRTI** (RTI::ObjectHandle theObject, RTI::AttributeHandle theAttribute) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::FederateInternalError)
- * virtual void **timeRegulationEnabled** (const RTI::FedTime& theFederateTime) throw(RTI::InvalidFederationTime, RTI::EnableTimeRegulationWasNotPending, RTI::FederateInternalError)
////////////////////////////////
- virtual void **timeConstrainedEnabled** (const RTI::FedTime& theFederateTime) throw(RTI::InvalidFederationTime, RTI::EnableTimeConstrainedWasNotPending, RTI::FederateInternalError)
- virtual void **timeAdvanceGrant** (const RTI::FedTime& theTime) throw(RTI::InvalidFederationTime, RTI::TimeAdvanceWasNotInProgress, RTI::FederateInternalError)
- virtual void **requestRetraction** (RTI::EventRetractionHandle theHandle) throw(RTI::EventNotKnown, RTI::FederateInternalError)

Public

- * Federation Management **Services***
////////////////////////////////
- * Declaration Management **Services***
////////////////////////////////
- * Object Management **Services***
////////////////////////////////
- * Ownership Management **Services***
////////////////////////////////
- * Time Management **Services***
////////////////////////////////

Documentation

- DtNullFederateAmbassador()
- virtual ~DtNullFederateAmbassador()
- * Federation Management Services*
////////////////////////////////
- * virtual void synchronizationPointRegistrationSucceeded(const char *label) throw(RTI::FederateInternalError)
////////////////////////////////
- virtual void synchronizationPointRegistrationFailed(const char *label) throw(RTI::FederateInternalError)
- virtual void announceSynchronizationPoint(const char *label, const char *tag) throw(RTI::FederateInternalError)

```

virtual void federationSynchronized( const char *label) throw(
RTI::FederateInternalError)

virtual void initiateFederateSave( const char *label) throw( RTI::UnableToPerformSave,
RTI::FederateInternalError)

virtual void federationSaved() throw( RTI::FederateInternalError)

virtual void federationNotSaved() throw( RTI::FederateInternalError)

virtual void requestFederationRestoreSucceeded( const char *label) throw(
RTI::FederateInternalError)

virtual void requestFederationRestoreFailed( const char *label, const char *reason)
throw( RTI::FederateInternalError)

virtual void federationRestoreBegun() throw( RTI::FederateInternalError)

virtual void initiateFederateRestore( const char *label, RTI::FederateHandle handle)
throw( RTI::SpecifiedSaveLabelDoesNotExist, RTI::CouldNotRestore,
RTI::FederateInternalError)

virtual void federationRestored() throw( RTI::FederateInternalError)

virtual void federationNotRestored() throw( RTI::FederateInternalError)

* Declaration Management Services*
    //////////////////////////////////

* virtual void startRegistrationForObjectClass( RTI::ObjectClassHandle theClass) throw(
RTI::ObjectClassNotPublished, RTI::FederateInternalError)
    //////////////////////////////////

virtual void stopRegistrationForObjectClass( RTI::ObjectClassHandle theClass) throw(
RTI::ObjectClassNotPublished, RTI::FederateInternalError)

virtual void turnInteractionsOn( RTI::InteractionClassHandle theHandle) throw(
RTI::InteractionClassNotPublished, RTI::FederateInternalError)

virtual void turnInteractionsOff( RTI::InteractionClassHandle theHandle) throw(
RTI::InteractionClassNotPublished, RTI::FederateInternalError)

* Object Management Services*
    //////////////////////////////////

* virtual void discoverObjectInstance( RTI::ObjectHandle theObject,
RTI::ObjectClassHandle theObjectClass, const char* theObjectName) throw(
RTI::CouldNotDiscover, RTI::ObjectClassNotKnown, RTI::FederateInternalError)
    //////////////////////////////////

virtual void reflectAttributeValues( RTI::ObjectHandle theObject, const
RTI::AttributeHandleValuePairSet& theAttributes, const RTI::FedTime& theTime, const char
*theTag, RTI::EventRetractionHandle theHandle) throw( RTI::ObjectNotKnown,
RTI::AttributeNotKnown, RTI::FederateOwnsAttributes, RTI::InvalidFederationTime,
RTI::FederateInternalError)

virtual void reflectAttributeValues( RTI::ObjectHandle theObject, const
RTI::AttributeHandleValuePairSet& theAttributes, const char *theTag) throw(
RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::FederateOwnsAttributes,
RTI::FederateInternalError)

```

virtual void receiveInteraction(RTI::InteractionClassHandle theInteraction, const RTI::ParameterHandleValuePairSet& theParameters, const RTI::FedTime& theTime, const char *theTag, RTI::EventRetractionHandle theHandle) throw(RTI::InteractionClassNotKnown, RTI::InteractionParameterNotKnown, RTI::InvalidFederationTime, RTI::FederateInternalError)

4

virtual void receiveInteraction(RTI::InteractionClassHandle theInteraction, const RTI::ParameterHandleValuePairSet& theParameters, const char *theTag) throw(RTI::InteractionClassNotKnown, RTI::InteractionParameterNotKnown, RTI::FederateInternalError)

virtual void removeObjectInstance(RTI::ObjectHandle theObject, const RTI::FedTime& theTime, const char *theTag, RTI::EventRetractionHandle theHandle) throw(RTI::ObjectNotKnown, RTI::InvalidFederationTime, RTI::FederateInternalError)

virtual void removeObjectInstance(RTI::ObjectHandle theObject, const char *theTag) throw(RTI::ObjectNotKnown, RTI::FederateInternalError)

virtual void attributesInScope(RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::FederateInternalError)

virtual void attributesOutOfScope(RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::FederateInternalError)

virtual void provideAttributeValueUpdate(RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeNotOwned, RTI::FederateInternalError)

virtual void turnUpdatesOnForObjectInstance(RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotOwned, RTI::FederateInternalError)

virtual void turnUpdatesOffForObjectInstance(RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotOwned, RTI::FederateInternalError)

* Ownership Management Services*

////////////////////////////////

* virtual void requestAttributeOwnershipAssumption(RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& offeredAttributes, const char *theTag) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeAlreadyOwned, RTI::AttributeNotPublished, RTI::FederateInternalError)

////////////////////////////////

virtual void attributeOwnershipDivestitureNotification(RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& releasedAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeNotOwned, RTI::AttributeDivestitureWasNotRequested, RTI::FederateInternalError)

virtual void attributeOwnershipAcquisitionNotification(RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& securedAttributes) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeAcquisitionWasNotRequested, RTI::AttributeAlreadyOwned, RTI::AttributeNotPublished, RTI::FederateInternalError)

virtual void attributeOwnershipUnavailable(RTI::ObjectHandle theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown,

RTI::AttributeNotKnown, RTI::AttributeAlreadyOwned,
RTI::AttributeAcquisitionWasNotRequested, RTI::FederateInternalError)

virtual void requestAttributeOwnershipRelease(RTI::ObjectHandle theObject, const
RTI::AttributeHandleSet& candidateAttributes, const char *theTag) throw(
RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::AttributeNotOwned,
RTI::FederateInternalError)

virtual void confirmAttributeOwnershipAcquisitionCancellation(RTI::ObjectHandle
theObject, const RTI::AttributeHandleSet& theAttributes) throw(RTI::ObjectNotKnown,
RTI::AttributeNotKnown, RTI::AttributeAlreadyOwned,
RTI::AttributeAcquisitionWasNotCanceled, RTI::FederateInternalError)

virtual void informAttributeOwnership(RTI::ObjectHandle theObject,
RTI::AttributeHandle theAttribute, RTI::FederateHandle theOwner) throw(
RTI::ObjectNotKnown, RTI::AttributeNotKnown, RTI::FederateInternalError)

virtual void attributeIsNotOwned(RTI::ObjectHandle theObject, RTI::AttributeHandle
theAttribute) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown,
RTI::FederateInternalError)

virtual void attributeOwnedByRTI(RTI::ObjectHandle theObject, RTI::AttributeHandle
theAttribute) throw(RTI::ObjectNotKnown, RTI::AttributeNotKnown,
RTI::FederateInternalError)

* Time Management Services*

////////////////

* virtual void timeRegulationEnabled(const RTI::FedTime& theFederateTime) throw(
RTI::InvalidFederationTime, RTI::EnableTimeRegulationWasNotPending,
RTI::FederateInternalError)

////////////////

virtual void timeConstrainedEnabled(const RTI::FedTime& theFederateTime) throw(
RTI::InvalidFederationTime, RTI::EnableTimeConstrainedWasNotPending,
RTI::FederateInternalError)

virtual void timeAdvanceGrant(const RTI::FedTime& theTime) throw(
RTI::InvalidFederationTime, RTI::TimeAdvanceWasNotInProgress, RTI::FederateInternalError)

virtual void requestRetraction(RTI::EventRetractionHandle theHandle) throw(
RTI::EventNotKnown, RTI::FederateInternalError)

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file objidhashlst.h:

class DtObjectIdHashlist

class DtObjectIdHashlist: Instances of DtObjectIdHashlist are hashlists, where you can store associations between DtObjectIds and arbitrary pieces of data

Public Methods

- **DtObjectIdHashlist** ([DtBoolean](#) intrusive = DtFalse)
default constructor
- virtual **~DtObjectIdHashlist** ()
destructor
- virtual [DtListItem](#)* **add** (const DtObjectId& key, void* data)
Add an association between key and data to the list
- virtual void* **remove** (const DtObjectId& key)
Remove the current association involving key from the list
- virtual void* **lookup** (const DtObjectId& key) const
Lookup the data associated with key
- virtual void* **operator[]** (const DtObjectId& key) const
- virtual const [DtList](#)* **list** () const
Get a DtList containing all of the void data that you have listed, regardless of keys*
- inline void* **operator[]** (const DtObjectId& key) const

Public

- *In the event that an item with key already exists, then add will replace the data and move it to the location in list*

Protected Fields

- [DtList](#)* **myList**
- [DtHashlist](#)* **myHashlist**

Protected Methods

- **DtObjectIdHashlist** (const [DtObjectIdHashlist](#)& orig)
copy constructor
 - [DtObjectIdHashlist](#)& **operator=** (const [DtObjectIdHashlist](#)& orig)
assignment operator
-

Documentation

class DtObjectIdHashlist: Instances of DtObjectIdHashlist are hashlists, where you can store associations between DtObjectIds and arbitrary pieces of data

DtObjectIdHashlist(DtBoolean intrusive = DtFalse)

default constructor. Pass DtTrue only if the data that you are listing are instances of a class derived from DtListedObject.

virtual ~DtObjectIdHashlist()

destructor

In the event that an item with key already exists, then add will replace the data and move it to the location in list. If your data was allocated, do not add unless you know key is unique, or a memory leak will result.

virtual DtListItem* add(const DtObjectId& key, void* data)

Add an association between key and data to the list

virtual void* remove(const DtObjectId& key)

Remove the current association involving key from the list. Returns the data that was associated with key.

virtual void* lookup(const DtObjectId& key) const

Lookup the data associated with key

virtual void* operator[] (const DtObjectId& key) const

virtual const DtList* list() const

Get a DtList containing all of the void* data that you have listed, regardless of keys

DtObjectIdHashlist(const DtObjectIdHashlist& orig)

copy constructor

DtObjectIdHashlist& operator=(const DtObjectIdHashlist& orig)

assignment operator

DtList* myList

DtHashlist* myHashlist

inline void* operator[] (const DtObjectId& key) const

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file drefobjlist.h:

class DtReflectedObjectList

class DtReflectedObjectList: Instances of DtReflectedObjectList are list of instances of classes derived from DtReflectedObject

Inheritance:

DtReflectedObjectList

Public Methods

	DtReflectedObjectList (DtExerciseConn * exConn) <i>Constructor</i>
virtual	~DtReflectedObjectList () <i>destructor Warning: makes non-virtual call to removeAndDelete</i>
virtual DtReflectedObject *	firstObj () const <i>Get the first/last object in the list</i>
virtual DtReflectedObject *	lastObj () const
virtual int	count () const <i>Get number of objects</i>
virtual DtExerciseConn *	exerciseConn () const <i>Get my exerciseConn</i>
virtual void	addObjectAdditionCallback (DtReflectedObjectCallbackFunction cb, void* userData) <i>Object added callback registration/deregistration</i>
virtual void	removeObjectAdditionCallback (DtReflectedObjectCallbackFunction cb, void* userData)
virtual void	addObjectRemovalCallback (DtReflectedObjectCallbackFunction cb, void* userData) <i>Object removed callback registration/deregistration</i>
virtual void	removeObjectRemovalCallback (DtReflectedObjectCallbackFunction cb, void* userData)
virtual void	print () <i>Print the SRs of all objects</i>
virtual void	setTimeoutProcessing (DtBoolean onOff) <i>Set whether or not timeout processing occurs</i>
virtual void	setTimeoutInterval (DtTime time) <i>Set/Get the remote object timeout period</i>
virtual DtTime	timeoutInterval ()

Protected Fields

DtExerciseConn *	myExConn
DtIntrusiveList	myObjectAddedCallbacks
DtIntrusiveList	myObjectRemovedCallbacks
DtIntrusiveList	myObjectList
DtHashlist	myObjectHashlist
DtTime	myTimeoutInterval
DtBoolean	myTimeoutProcessingFlag

Protected Methods

	DtReflectedObjectList (const DtReflectedObjectList & orig) <i>copy constructor</i>
DtReflectedObjectList &	operator= (const DtReflectedObjectList & orig) <i>assignment operator</i>
virtual DtReflectedObject *	newReflectedObj (const DtStateMsg & msg) <i>Create a new object from a state message</i>
virtual DtReflectedObject *	processStateMessage (const DtStateMsg & msg) <i>process an incoming state message</i>
virtual void	doTimeouts () <i>Remove any remote objects which have not been updated during the timeout period</i>
virtual void	removeAndDelete (DtReflectedObject * obj, char* msg = NULL) <i>Remove the remote object from the list if found and delete it</i>
virtual void	objectAdded (DtReflectedObject * object) <i>Called after a remote object has been added to the list</i>
virtual DtBoolean	shouldBeTimedOut (DtReflectedObject * object) <i>Returns whether or not the object should be timed out</i>
void	emptyLists () <i>Empty the object list and the timeout list</i>
virtual void	processObjectAdditionCallbacks (DtReflectedObject * object) <i>Process object addition/removal callbacks</i>
virtual void	processObjectRemovalCallbacks (DtReflectedObject * object)
virtual const DtBaseHashKey &	hashKeyFromStateMessage (const DtStateMsg * msg) <i>Get a hash key from a state message</i>
virtual int	acceptPdu (const DtStateMsg &) const <i>Determines if a PDU should be processed or not</i>
static void	doTimeoutsPostDrainCb (void* usr) <i>calls doTimeouts</i>

Documentation

class DtReflectedObjectList:

Instances of DtReflectedObjectList are list of instances of classes derived from DtReflectedObject. A DtReflectedObjectList manages the comings and goings of its reflected objects based on discoveries and removes received from the exercise connection.

- `DtReflectedObjectList(DtExerciseConn* exConn)`
Constructor
- `virtual ~DtReflectedObjectList()`
destructor Warning: makes non-virtual call to removeAndDelete
- `virtual DtReflectedObject* firstObj() const`
Get the first/last object in the list
- `virtual DtReflectedObject* lastObj() const`
- `virtual int count() const`
Get number of objects
- `virtual DtExerciseConn* exerciseConn() const`
Get my exerciseConn
- `virtual void addObjectAdditionCallback( DtReflectedObjectCallbackFunction cb, void* userData)`
Object added callback registration/deregistration
- `virtual void removeObjectAdditionCallback( DtReflectedObjectCallbackFunction cb, void* userData)`
- `virtual void addObjectRemovalCallback( DtReflectedObjectCallbackFunction cb, void* userData)`
Object removed callback registration/deregistration
- `virtual void removeObjectRemovalCallback( DtReflectedObjectCallbackFunction cb, void* userData)`
- `virtual void print()`
Print the SRs of all objects
- `virtual void setTimeoutProcessing(DtBoolean onOff)`
Set whether or not timeout processing occurs. Defaults to off.
- `virtual void setTimeoutInterval(DtTime time)`
Set/Get the remote object timeout period
- `virtual DtTime timeoutInterval()`
- `DtReflectedObjectList(const DtReflectedObjectList& orig)`
copy constructor
- `DtReflectedObjectList& operator=(const DtReflectedObjectList& orig)`
assignment operator

- virtual [DtReflectedObject](#)* newReflectedObj(const [DtStateMsg](#)& msg)
 Create a new object from a state message
- virtual [DtReflectedObject](#)* processStateMessage(const [DtStateMsg](#)& msg)
 process an incoming state message
- virtual void doTimeouts()
 Remove any remote objects which have not been updated during the timeout period. Gets called automatically by a postDrainCallback registered by the constructor. Users should not need to call this.
- virtual void removeAndDelete([DtReflectedObject](#)* obj, char* msg = NULL)
 Remove the remote object from the list if found and delete it. Prints a message on deletion if specified.
- virtual void objectAdded([DtReflectedObject](#)* object)
 Called after a remote object has been added to the list. Defaults to do nothing. Subclasses can override this function to add functionality on addition.
- virtual [DtBoolean](#) shouldBeTimedOut([DtReflectedObject](#)* object)
 Returns whether or not the object should be timed out. Assumes that timeout processing is enabled.
- void emptyLists()
 Empty the object list and the timeout list. Non-virtual since meant to be called from destructor. Warning: makes non-virtual call to removeAndDelete
- virtual void processObjectAdditionCallbacks([DtReflectedObject](#)* object)
 Process object addition/removal callbacks
- virtual void processObjectRemovalCallbacks([DtReflectedObject](#)* object)
- virtual const [DtBaseHashKey](#)& hashKeyFromStateMessage(const [DtStateMsg](#)* msg)
 Get a hash key from a state message
- virtual int acceptPdu(const [DtStateMsg](#)&) const
 Determines if a PDU should be processed or not. Right now, this function returns 1.
- static void doTimeoutsPostDrainCb(void* usr)
 calls doTimeouts
- [DtExerciseConn](#)* myExConn
- [DtIntrusiveList](#) myObjectAddedCallbacks
- [DtIntrusiveList](#) myObjectRemovedCallbacks
- [DtIntrusiveList](#) myObjectList
- [DtHashlist](#) myObjectHashlist
- DtTime myTimeoutInterval
- [DtBoolean](#) myTimeoutProcessingFlag

Direct child classes:

[DtReflectedEntityList](#)

[DtReflectedEmitterSystemList](#)

[DtReflectedAggregateList](#)

[Alphabetic index](#) [Hierarchy of classes](#)

In file dreflentlist.h:

class DtReflectedEntityList: public DtReflectedObjectList

Inheritance:

DtReflectedEntityList - DtReflectedObjectList

Public Methods

- **DtReflectedEntityList** ([DtExerciseConn](#)* exConn, [DtBoolean](#) smoothCapable = DtFalse)
Constructor
- virtual **~DtReflectedEntityList** ()
Destructor
- virtual [DtReflectedEntity](#)* **lookup** (const DtObjectId& id) const
lookup an entity from its object ID
- virtual [DtReflectedEntity](#)* **first** () const
get the first and last elements of the entity list
- virtual [DtReflectedEntity](#)* **last** () const
- virtual void **addEntityAdditionCallback** (DtEntityCallbackFunction cb, void* userData)
Entity added callback registration/deregistration
- virtual void **removeEntityAdditionCallback** (DtEntityCallbackFunction cb, void* userData)
- virtual void **addEntityRemovalCallback** (DtEntityCallbackFunction cb, void* userData)
Entity removed callback registration/deregistration
- virtual void **removeEntityRemovalCallback** (DtEntityCallbackFunction cb, void* userData)
- virtual void **print** ()
Print the SRs of all objects
- virtual [DtReflectedEntity](#)* **processEntityState** (const [DtEntityStatePdu](#)& pdu)
process an incoming entity state pdu

Protected Fields

- [DtBoolean](#) mySmoothCapable
- [DtObjectIdHashKey](#)* myHashKey

Protected Methods

 DtReflectedEntityList &	operator= (const DtReflectedEntityList & orig) <i>Assignment operator is protected because it is not fully implemented</i>
	DtReflectedEntityList (const DtReflectedEntityList & orig) <i>Copy constructor is protected because it is not fully implemented</i>
 virtual DtReflectedObject *	newReflectedObj (const DtStateMsg & msg) <i>Create a new object from a state message</i>
 virtual DtReflectedObject *	processStateMessage (const DtStateMsg & msg) <i>process an incoming state message</i>
 virtual void	objectAdded (DtReflectedObject * obj) <i>Called after an entity is added and before addition callbacks are processed</i>
 virtual void	finalPdulsSetAction (DtReflectedEntity * entity) <i>Called when an entity sends its final PDU</i>
 virtual const DtBaseHashKey &	hashKeyFromStateMessage (const DtStateMsg * msg) <i>Get a hash key from a state message</i>
 virtual DtReflectedEntity *	newReflectedEntity (const DtEntityIdentifier & id, const DtEntityType & type) const <i>Create a new reflected entity</i>
 virtual void	entityAdded (DtReflectedEntity * ent) <i>Called after an entity is added and before addition callbacks are processed</i>
 static void	entityStateCb (DtEntityStatePdu * pdu, void* usr) <i>callback for the exercise connection</i>

Inherited from [DtReflectedObjectList](#):

Public Methods

-  virtual [DtReflectedObject](#)* **firstObj**() const
-  virtual [DtReflectedObject](#)* **lastObj**() const
-  virtual int **count**() const
-  virtual [DtExerciseConn](#)* **exerciseConn**() const
-  virtual void **addObjectAdditionCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
-  virtual void **removeObjectAdditionCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
-  virtual void **addObjectRemovalCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
-  virtual void **removeObjectRemovalCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
-  virtual void **setTimeoutProcessing**([DtBoolean](#) onOff)
-  virtual void **setTimeoutInterval**(DtTime time)
-  virtual DtTime **timeoutInterval**()

Protected Fields

- [DtExerciseConn](#)* myExConn
- [DtIntrusiveList](#) myObjectAddedCallbacks
- [DtIntrusiveList](#) myObjectRemovedCallbacks
- [DtIntrusiveList](#) myObjectList
- [DtHashlist](#) myObjectHashlist
- DtTime myTimeoutInterval
- [DtBoolean](#) myTimeoutProcessingFlag

Protected Methods

- virtual void **doTimeouts**()
 - virtual void **removeAndDelete**([DtReflectedObject](#)* obj, char* msg = NULL)
 - virtual [DtBoolean](#) **shouldBeTimedOut**([DtReflectedObject](#)* object)
 - void **emptyLists**()
 - virtual void **processObjectAdditionCallbacks**([DtReflectedObject](#)* object)
 - virtual void **processObjectRemovalCallbacks**([DtReflectedObject](#)* object)
 - virtual int **acceptPdu**(const [DtStateMsg](#)&) const
 - static void **doTimeoutsPostDrainCb**(void* usr)
-

Documentation

- **DtReflectedEntityList**([DtExerciseConn](#)* exConn, [DtBoolean](#) smoothCapable = DtFalse)
Constructor
- **virtual ~DtReflectedEntityList**()
Destructor
- **virtual [DtReflectedEntity](#)* lookup**(const DtObjectId& id) const
lookup an entity from its object ID
- **virtual [DtReflectedEntity](#)* first**() const
get the first and last elements of the entity list
- **virtual [DtReflectedEntity](#)* last**() const
- **virtual void addEntityAdditionCallback**(DtEntityCallbackFunction cb, void* userData)
Entity added callback registration/deregistration

- `virtual void removeEntityAdditionCallback( DtEntityCallbackFunction cb, void* userData)`
- `virtual void addEntityRemovalCallback( DtEntityCallbackFunction cb, void* userData)`
Entity removed callback registration/deregistration
- `virtual void removeEntityRemovalCallback( DtEntityCallbackFunction cb, void* userData)`
- `virtual void print()`
Print the SRs of all objects
- `virtual DtReflectedEntity* processEntityState(const DtEntityStatePdu& pdu)`
process an incoming entity state pdu. Then calls down to base class processStateMessage().
- `DtReflectedEntityList& operator=( const DtReflectedEntityList& orig)`
Assignment operator is protected because it is not fully implemented
- `DtReflectedEntityList(const DtReflectedEntityList& orig)`
Copy constructor is protected because it is not fully implemented
- `virtual DtReflectedObject* newReflectedObj(const DtStateMsg& msg)`
Create a new object from a state message. Calls newReflectedEntity() to create a new entity, and then tells the esr to use a smoother if smoothing is enabled.
- `virtual DtReflectedObject* processStateMessage(const DtStateMsg& msg)`
process an incoming state message. Calls processEntityState().
- `virtual void objectAdded(DtReflectedObject* obj)`
Called after an entity is added and before addition callbacks are processed. Calls entityAdded().
- `virtual void finalPduIsSetAction(DtReflectedEntity* entity)`
Called when an entity sends its final PDU. Currently, this removes and deletes the entity.
- `virtual const DtBaseHashKey& hashKeyFromStateMessage(const DtStateMsg* msg)`
Get a hash key from a state message
- `virtual DtReflectedEntity* newReflectedEntity(const DtEntityIdentifier& id, const DtEntityType& type) const`
Create a new reflected entity
- `virtual void entityAdded(DtReflectedEntity* ent)`
Called after an entity is added and before addition callbacks are processed. Right now, this function only calls DtInfo.
- `static void entityStateCb(DtEntityStatePdu* pdu, void* usr)`
callback for the exercise connection
- `DtBoolean mySmoothCapable`
- `DtObjectIdHashKey* myHashKey`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file drefemsyslist.h:

class DtReflectedEmitterSystemList: public DtReflectedObjectList

Inheritance:

DtReflectedEmitterSystemList - DtReflectedObjectList

Public Methods

	DtReflectedEmitterSystemList (DtExerciseConn * exConn) <i>Constructor</i>
 virtual	~DtReflectedEmitterSystemList () <i>Destructor</i>
 virtual DtReflectedEmitterSystem *	lookup (const DtGlobalObjectDesignator& emittingEntity, int systemId) const <i>lookup an emitter from its object ID and system ID</i>
 virtual DtReflectedEmitterSystem *	first () const <i>get the first and last elements of the entity list</i>
 virtual DtReflectedEmitterSystem *	last () const
 virtual void	addEmitterAdditionCallback (DtEmitterSystemCallbackFunction cb, void* userData) <i>Emitter added callback registration/deregistration</i>
 virtual void	removeEmitterAdditionCallback (DtEmitterSystemCallbackFunction cb, void* userData)
 virtual void	addEmitterRemovalCallback (DtEmitterSystemCallbackFunction cb, void* userData) <i>Emitter removed callback registration/deregistration</i>
 virtual void	removeEmitterRemovalCallback (DtEmitterSystemCallbackFunction cb, void* userData)
 virtual void	print () <i>Print the SRs of all objects</i>

Protected Methods

 DtReflectedEmitterSystemList &	operator= (const DtReflectedEmitterSystemList & orig) <i>Assignment operator is protected because it is not fully implemented</i>
	DtReflectedEmitterSystemList (const DtReflectedEmitterSystemList & orig) <i>Copy constructor is protected because it is not fully implemented</i>
 virtual DtReflectedObject *	newReflectedObj (const DtStateMsg & msg) <i>Create a new object from a state message</i>

- virtual [DtReflectedObject](#)* **processStateMessage** (const [DtStateMsg](#)& msg)
process an incoming state message
 - virtual void **objectAdded** ([DtReflectedObject](#)* obj)
Called after an entity is added and before addition callbacks are processed
 - virtual const [DtBaseHashKey](#)& **hashKeyFromStateMessage** (const [DtStateMsg](#)* msg)
Get a hash key from a state message
 - static void **emEmissionCb** ([DtElectromagneticEmissionPdu](#)* pdu, void* usr)
callback for the exercise connection
-

Inherited from [DtReflectedObjectList](#):

Public Methods

- virtual [DtReflectedObject](#)* **firstObj**() const
- virtual [DtReflectedObject](#)* **lastObj**() const
- virtual int **count**() const
- virtual [DtExerciseConn](#)* **exerciseConn**() const
- virtual void **addObjectAdditionCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
- virtual void **removeObjectAdditionCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
- virtual void **addObjectRemovalCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
- virtual void **removeObjectRemovalCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
- virtual void **setTimeoutProcessing**([DtBoolean](#) onOff)
- virtual void **setTimeoutInterval**(DtTime time)
- virtual DtTime **timeoutInterval**()

Protected Fields

- [DtExerciseConn](#)* myExConn
- [DtIntrusiveList](#) myObjectAddedCallbacks
- [DtIntrusiveList](#) myObjectRemovedCallbacks
- [DtIntrusiveList](#) myObjectList
- [DtHashlist](#) myObjectHashlist
- DtTime myTimeoutInterval
- [DtBoolean](#) myTimeoutProcessingFlag

Protected Methods

- virtual void **doTimeouts**()
 - virtual void **removeAndDelete**([DtReflectedObject](#)* obj, char* msg = NULL)
 - virtual [DtBoolean](#) **shouldBeTimedOut**([DtReflectedObject](#)* object)
 - void **emptyLists**()
 - virtual void **processObjectAdditionCallbacks**([DtReflectedObject](#)* object)
 - virtual void **processObjectRemovalCallbacks**([DtReflectedObject](#)* object)
 - virtual int **acceptPdu**(const [DtStateMsg](#)&) const
 - static void **doTimeoutsPostDrainCb**(void* usr)
-

Documentation

- [DtReflectedEmitterSystemList](#)([DtExerciseConn](#)* exConn)
Constructor
- virtual ~[DtReflectedEmitterSystemList](#)()
Destructor
- virtual [DtReflectedEmitterSystem](#)* **lookup**(const [DtGlobalObjectDesignator](#)& emittingEntity, int systemId) const
lookup an emitter from its object ID and system ID
- virtual [DtReflectedEmitterSystem](#)* **first**() const
get the first and last elements of the entity list
- virtual [DtReflectedEmitterSystem](#)* **last**() const
- virtual void **addEmitterAdditionCallback**([DtEmitterSystemCallbackFunction](#) cb, void* userData)
Emitter added callback registration/deregistration
- virtual void **removeEmitterAdditionCallback**([DtEmitterSystemCallbackFunction](#) cb, void* userData)
- virtual void **addEmitterRemovalCallback**([DtEmitterSystemCallbackFunction](#) cb, void* userData)
Emitter removed callback registration/deregistration
- virtual void **removeEmitterRemovalCallback**([DtEmitterSystemCallbackFunction](#) cb, void* userData)
- virtual void **print**()
Print the SRs of all objects
- [DtReflectedEmitterSystemList](#)& **operator=**(const [DtReflectedEmitterSystemList](#)& orig)

Assignment operator is protected because it is not fully implemented

- `DtReflectedEmitterSystemList(const DtReflectedEmitterSystemList& orig)`

Copy constructor is protected because it is not fully implemented

- `virtual DtReflectedObject* newReflectedObj(const DtStateMsg& msg)`

Create a new object from a state message

- `virtual DtReflectedObject* processStateMessage(const DtStateMsg& msg)`

process an incoming state message

- `virtual void objectAdded(DtReflectedObject* obj)`

Called after an entity is added and before addition callbacks are processed. Right now, this function only calls DtInfo.

- `virtual const DtBaseHashKey& hashKeyFromStateMessage(const DtStateMsg* msg)`

Get a hash key from a state message

- `static void emEmissionCb(DtElectromagneticEmissionPdu* pdu, void* usr)`

callback for the exercise connection

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

In file drefflagglist.h:

class DtReflectedAggregateList: public DtReflectedObjectList

Inheritance:

DtReflectedAggregateList - DtReflectedObjectList

Public Methods

	DtReflectedAggregateList (DtExerciseConn * exConn, DtBoolean smoothCapable = DtFalse) <i>Constructor</i>
	~DtReflectedAggregateList () <i>Destructor</i>
virtual	lookup (const DtObjectId& id) const <i>lookup an aggregate from its object ID</i>
virtual DtReflectedAggregate *	first () const <i>get the first and last elements of the aggregate list</i>
virtual DtReflectedAggregate *	last () const
virtual void	addAggregateAdditionCallback (DtAggregateCallbackFunction cb, void* userData) <i>Aggregate added callback registration/deregistration</i>
virtual void	removeAggregateAdditionCallback (DtAggregateCallbackFunction cb, void* userData)
virtual void	addAggregateRemovalCallback (DtAggregateCallbackFunction cb, void* userData) <i>Aggregate removed callback registration/deregistration</i>
virtual void	removeAggregateRemovalCallback (DtAggregateCallbackFunction cb, void* userData)
virtual void	print () <i>Print the SRs of all objects</i>
virtual DtReflectedAggregate *	processAggregateState (const DtAggregateStatePdu & pdu) <i>process an incoming aggregate state pdu</i>

Protected Fields

DtBoolean	mySmoothCapable
DtObjectIdHashKey *	myHashKey

Protected Methods

 DtReflectedAggregateList &	operator= (const DtReflectedAggregateList & orig) <i>Assignment operator is protected because it is not fully implemented</i>
	DtReflectedAggregateList (const DtReflectedAggregateList & orig) <i>Copy constructor is protected because it is not fully implemented</i>
 virtual DtReflectedObject *	newReflectedObj (const DtStateMsg & msg) <i>Create a new object from a state message</i>
 virtual DtReflectedObject *	processStateMessage (const DtStateMsg & msg) <i>process an incoming state message</i>
 virtual void	objectAdded (DtReflectedObject * obj) <i>Called after an aggregate is added and before addition callbacks are processed</i>
 virtual const DtBaseHashKey &	hashKeyFromStateMessage (const DtStateMsg * msg) <i>Get a hash key from a state message</i>
 virtual DtReflectedAggregate *	newReflectedAggregate (const DtEntityIdentifier & id, const DtEntityType & type) const <i>Create a new reflected aggregate</i>
 virtual void	aggregateAdded (DtReflectedAggregate * agg) <i>Called after an aggregate is added and before addition callbacks are processed</i>
 static void	aggregateStateCb (DtAggregateStatePdu * pdu, void* usr) <i>callback for the exercise connection</i>

Inherited from [DtReflectedObjectList](#):

Public Methods

-  virtual [DtReflectedObject](#)* **firstObj**() const
-  virtual [DtReflectedObject](#)* **lastObj**() const
-  virtual int **count**() const
-  virtual [DtExerciseConn](#)* **exerciseConn**() const
-  virtual void **addObjectAdditionCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
-  virtual void **removeObjectAdditionCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
-  virtual void **addObjectRemovalCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
-  virtual void **removeObjectRemovalCallback**(DtReflectedObjectCallbackFunction cb, void* userData)
-  virtual void **setTimeoutProcessing**([DtBoolean](#) onOff)
-  virtual void **setTimeoutInterval**(DtTime time)
-  virtual DtTime **timeoutInterval**()

Protected Fields

- [DtExerciseConn](#)* myExConn
- [DtIntrusiveList](#) myObjectAddedCallbacks
- [DtIntrusiveList](#) myObjectRemovedCallbacks
- [DtIntrusiveList](#) myObjectList
- [DtHashlist](#) myObjectHashlist
- DtTime myTimeoutInterval
- [DtBoolean](#) myTimeoutProcessingFlag

Protected Methods

- virtual void **doTimeouts**()
 - virtual void **removeAndDelete**([DtReflectedObject](#)* obj, char* msg = NULL)
 - virtual [DtBoolean](#) **shouldBeTimedOut**([DtReflectedObject](#)* object)
 - void **emptyLists**()
 - virtual void **processObjectAdditionCallbacks**([DtReflectedObject](#)* object)
 - virtual void **processObjectRemovalCallbacks**([DtReflectedObject](#)* object)
 - virtual int **acceptPdu**(const [DtStateMsg](#)&) const
 - static void **doTimeoutsPostDrainCb**(void* usr)
-

Documentation

- **DtReflectedAggregateList**([DtExerciseConn](#)* exConn, [DtBoolean](#) smoothCapable = DtFalse)
Constructor
- **virtual ~DtReflectedAggregateList**()
Destructor
- **virtual [DtReflectedAggregate](#)* lookup**(const DtObjectId& id) const
lookup an aggregate from its object ID
- **virtual [DtReflectedAggregate](#)* first**() const
get the first and last elements of the aggregate list
- **virtual [DtReflectedAggregate](#)* last**() const
- **virtual void addAggregateAdditionCallback**(DtAggregateCallbackFunction cb, void* userData)

Aggregate added callback registration/deregistration

virtual void removeAggregateAdditionCallback(DtAggregateCallbackFunction cb, void* userData)

virtual void addAggregateRemovalCallback(DtAggregateCallbackFunction cb, void* userData)

Aggregate removed callback registration/deregistration

virtual void removeAggregateRemovalCallback(DtAggregateCallbackFunction cb, void* userData)

virtual void print()

Print the SRs of all objects

virtual DtReflectedAggregate* processAggregateState(const DtAggregateStatePdu& pdu)
process an incoming aggregate state pdu. Then calls down to base class processStateMessage().

DtReflectedAggregateList& operator=(const DtReflectedAggregateList& orig)

Assignment operator is protected because it is not fully implemented

DtReflectedAggregateList(const DtReflectedAggregateList& orig)

Copy constructor is protected because it is not fully implemented

virtual DtReflectedObject* newReflectedObj(const DtStateMsg& msg)

Create a new object from a state message

virtual DtReflectedObject* processStateMessage(const DtStateMsg& msg)

process an incoming state message

virtual void objectAdded(DtReflectedObject* obj)

Called after an aggregate is added and before addition callbacks are processed. Right now, this function only calls DtInfo.

virtual const DtBaseHashKey& hashKeyFromStateMessage(const DtStateMsg* msg)

Get a hash key from a state message

virtual DtReflectedAggregate* newReflectedAggregate(const DtEntityIdentifier& id, const DtEntityType& type) const

Create a new reflected aggregate

virtual void aggregateAdded(DtReflectedAggregate* agg)

Called after an aggregate is added and before addition callbacks are processed. Right now, this function only calls DtInfo.

static void aggregateStateCb(DtAggregateStatePdu* pdu, void* usr)

callback for the exercise connection

DtBoolean mySmoothCapable

DtObjectIdHashKey* myHashKey

This class has no child classes.

In file vlutil.h:

endif class DtSelectParam

Public Methods

- **DtSelectParam** ()
- int **select** (DtTime timeout)
- void **addRead** (int fd)
- void **delRead** (int fd)
- int **canRead** (int fd)

Private Fields

- int **maxfd**
 - fd_set **read0**
 - fd_set **read**
-

Documentation

- **DtSelectParam** ()
 - int **select** (DtTime timeout)
 - void **addRead** (int fd)
 - void **delRead** (int fd)
 - int **canRead** (int fd)
 - int **maxfd**
 - fd_set **read0**
 - fd_set **read**
-

This class has no child classes.

In file tcpforwarder.h:

class DtTcpForwarder

Public Methods

- **DtTcpForwarder** (int port=DtDEFAULT_PORT)
- virtual **~DtTcpForwarder** ()
- void **sendToAll** (caddr_t packet, size_t size)
- void **readAndForward** (DtTime maxTimeOverAll, DtTime maxTimePerSocket)
- [DtTcpSocketMgr](#)* **socketMgr** ()

Private Fields

- [DtTcpSocketMgr](#)* **mySocketMgr**
 - [DtList](#)* **mySockList**
 - [DtTcpSocket](#)* **myBeginSocket**
 - [DtListItem](#)* **myBeginItem**
-

Documentation

- **DtTcpForwarder**(int port=DtDEFAULT_PORT)
 - virtual **~DtTcpForwarder**()
 - void **sendToAll**(caddr_t packet, size_t size)
 - void **readAndForward**(DtTime maxTimeOverAll, DtTime maxTimePerSocket)
 - [DtTcpSocketMgr](#)* **socketMgr**()
 - [DtTcpSocketMgr](#)* **mySocketMgr**
 - [DtList](#)* **mySockList**
 - [DtTcpSocket](#)* **myBeginSocket**
 - [DtListItem](#)* **myBeginItem**
-

This class has no child classes.

In file tcpsocketmgr.h:

class DtTcpSocketMgr

Public Fields

 [DtList](#)* mySockList

Public Methods

 **DtTcpSocketMgr** (int port=DtDEFAULT_PORT)

 virtual **~DtTcpSocketMgr** ()

 void **processConnections** ()

 int **isOk** ()

 [DtList](#)* **socketList** ()

Private Fields

 int **serverSockfd**

 int **myOk**

Documentation

 **DtTcpSocketMgr**(int port=DtDEFAULT_PORT)

 virtual **~DtTcpSocketMgr**()

 void **processConnections**()

 int **isOk**()

 [DtList](#)* **socketList**()

 int **serverSockfd**

 [DtList](#)* **mySockList**

 int **myOk**

This class has no child classes.

In file `elecemitpdu.h`:

class DtTrackJamTarget

Public Fields

 [DtEntityIdentifier](#) `entityId`
 `int` `emitterId`
 `int` `beamId`

Public Methods

 `void printData () const`

Documentation

 [DtEntityIdentifier](#) `entityId`
 `int emitterId`
 `int beamId`
 `void printData() const`

This class has no child classes.

In file utmview.h:

class DtUtmView

class DtUtmView: Instances of DtUtmView are used to create a UTM coordinate view of the geocentric coordinates in an Entity State Repository

Public Methods

	DtUtmView (DtBaseEntityStateRepository * esr) <i>constructor</i>
virtual	~DtUtmView () <i>destructor</i>
	DtUtmView (const DtUtmView & orig) <i>copy constructor</i>
DtUtmView &	operator= (const DtUtmView & orig) <i>assignment operator</i>
virtual void	setLocation (const DtUtmCoord & aLocation) <i>Set/Get the location in UTM coordinates</i>
virtual const DtUtmCoord &	location () const
virtual void	setVelocity (const DtUtmCoord & aVelocity) <i>Set/Get the velocity</i>
virtual const DtUtmCoord &	velocity () const
virtual void	setOrientation (const DtTaitBryan & orient) <i>Set/Get the orientation</i>
virtual const DtTaitBryan &	orientation () const
virtual DtConstDcm	bodyToLocal () const <i>Get the body to local coordinate matrix</i>
virtual void	setAcceleration (const DtUtmCoord & anAcceleration) <i>Set/Get the acceleration</i>
virtual const DtUtmCoord &	acceleration () const
virtual void	setRotationalVelocity (DtConstVector bodyRates) <i>Set/Get the rotational velocity</i>
virtual DtConstVector	rotationalVelocity () const
inline DtUtmView *	self () const

Protected Fields

DtBaseEntityStateRepository *	myESR
DtUtmCoord	myLocalLoc <i>used as return values by inspectors</i>

DtUtmCoord	myLocalVel
DtUtmCoord	myLocalAccel
DtTaitBryan	myLocalOrient
DtDcm	myBodyToLocal
DtDcm	myUtmToGeocMatrix
DtVector	myGeocLocForUtmToGeoc
DtDcm	myGeocToUtmMatrix
DtVector	myGeocLocForGeocToUtm

Protected Methods

DtUtmView*	self () const <i>Cast off const-ness for trivial internal change</i>
virtual DtConstDcm	utmToGeocMatrix (DtConstVector geocLoc) const <i>Get the matrix which transforms from UTM to geocentric</i>
virtual DtConstDcm	geocToUtmMatrix (DtConstVector geocLoc) const <i>Get the matrix which transforms form geocentric to UTM</i>

Documentation

class DtUtmView: Instances of DtUtmView are used to create a UTM coordinate view of the geocentric coordinates in an Entity State Repository

- [DtUtmView](#) ([DtBaseEntityStateRepository*](#) esr)
constructor
- virtual ~DtUtmView()
destructor
- DtUtmView(const [DtUtmView](#)& orig)
copy constructor
- [DtUtmView](#)& operator=(const [DtUtmView](#)& orig)
assignment operator
- virtual void setLocation(const [DtUtmCoord](#)& aLocation)
Set/Get the location in UTM coordinates
- virtual const [DtUtmCoord](#)& location() const
- virtual void setVelocity(const [DtUtmCoord](#)& aVelocity)
Set/Get the velocity
- virtual const [DtUtmCoord](#)& velocity() const

- `virtual void setOrientation(const DtTaitBryan& orient)`
Set/Get the orientation
- `virtual const DtTaitBryan& orientation() const`
- `virtual DtConstDcm bodyToLocal() const`
Get the body to local coordinate matrix. This is just a matrix representation of the euler angles returned by orientation().
- `virtual void setAcceleration(const DtUtmCoord& anAcceleration)`
Set/Get the acceleration
- `virtual const DtUtmCoord& acceleration() const`
- `virtual void setRotationalVelocity(DtConstVector bodyRates)`
Set/Get the rotational velocity
- `virtual DtConstVector rotationalVelocity() const`
- `DtUtmView* self() const`
Cast off const-ness for trivial internal change. Non-virtual since can't override on return type.
- `virtual DtConstDcm utmToGeocMatrix(DtConstVector geocLoc) const`
Get the matrix which transforms from UTM to geocentric
- `virtual DtConstDcm geocToUtmMatrix(DtConstVector geocLoc) const`
Get the matrix which transforms form geocentric to UTM
- `DtBaseEntityStateRepository* myESR`
- `DtUtmCoord myLocalLoc`
used as return values by inspectors. Do not necessarily hold current state.
- `DtUtmCoord myLocalVel`
- `DtUtmCoord myLocalAccel`
- `DtTaitBryan myLocalOrient`
- `DtDcm myBodyToLocal`
- `DtDcm myUtmToGeocMatrix`
- `DtVector myGeocLocForUtmToGeoc`
- `DtDcm myGeocToUtmMatrix`
- `DtVector myGeocLocForGeocToUtm`
- `inline DtUtmView* self() const`

This class has no child classes.

[Alphabetic index](#) [Hierarchy of classes](#)

#define	DtAbsNotifyObserver_H_
#define	DtAntennaPatternRecord_H_
#define	DtApproximator_H_
#define	DtArtPartSpec_H_
const int	DtMAX_PARAMS_PER_PART
typedef struct DtArtPartSpec int	partType
int	attachedType
int	params [DtMAX_PARAMS_PER_PART]
	DtArtPartSpec
const int	DtAttachedToBase
#define	DtNoArtParts((DtArtPartSpec*)
#define	DtNullArtPartSpec
#define	DtArtPart_H_
typedef struct DtInternalArtParam float	myValue
int	<i>Used only for internal representation of Art Param information</i>
int	myParamNum
DtTime	myNext
	myTimeSet
	DtInternalArtParam
#define	DtArtPartThresholder_H_
#define	DtAttachedPartList_H_
const int	DtNoAttachedParts [1]
	<i>This constant can be passed to a VR-Link function that expects a list of attached part stations to indicate an empty list</i>
#define	DtBaseExerciseConn_H_
#define	DtBaseHashKey_H_
#define	DtBaseMsg_H_
typedef struct DtBufferPlaceHolder*	DtBufferPtr
#define	DtUSE_INTERNAL_BUFFER_REP
#define	DtUSE_INTERNAL_BUFFER(DtBufferPtr(NULL)
#define	DtBaseReflectedEntity_H_
#define	DtBeamAntennaPatternRecord_H_
#define	DtCartesianView_H_
#define	_CLIENTSOCKET_H_
#define	DtCsFlagsDefault(DtCsFlagsReadWrite DtCsFlagsServerChooseDropMode)
#define	DtCsQDefaultSize
#define	COLLISION_H_
#define	_DATUMIDTYPE_H_
	<i>This file provides protocol independence for Collision Interactions</i>
#define	DtTYPE_NOT_DEFINED
#define	DtTYPE_BYTE_ARRAY

#define	DtTYPE_FLOAT32
#define	DtTYPE_FLOAT64
#define	DtTYPE_INT32
#define	DtTYPE_UNSIGNED32
#define	DtTYPE_STRING
int	DtDatumId2Type (DtDatumParam id)
int	DtAddDatumId2TypeMapping (DtDatumParam id, int dataType, int warn = 0)
#define	DtDatumRecordSet_H_
#define	DtNullDatumParam(0)
#define	DtNullVarDatumSpec
#define	DtFIXED
	<i>For backwards compatibility:</i>
#define	DtVAR
#define	DtDATUM_VALID
#define	DtDATUM_BAD_LENGTH
#define	DtDATUM_BAD_ID
#define	DtDATUM_BAD_INDEX
#define	DtDcm_H_
typedef const DtDcm&	DtConstDcm
typedef DtDcm&	DtDcmRef
#define	DtDeadReckoner_H_
#define	DtDesignatorRepository_H_
#define	DtDirectedRotRate_H_
#define	DtEmbeddedSystemRepository_H_
#define	DtEmitterSystemRepository_H_
#define	_ENUM2STRING_H_
const char*	DtEnumString (DtPduKind kind)
const char*	DtAddEnumString (DtPduKind kind, const char *str, int warn = 0)
#define	DtADD_ENUM_STRING(k)
#define	Euler_H_
void	DtEuler_to_BodyToRef (const DtTaitBryan &euler, DtDcmRef body2ref) <i>Given Euler angles between a body and a reference frame, produce the transformation matrix that converts from the body frame to the reference frame (or visa-versa)</i>
void	DtEuler_to_RefToBody (const DtTaitBryan &euler, DtDcmRef ref2body)
void	DtBodyToRef_to_Euler (DtConstDcm body2ref, DtTaitBryan *euler) <i>Given the transformation matrix that converts from a body frame to a reference frame (or visa-versa), produce the Euler angles between the body and the reference frames</i>
void	DtRefToBody_to_Euler (DtConstDcm ref2body, DtTaitBryan *euler)
void	DtEulerToEuler (const DtTaitBryan &in, DtConstDcm in2out, DtTaitBryan *out) <i>Given Euler angles with respect to reference frame "in", and the transformation matrix between frame "in" and frame "out", produce the Euler angles with respect to reference frame "out"</i>
#define	DtFedReader_H_

```

#define
#define
#define
void

void
typedef struct DtSpheroid DtFloat64

DtFloat64
typedef struct DtMapDatum DtSpheroid
DtFloat64
void

const DtMapDatum*
#define
#define
#define
const DtMapDatum
const DtMapDatum
const DtMapDatum
const DtMapDatum
#define
typedef DtGlobalIdHashlist
#define
#define
#define

```

Include obsolete old version for backwards compatibility

```

static const int

typedef DtHashlist
#define
#define
#define
#define
#define
#define
#define

```

```

DtFedTarget_H_
DtFilename_H_
DtGeodeticCoord_H_
DtGeocToGeod(DtConstVector in, DtGeodeticCoord& out)
Conversions between geocentric and geodetic: Use of these functions is discouraged in favor of DtGeodeticCoord member functions which perform conversions
DtGeodToGeoc(const DtGeodeticCoord& in, DtVectorRef out)
semiMajor
Spheroid
semiMinor
DtSpheroid
spheroid
datumShift[3]
DtMapDatum
DtUseMapDatum(const DtMapDatum* md)
Set and Get current map datum NOTE: map datum memory passed to DtUseMapDatum() must remain valid until next change of datum
DtCurrentMapDatum()
DtWGS84Spheroid
Here are a few of the most important spheroids; we should put all 13 in
DtInternationalSpheroid
DtClark1866Spheroid
DtED50
DtNASANAD27
DtNASBNAD27
DtCONUSNAD27
DtGlobalIdHashlist_H_
DtGlobalIdHashList
DtGroupSocket_H_
DtHashItem_H_
DtHashlist_H_
dfltNumBuckets
Not static member since g++ doesn't like to use static members as default arguments
DtHashList
INTERACTION_H_
DtIntHashKey_H_
This file is used as a convenience to group interaction concepts together
DtLgrEndPdu_H_
DtLgrEofActionPdu_H_
DtLgrHeartbeatPdu_H_
DtLgrPausePdu_H_
DtLgrPlayPdu_H_

```

#define	DtLgrPlayActionPdu_H_
#define	DtLgrPbFilePdu_H_
#define	DtLgrQuitPdu_H_
#define	DtLgrRecordActionPdu_H_
#define	DtLgrRecFilePdu_H_
#define	DtLgrRecordPdu_H_
#define	DtLgrSetTimePdu_H_
#define	DtLgrSkipTimePdu_H_
#define	DtLgrStartPdu_H_
#define	DtLgrStopPdu_H_
#define	DtLgrTimeoutPdu_H_
#define	DtLgrTimeScalePdu_H_
#define	_LibMatrix_h
#define	DtRad2Deg(x)
#define	DtDeg2Rad(x)
#define	DtEPico
	<i>A selection of constants useful as tolerance and threshold levels, etc</i>
#define	DtENano
#define	DtEMicro
#define	DtEMilli
#define	DtECenti
#define	DtEDeci
#define	DtEHalf
#define	DtENineTenths
#define	DtEOne
#define	DtEDeca
#define	DtEHecto
#define	DtEKilo
#define	DtEMega
#define	DtEGiga
#define	DtEq(x, y, e)
	<i>A macro to make use of these tolerances</i>
#define	DtSetVec(v, x, y, z)
	<i>Vector routines</i>
	DtVecInit (DtVectorRef v)
void	DtVecCopy (DtConstVector from, DtVectorRef to)
void	<i>Initialize to all zeros</i>
void	DtVecNormalize (DtConstVector v, DtVectorRef result)
int	DtVecCheck (DtConstVector v)
void	DtVecScale (DtConstVector v, DtReal scale_factor, DtVectorRef result)
	<i>Is this vector normalized?</i>
void	DtVecNeg (DtConstVector v1, DtVectorRef result)
void	DtVecAdd (DtConstVector v1, DtConstVector v2, DtVectorRef result)
	<i>result = -vec</i>

- void
- DtReal
- DtReal
- void
- void
- DtReal

DtVecSub(DtConstVector v1, DtConstVector v2, DtVectorRef result)

extern void
extern void
extern void
extern void
extern void
extern void
typedef struct DtReal

DtReal
DtReal
DtReal

void

void
void

void
void
void

Periodic modulus

double

double

double

double

DtDirectedRotRate

void

DtReal

#define

#define

void

#define

#define

void

void

void

void

[DtElrForm](#)(DtEParm axis, DtReal angle, DtEParm result)

[DtElrIdent](#)(DtEParm e)

[DtElrToMat](#)(DtEParm A_e_B, DtDcmRef A_c_B)

[DtElrTranspose](#)(DtEParm e, DtEParm result)

[DtDcmToElr](#)(DtConstDcm C, DtEParm e)

[DtVecElrTransform](#)(DtEParm v_A, DtEParm A_to_B, DtEParm v_B)

[DtQ0](#)

Quaternion routines

[DtQ1](#)

[DtQ2](#)

[DtQ3](#)

[DtQuaternion](#)

[DtQuatQuatMul](#)(DtQuaternion *qresult, DtQuaternion *DtQ1, DtQuaternion *DtQ2)

[DtDcmFromQuaternion](#)(DtDcmRef m, DtQuaternion *q)

[DtQuaternionDump](#)(DtQuaternion *q)

*void DtQuatScale(DtQuaternion *qresult, DtQuaternion *q, DtReal delta_t);*

[DtQuaternionFromTaitBryan](#)(DtQuaternion *q, const DtTaitBryan &tb)

[DtBodyToMat](#)(DtBodyRates av, DtConstDcm HcW, DtDcmRef cDot)

[DtTbToMat](#)(const DtTaitBryan &tb, DtDcmRef HcW)

[DtModPer](#)(double x, double incl, double excl)

*For x with period ABS(incl-excl), return DtModPerLo(x, incl, excl) if incl < excl
return DtModPerHi(x, excl, incl) if incl > excl*

[DtModPerZero](#)(double x, double per)

For x with period per, return 0 <= x < per

[DtModPerLo](#)(double x, double min, double max)

For x with period (max-min), return min <= x < max

[DtModPerHi](#)(double x, double min, double max)

For x with period (max-min), return min < x <= max

[DtDirectRotRate](#)(DtConstVector pqr)

directed rotation rates

[DtMakeRotationDcm](#)(DtDirectedRotRate drr, double dt, DtDcmRef m)

[DtMM_Mult](#)(DtConstDcm m1, DtConstDcm m2, DtDcmRef result,
generic_mm_multflag flag=TRANSPPOSE_NONE)

ADD

*Generic Vector Sum :Ans=A+s*B*

SUBTRACT

[DtSum](#)(DtConstVector a, DtConstVector b, DtReal s, DtVectorRef ans)

ZERO

generic initialize matrix routines

IDENTITY

[DtM_Init](#)(DtConstDcm input, DtDcmRef result, int flag)

[DtSetSym](#)(DtDcmRef v, DtReal aa, DtReal ab, DtReal ac, DtReal bb, DtReal bc, DtReal cc)

[DtVV_Product](#)(DtConstVector a, DtConstVector b, DtDcmRef ans)

[DtCreateCross](#)(DtConstVector r, DtDcmRef ans)

```

void
void
double
typedef DtReal

typedef
extern DtThreeIndex
double

double

double
double
double
double
#define
defined(__cplusplus), not __cplusplus, because
__SUNPRO_CC defines it to be empty

typedef unsigned char
typedef unsigned short
typedef unsigned int
typedef int
typedef unsigned long
typedef long int
typedef char
typedef short int
typedef float
typedef double
typedef DtFloat64
#define
#define
#define
#define
#define
#define
#define
#define
#define
#define
extern C int
extern char*
extern int

```

```

DtM(DtConstVector v, DtDcmRef m)
DtRotate(DtDcmRef ans, DtConstVector direction, double theta)
DtVecInfNorm(DtConstVector v1)
DtThreeIndex[3][3][3]
Unobvious Constants
DtReal(*DtThreeIndexPtr)
DtLevi_Civita
DtDistance(DtConstVector p1, DtConstVector p2)
additional Routines
DtDistSqr(DtConstVector p1, DtConstVector p2)
||p1-p2||_{2}
DtMax(double x, double y)
DtMin(double x, double y)
DtLimit(double x, double hi, double lo)

_MachineTypes_h

DtU8
DtU16
DtU32
DtInt32
DtU32
DtInt32
DtInt8
DtInt16
DtFloat32
DtFloat64
DtReal
__declspec( dllexport )
Windows requires extra work to export from DLLs
__declspec( dllimport )
__declspec( dllexport )
__declspec( dllimport )
__declspec( dllexport )
__declspec( dllimport )
__declspec( dllexport )
__declspec( dllimport )
__declspec( dllimport )
define
define
getopt(int argc, char** argv, char* opts)
optarg
optind

```

extern int

#define

#define

#define

#define

#define

For DtVector

For DtDcm

For DtTaitBryan

typedef DtVector

For DtDirectedRotRate and DtBodyRates

#define

#define

#define

#define

#define

int

caddr_t

*The functionality of the older constructor
DtNetSocket(const char *device, int port) has been
replaced by the call DtNetSocket(int port,
DtInetBroadcastOfDevice(char *device));*

#define

typedef DtU32

#define

typedef DtU8

typedef DtInt8

typedef DtU16

typedef DtU32

typedef DtInt16

typedef DtInt32

typedef DtFloat32

typedef DtFloat64

#define

[opterr](#)

MathTypes_H_

SQR(x)

MIN(x, y)

MAX(x, y)

LIMIT(low, x, high)

[DtBodyRates](#)

Angular rotation about body X, Y, Z axes

[DtRead](#)((DtU8)

DtWrite((DtU8)

DtReadWrite(DtRead | DtWrite)

DtQuiet((DtU8)

NULL

[datalen](#)

[data](#)

[GtQueueElement](#)

DtNetTimeStamp_H_

example device names are "ec0" and "le0"

[DtTimeStamp](#)

Host representation of a timeStamp

_NetTypes

[DtNetU8](#)

[DtNetInt8](#)

[DtNetU16](#)

[DtNetU32](#)

[DtNetInt16](#)

[DtNetInt32](#)

[DtNetFloat32](#)

[DtNetFloat64](#)

DtGlobalObjectDesignatorList_H_

class DtGlobalObjectDesignatorList: Instances of DtGlobalObjectDesignatorList are list of globally significant object identifiers

#define
#define
typedef DtObjectIdHashlist
#define
#define
extern FILE*
void
void
void
void
void
void
void
void
#define
void
char*

#define
#define
#define
typedef void

#define
typedef struct DtTime
DtViewControlPdu*

#define
#define

#define
#define
#define
#define

#define
#define
#define
#define

DtObjectIdHashKey_H_
DtObjectIdHashlist_H_
[DtObjectIdHashList](#)
[_PAIRSOCK_H](#)
[_PRINT_UTIL_H_](#)
[DtDumpFile](#)
[DtDumpInteger](#)(const char *label, long i)
[DtDumpFloat](#)(const char *label, double f)
[DtDumpString](#)(const char *label, const char *string)
[DtDumpEnum](#)(const char *label, const char *enumStr, int enumNum)
[DtDumpHexBytes](#)(const char *label, const void *bytes, int size)
[DtDumpHexWord](#)(const char *label, DtU32 word)
[DtDumpBoolean](#)(const char *label, DtBoolean val, int onlyIfTrue = 0)

[_ProcessControl_h](#)
[DtSleep](#)(DtTime sleep_period)
[DtPollInputLine](#)(void)
polls line-buffered standard input returns NULL if no input returns internal buffer of line if have input exits on ^D
[DtRadioReceiverRepository_H_](#)
[DtRadioTransmitterRepository_H_](#)
[DtReflectedObjectCallbackInfo_H_](#)
[\(*DtReflectedObjectCallbackFunction\)](#)(DtReflectedObject* obj, void* userData)
Callback function that takes a DtReflectedObject and user data
[SCRIPT_H_](#)
[t](#)
[pdu](#)
[StealthInstruction](#)

[SIGNAL_INTER_H_](#)
[DtSmoother_H_](#)
This file provides protocol independence for Signal Interactions
[DtSmoothOriWithDr_H_](#)

[DtSmoothPosWithDr_H_](#)
[DtSmoothPos_H_](#)
[_Socket_h](#)
DtSocket is a misnomer This provides net-level Packet IO Sometimes, it is implemented with sockets, sometimes not
[DtDEFAULT_PORT](#)
[DtUSE_DEFAULT_IP_ADDR](#)(DtInetAddr(0))
[DtSOCKET_PACKET](#)
[DtSOCKET_NO_PACKET](#)

#define
 #define
 DtSocket*
 #define
 #define
 #define
 #define
 #define
 #define
 #define
 void

 void

 #define
 #define
 #define
 void

 void
 void

 void
 DtConstDcm
 DtConstDcm
 double

 double
 void
 typedef struct double
 typedef struct double
 typedef struct double
 DtPolarDirection

 Some constants:
 const DtDegMinSec

DtSOCKET_SERVER_MSG
 DtSOCKET_FILTERED_PACKET
[DtNewDfltSocket](#)(int port, DtInetAddr destAddr = 0)
 DtString_H_
 DtStringHashKey_H_
 DtTaitBryan_H_
[DtTcpForwarder_H_](#)
 DtTcpSocket_H_
 DtTcpSocketMgr_H_
 DtThresholder_H_
[TopoCoord_H_](#)
[DtLatLon_to_GeocToTopo](#)(const DtGeodeticCoord& g, DtDcmRef geocToTopo)
The matrix produced by DtLatLon_to_GeocToTopo can be used to rotate a geocentric vector to a topographic vector like this: DtDcmVecMul(geocToTopo, Vgeoc, Vtopo);
[DtGeocToTopoTransform](#)(double latitude, double longitude, DtCoordTransform* geoc2topo)
This function can initialize a DtCoordTransform so that it can provide transformations between the geocentric coordinate system, and a particular topographic coordinate system defined by latitude and longitude
 DtTopoView_H_
 DtTimeStampType_H_
 DtUtmCoord_H_
[DtGeodToUtm](#)(const DtGeodeticCoord& in, DtUtmCoord& out)
Use of these functions is discouraged in favor of DtUtmCoord member functions which perform conversions
[DtUtmToGeod](#)(const DtUtmCoord& utm, DtGeodeticCoord& out)
[DtGeocToUtm](#)(DtConstVector in, DtUtmCoord& out)
Use of these functions is discouraged in favor of DtUtmCoord member functions which perform conversions
[DtUtmToGeoc](#)(const DtUtmCoord& in, DtVectorRef out)
[DtNedToEnuDcm](#)()
[DtEnuToNedDcm](#)()
[DtUtmGridDeclination](#)(const DtUtmCoord& utm)
Use of these functions is discouraged in favor of DtUtmCoord member functions which perform conversions
[DtUtmGridDeclination](#)(double lat, double lon)
[DtGridDeclinationDcm](#)(double gridDecl, DtDcmRef utm2ned)
[deg](#)
[min](#)
[sec](#)
[direction](#)
[DtDegMinSec](#)

[DtUtmLatRef1_1](#)
Origin of the IITSEC '92 terrain database

#define	DT_HAVE_NET_IF_H
#define	DT_HAVE_SYS_UIO_H
#define	DT_HAVE_DAEMON
#define	DT_HAVE_NUMERIC_CONSTANTS
#define	DT_HAVE_PROCESS_H
#define	DT_CANT_CAST_TO_CONST_2DARRAY
#define	REUSE_ALLOWED
#define	DT_HAVE_DRAND_48
#define	DT_NO_RANDOM
#define	DT_NO_RAND
#define	M_LN2
#define	M_PI
#define	M_SQRT2
#define	M_E
#define	M_LOG2E
#define	M_LOG10E
#define	M_LN10
#define	M_PI_2
#define	M_PI_4
#define	M_1_PI
#define	M_2_PI
#define	M_2_SQRTPI
typedef char*	caddr_t
#define	DtList_H_
#define	looptlist(theData, type, list)
#define	looptlist2(theData, type, list)
#define	_VLPRINT_H
#define	DtNIFatal
	<i>Notification Levels</i>
#define	DtNIWarn
#define	DtNIInfo
#define	DtNIVerbose
extern int	DtNotifyLevel
default is DtNIInfo	
void	DtFatalPerror (const char *s, ...)
void	DtWarnPerror (const char *s, ...)
void	DtFatalNetPerror (const char *s, ...)
void	DtWarnNetPerror (const char *s, ...)
void	DtWarn (const char *s, ...)
void	DtInfo (const char *s, ...)

void
void
#define
typedef DtReal
const double
const double
void

Simulation Time Calls

DtTime

void

DtTime

DtTime

DtTime

extern int
#define
#define
#define
#define
#define
#define
#define
#define
#define
#define
#define
#define
#define
inline int
#define
char*
char*
extern C void
int
inline void

[DtVerbose](#)(const char *s, ...)

[setFileOutput](#)(const char *filename, int con=0, int son=0)

VL_TIME_H_

[DtTime](#)

[DtHour2Sec](#)

[DtSec2Hour](#)

[DtTimeInit](#)(void)

Initialize all VR-Link time functionality This must be called before any other time-related functions

[DtGetSimTime](#)(void)

Returns the last value passed to DtSetSimTime - the current value of VR-Link simulation time

[DtIsSimTimeSet](#)(void)

Abort if DtSetSimTime was never called

[DtGetElapsedRealTime](#)(void)

Return real seconds elapsed since DtTimeInit was called

[DtGetApproximateElapsedRealTime](#)(void)

Returns the value of DtGetElapsedRealTime at the time that DtSetSimTime was last invoked

[DtAbsRealTime](#)(void)

Return current wallclock time - seconds since Jan 1, 1970

[DtUseSimTimeForApproxRealTime](#)

_VLUTIL_H_

[DtDELETE](#)(x)

DtISON(a, x)

DtISOFF(a, x)

DtTURNON(a, x)

DtTURNOFF(a, x)

DtMIN(x, y)

DtMAX(x, y)

DtLIMIT(low, x, hi)

DtABS(x)

DtSQR(x)

DtNUMBER(foo)

DtOFFSET(type, field)

DtSIZEOF(type, field)

[DtROUND_TO_MULT](#)(int num, int mult)

Rounds num up to a multiple of mult

DtATOI(s)

[DtDeg2DmsString](#)(double val, char *buff, int prec)

[DtDeg2DmsString](#)(double val)

[bzero](#)(char *b, int length)

[DtSelect](#)(int n, fd_set *i, fd_set *o, fd_set *e, DtTime t)

[DtClose](#)(int &fd)

```

else inline void
char*
char*
typedef DtU32
const DtInetAddr
const DtInetAddr
const char*
converts to internet dot notation to ulong in host order
DtInetAddr
DtInetAddr
DtInetAddr
DtInetAddr
DtInetAddr
DtInetAddr
int
#define
typedef const DtVector&
typedef DtVector&
#define
#define
#define
typedef DtAcknowledgePdu
typedef void
#define
typedef void
#define
#define
#define
typedef DtActionRequestPdu
typedef void
#define
#define
typedef DtActionResponsePdu
typedef void
#define
#define

```

```

DtClose(int &fd)
DtToLower(char* str)
Convert str to all lowercase
DtToUpper(char* str)
Convert str to all uppercase
DtInetAddr
----- inet utils -----
DtInetAddrAny
loopbackAddr
DtInetAddrString(DtInetAddr addr)
converts ulong to internet dot notation (returns internal static buffer)
DtStringToInetAddr(char *str)
DtStringToInetAddr(const char *str)
DtInetBroadcastOfDevice(const char *device)
DtInetAddrOfDevice(const char *device)
DtNetMaskOfDevice(const char *device)
DtFirstHostInetAddr()
DtIsMulticastAddr(DtInetAddr addr)
DtVector_H_
DtConstVector
DtVectorRef
ACK_INTERACTION_H_
DtAcknowledgePdu_H_
This file provides protocol independence for Acknowledge Interactions
DtAcknowledgeStaticSize
DtAcknowledgeInteraction
(*DtAcknowledgeInteractionCb)(DtAcknowledgeInteraction *pdu, void *usr)
DtAcknowledgeRPdu_H_
(*DtAcknowledgeRPduCb)(DtAcknowledgeRPdu* pdu, void* usr)
DtAcknowledgeRPdu callback function signature
DtActionRequestPdu_H_
DtActionRequestStaticSize
DtActionRequestStaticSize
DtActionRequestInteraction
(*DtActionRequestInteractionCb)(DtActionRequestInteraction *pdu, void *usr)
DtActionResponseStaticSize
DtActionResponseStaticSize
DtActionResponseInteraction
(*DtActionResponseInteractionCb)(DtActionResponsePdu *pdu, void *usr)
ACTION_REQUEST_INTERACTION_H_
DtActionRequestRPdu_H_
This file provides protocol independence for ActionRequest Interactions

```

```

typedef void

#define

#define

typedef void

#define

#define

#define

#define

#define

#define

#define

typedef void

#define

typedef void

#define

#define

#define

#define

#define

const int

typedef DtCollisionPdu

typedef void

#define

typedef void

#define

#define

typedef DtCommentPdu

typedef void

typedef DtCommentPdu

#define

typedef void

#define

typedef void

#define

```

```

(\*DtActionRequestRPduCb)(DtActionRequestRPdu* pdu, void* usr)
DtActionRequestRPdu callback function signature

ACTION\_RESPONSE\_H

DtActionResponseRPdu_H_
This file provides protocol independence for ActionResponse Interactions
(\*DtActionResponseRPduCb)(DtActionResponseRPdu* pdu, void* usr)
DtActionResponseRPdu callback function signature

DtSilentAggregateList\_H

DtAggregateEntityDecoder_H_

DtAggregateEntityEncoder_H_

DtSilentEntityList_H_

AGGREGATE_PUB_H_

DtAggregateStateRepository_H_
This file provides protocol independence for local aggregates

DtAggregateStatePdu_H_
(\*DtAggregateStatePduCb)(DtAggregateStatePdu* pdu, void* usr)
DtAggregateStatePdu callback function signature

DtArealObjectStatePdu_H_
(\*DtArealObjectStatePduCb)( DtArealObjectStatePdu* pdu, void* usr)
DtArealObjectStatePdu callback function signature

DtArtPartList_H_

DtBaseEntityDecoder\_H

DtBaseEntityEncoder_H_

DtBaseEntityStateRepository_H_

DtBaseReflectedAggregate\_H

_COLLPDU_H_

DtCollisionStaticSize

DtCollisionInteraction

(\*DtCollisionInteractionCb)(DtCollisionInteraction *inter, void *usr)

DtCollisionElasticPdu_H_
(\*DtCollisionElasticPduCb)(DtCollisionElasticPdu* pdu, void* usr)
DtCollisionElasticPdu callback function signature

COMMENT_H_

DtCommentPdu_H_
This file provides protocol independence for Comment Interactions

DtCommentStaticSize

DtCommentInteraction

(\*DtCommentPduCb)(DtCommentPdu *pdu, void *usr)

DtMessagePdu

DtCommentRPdu_H_
(\*DtCommentRPduCb)(DtCommentRPdu* pdu, void* usr)
DtCommentRPdu callback function signature

DtCreateEntityRPdu\_H

(\*DtCreateEntityRPduCb)(DtCreateEntityRPdu* pdu, void* usr)
DtCreateEntityRPdu callback function signature

CREATEENTINTER\_H

```

```

#define
#define
typedef DtCreateEntityPdu
typedef void
#define
#define
#define
static inline const DtObjectId&
#define
#ifdef __GNUC__ inline const DtObjectId&
#define
#define
#define
typedef DtDataPdu
typedef void
#define
#define
typedef void
#define
typedef void
#define
typedef void
#define
#define
#define
#define
static inline const DtObjectId&
static inline const DtEntityType&
#define
#define
inline const DtEntityType&
#define
#define
#define
#define
typedef void

```

```

DtCreateEntityPdu_H_
This file provides protocol independence for CreateEntity Interactions
DtCreateStaticSize
DtCreateEntityInteraction
(*DtCreateEntityPduCb)(DtCreateEntityPdu *pdu, void *usr)
DtAggregateStateDecoder_H_
DtAggregateStateEncoder_H_
DtDisAggregatePublisher_H_
gFindOwnId()
gFindOwnId()
gFindOwnId()
DATA_H_
DtDataPdu_H_
This file provides protocol independence for Comment Interactions
DtDataStaticSize
DtDataInteraction
(*DtDataPduCb)(DtDataPdu *pdu, void *usr)
DATA_QUERY_H_
DtDataQueryRPdu_H_
This file provides protocol independence for ActionResponse Interactions
(*DtDataQueryRPduCb)(DtDataQueryRPdu* pdu, void* usr)
DtDataQueryRPdu callback function signature
DtDataRPdu_H_
(*DtDataRPduCb)(DtDataRPdu* pdu, void* usr)
DtDataRPdu callback function signature
DtDatumMap_H_
(*DtGlobalSetFcn)(const DtSetDataPdu &pdu, int fixedOrVar, int index, void
*usr)
(*DtGlobalGetFcn)(DtDataPdu *pdu, int fixedOrVar, int index, void *usr)
DtStateDecoderFactory_H_
DtEntityStateDecoder_H_
DtEntityEncoder_H_
DtDisEntityPublisher_H_
gFindOwnId()
gGuiseSameAsType()
gFindOwnId()
gGuiseSameAsType()
gGuiseSameAsType()
DtDesignatorPdu_H_
DtDesignatorStaticSizeVERS5
DtDesignatorStaticSizeVERS4
DtDesignatorStaticSize
DtDesignatorStaticSize
(*DtDesignatorPduCb)(DtDesignatorPdu *pdu, void *usr)

```

```
#define
#define
#define
typedef DtDetonationPdu
typedef void
#define
#define
const DtExerciseld
typedef int
#define
#define
#define
#define
#define
#define
#define
#define
#define
#define
const DtPduKind
const DtPduKind
const DtPduKind
const DtPduKind
const DtPduKind
const DtPduKind
const DtPduKind
const DtPduKind
#define
#define
typedef DtRadioDetailedModulationAmplitude
typedef DtDetailedModulationUnmodulated
typedef DtEncodingClass
const DtEncodingClass
const DtEncodingClass
const DtEncodingClass
const DtEncodingClass
#define
```

DETONATION_INTERACTION_H_

_detonationPdu_h
This file provides protocol independence for Detonation Interactions

DtDetonateStaticSize

[**DtDetonationInteraction**](#)

[**\(*DtDetonationInteractionCb\)**](#)(DtDetonationInteraction *inter, void *usr)

DtDisExerciseConn_H_

DtMaxExerciseld

[**DtAnyExercise**](#)

[**\(*DtPredicate\)**](#)(DtPdu *pdu, void *arg)

DtNET_READ_SUCCESS

DtNET_READ_NO_PACKETS

DtNET_READ_BAD_PACKET

DtNET_READ_SERVER_MSG

DtNET_READ_SIZE_MISMATCH

DtPROCESS_PDU_SUCCESS

DtPROCESS_PDU_WRONG_EXID

DtPROCESS_PDU_NO_PDU

DtNET_READ_FILTERED_PACKET

[**DtDInteraction_H_**](#)

_DISENUMS_H_

[**DtNPCollisionPduKind**](#)

[**DtNPFieldPduKind**](#)

[**DtViewControlPduKind**](#)

[**DtMOPMigratorPduKind**](#)

[**DtNPSurfaceContactPduKind**](#)

[**DtGenericPduKind**](#)

[**DtLgrControlPduKind**](#)

DtViewControlKind
backwards compatible

DtRadioOther

[**DtRadioDetailedModulation**](#)

[**DetailedModulationUnmodulated**](#)

[**DtRadioSignalEncoding**](#)

[**DtEncodedVoice**](#)

[**DtRawBinary**](#)

[**DtApplicationSpecific**](#)

[**DtPreRecordedVoice**](#)

DtEXTEND_BIT_SHIFT_ENUM(attr, val)
This macro creates definitions that look like this: const DtAppearAttributeBitShift DtAppDensityBitShift = DtAppearAttributeBitShift(16);

[**DtEXTEND_BIT_SHIFT_ENUM**](#)(ActiveState, 23)

[**DtEXTEND_BIT_SHIFT_ENUM**](#)(Launcher, 16)

DtEXTEND_BIT_SHIFT_ENUM(CamoType, 17)

DtEXTEND_BIT_SHIFT_ENUM(Concealed, 19)

DtEXTEND_BIT_SHIFT_ENUM(Tent, 24)

DtEXTEND_BIT_SHIFT_ENUM(Ramp, 25)

DtEXTEND_BIT_SHIFT_ENUM(AfterBurner, 16)

DtEXTEND_BIT_SHIFT_ENUM(LaunchFlash, 16)

DtEXTEND_BIT_SHIFT_ENUM(LifeformState, 16)

DtEXTEND_BIT_SHIFT_ENUM(Weapon1, 24)

DtEXTEND_BIT_SHIFT_ENUM(Weapon2, 26)

DtEXTEND_BIT_SHIFT_ENUM(Density, 16)

DtEXTEND_BIT_SHIFT_ENUM(MinefieldType, 0)

Minefield appearance

DtEXTEND_BIT_SHIFT_ENUM(MinefieldActiveStatus, 2)

DtEXTEND_BIT_SHIFT_ENUM(MinefieldLane, 3)

DtEXTEND_BIT_SHIFT_ENUM(MinefieldState, 13)

DtEXTEND_BIT_SHIFT_ENUM(ObjectPercentComplete, 0)

Point, Linear, and Areal Object appearance

DtEXTEND_BIT_SHIFT_ENUM(ObjectDamage, 8)

DtEXTEND_BIT_SHIFT_ENUM(ObjectPredistributed, 10)

DtEXTEND_BIT_SHIFT_ENUM(ObjectState, 11)

DtEXTEND_BIT_SHIFT_ENUM(ObjectSmoking, 12)

DtEXTEND_BIT_SHIFT_ENUM(ObjectFlaming, 13)

Point object specific appearance

DtEXTEND_BIT_SHIFT_ENUM(ObjectAbatisBreach, 0)

This is for the log crib, abatis, vehicle defilade, and infantry fighting position breach appearance

DtEXTEND_BIT_SHIFT_ENUM(ObjectBurstOpacity, 0)

DtEXTEND_BIT_SHIFT_ENUM(ObjectBurstSize, 8)

DtEXTEND_BIT_SHIFT_ENUM(ObjectBurstHeight, 16)

DtEXTEND_BIT_SHIFT_ENUM(ObjectBurstNumOfBursts, 24)

DtEXTEND_BIT_SHIFT_ENUM(ObjectBurstChemicalSmoke, 30)

DtEXTEND_BIT_SHIFT_ENUM(ObjectCraterSize, 0)

DtEXTEND_BIT_SHIFT_ENUM(ObjectRibbonBridgeNumOfSegments, 0)

Linear object specific appearance

DtEXTEND_BIT_SHIFT_ENUM(ObjectTankDitchBreach, 0)

This is for the tank ditch and concertina wire breach appearance

DtEXTEND_BIT_SHIFT_ENUM(ObjectTankDitchBreachLength, 16)

DtEXTEND_BIT_SHIFT_ENUM(ObjectTankDitchBreachLocation, 24)

DtEXTEND_BIT_SHIFT_ENUM(ObjectExhaustSmokeOpacity, 0)

DtEXTEND_BIT_SHIFT_ENUM(ObjectExhaustSmokeAttached, 8)

DtEXTEND_BIT_SHIFT_ENUM(ObjectExhaustSmokeChemical, 9)

DtEXTEND_BIT_SHIFT_ENUM(ObjectMinefieldLaneMarkerVisibleSide, 0)

DtEXTEND_BIT_SHIFT_ENUM(ObjectMinefieldBreach, 0)

Areal object specific appearance

DtEXTEND_BIT_SHIFT_ENUM(ObjectMinefieldMineCount, 16)

DtEXTEND_ATTR_ENUM(attr, val)

This macro creates definitions that look like this: const DtAppearAttribute DtAppDensity = DtAppearAttribute(16 << DtAppDensityBitShift);

#define

[DtEXTEND_ATTR_ENUM](#)(DtAppActiveState, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppLauncher, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppCamoType, 0x3)

[DtEXTEND_ATTR_ENUM](#)(DtAppConcealed, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppTent, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppRamp, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppAfterBurner, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppLaunchFlash, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppLifeformState, 0xf)

[DtEXTEND_ATTR_ENUM](#)(DtAppWeapon1, 0x3)

[DtEXTEND_ATTR_ENUM](#)(DtAppWeapon2, 0x3)

[DtEXTEND_ATTR_ENUM](#)(DtAppDensity, 0xf)

[DtEXTEND_ATTR_ENUM](#)(DtAppMinefieldType, 0x3)

Minefield appearance

[DtEXTEND_ATTR_ENUM](#)(DtAppMinefieldActiveStatus, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppMinefieldLane, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppMinefieldState, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectPercentComplete, 0xff)

Point, Linear, and Areal Object appearance

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectDamage, 0x3)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectPredistributed, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectState, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectSmoking, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectFlaming, 0x1)

Point object specific appearance

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectAbatisBreach, 0x3)

This is for the log crib, abatis, vehicle defilade, and infantry fighting position breach appearance

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectBurstOpacity, 0xff)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectBurstSize, 0xff)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectBurstHeight, 0xff)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectBurstNumOfBursts, 0x3f)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectBurstChemicalSmoke, 0x3)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectCraterSize, 0xff)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectRibbonBridgeNumOfSegments, 0xff)

Linear object specific appearance

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectTankDitchBreach, 0x3)

This is for the tank ditch and concertina wire breach appearance

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectTankDitchBreachLength, 0xff)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectTankDitchBreachLocation, 0xff)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectExhaustSmokeOpacity, 0xff)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectExhaustSmokeAttached, 0x1)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectExhaustSmokeChemical, 0x3)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectMinefieldLaneMarkerVisibleSide, 0x3)

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectMinefieldBreach, 0x3)

Areal object specific appearance

[DtEXTEND_ATTR_ENUM](#)(DtAppObjectMinefieldMineCount, 0xffff)

#define

DtEXTEND_ATTR_VAL_ENUM(name, val, attr)

DtEXTEND_ATTR_VAL_ENUM(DtPaintSchemeUniformBitVal, 0, PaintScheme)

DtEXTEND_ATTR_VAL_ENUM(DtPaintSchemeCamouflageBitVal, 1, PaintScheme)

DtEXTEND_ATTR_VAL_ENUM(DtMobilityKillNoneBitVal, 0, MobilityKill)

DtEXTEND_ATTR_VAL_ENUM(DtMobilityKillDisabledBitVal, 1, MobilityKill)

DtEXTEND_ATTR_VAL_ENUM(DtFirePowerKillNoneBitVal, 0, FirePowerKill)

DtEXTEND_ATTR_VAL_ENUM(DtFirePowerKillDisabledBitVal, 1, FirePowerKill)

DtEXTEND_ATTR_VAL_ENUM(DtDamageNoneBitVal, 0, Damage)

DtEXTEND_ATTR_VAL_ENUM(DtDamageSlightBitVal, 1, Damage)

DtEXTEND_ATTR_VAL_ENUM(DtDamageModerateBitVal, 2, Damage)

DtEXTEND_ATTR_VAL_ENUM(DtDamageDestroyedBitVal, 3, Damage)

DtEXTEND_ATTR_VAL_ENUM(DtSmokeNoneBitVal, 0, Smoke)

DtEXTEND_ATTR_VAL_ENUM(DtSmokeRisingBitVal, 1, Smoke)

DtEXTEND_ATTR_VAL_ENUM(DtSmokeEngineBitVal, 2, Smoke)

DtEXTEND_ATTR_VAL_ENUM(DtSmokeEngineAndPlumeBitVal, 3, Smoke)

DtEXTEND_ATTR_VAL_ENUM(DtTrailingEffectsNoneBitVal, 0, TrailingEffects)

DtEXTEND_ATTR_VAL_ENUM(DtTrailingEffectsSmallBitVal, 1, TrailingEffects)

DtEXTEND_ATTR_VAL_ENUM(DtTrailingEffectsMediumBitVal, 2, TrailingEffects)

DtEXTEND_ATTR_VAL_ENUM(DtTrailingEffectsLargeBitVal, 3, TrailingEffects)

DtEXTEND_ATTR_VAL_ENUM(DtHatchNABitVal, 0, Hatch)

DtEXTEND_ATTR_VAL_ENUM(DtHatchClosedBitVal, 1, Hatch)

DtEXTEND_ATTR_VAL_ENUM(DtHatchPoppedBitVal, 2, Hatch)

DtEXTEND_ATTR_VAL_ENUM(DtHatchPoppedAndPersonBitVal, 3, Hatch)

DtEXTEND_ATTR_VAL_ENUM(DtHatchOpenBitVal, 4, Hatch)

DtEXTEND_ATTR_VAL_ENUM(DtHatchOpenAndPersonBitVal, 5, Hatch)

DtEXTEND_ATTR_VAL_ENUM(DtLightsNoneBitVal, 0, Lights)

DtEXTEND_ATTR_VAL_ENUM(DtLightsRunningBitVal, 1, Lights)

DtEXTEND_ATTR_VAL_ENUM(DtLightsNavigationBitVal, 2, Lights)

DtEXTEND_ATTR_VAL_ENUM(DtLightsFormationBitVal, 3, Lights)

DtEXTEND_ATTR_VAL_ENUM(DtFlamingNoBitVal, 0, Flaming)

DtEXTEND_ATTR_VAL_ENUM(DtFlamingYesBitVal, 1, Flaming)

DtEXTEND_ATTR_VAL_ENUM(DtNotFinalPduBitVal, 0, FinalPdu)

DtEXTEND_ATTR_VAL_ENUM(DtFinalPduBitVal, 1, FinalPdu)

DtEXTEND_ATTR_VAL_ENUM(DtNotFrozenBitVal, 0, Frozen)

DtEXTEND_ATTR_VAL_ENUM(DtFrozenBitVal, 1, Frozen)

DtEXTEND_ATTR_VAL_ENUM(DtPowerPlantOffBitVal, 0, PowerPlantStatus)

DtEXTEND_ATTR_VAL_ENUM(DtPowerPlantOnBitVal, 1, PowerPlantStatus)

DtEXTEND_ATTR_VAL_ENUM(DtActiveBitVal, 0, ActiveState)

DtEXTEND_ATTR_VAL_ENUM(DtDeactivatedBitVal, 1, ActiveState)

DtEXTEND_ATTR_VAL_ENUM(DtLauncherNotRaisedBitVal, 0, Launcher)

DtEXTEND_ATTR_VAL_ENUM(DtLauncherRaisedBitVal, 1, Launcher)

DtEXTEND_ATTR_VAL_ENUM(DtDesertCamouflageBitVal, 0, CamoType)

[DtEXTEND_ATTR_VAL_ENUM](#)(DtWinterCamouflageBitVal, 1, CamoType)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtForestCamouflageBitVal, 2, CamoType)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtNotConcealedBitVal, 0, Concealed)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtConcealedBitVal, 1, Concealed)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtTentNotExtendedBitVal, 0, Tent)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtTentExtendedBitVal, 1, Tent)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtRampUpBitVal, 0, Ramp)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtRampDownBitVal, 1, Ramp)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtAfterBurnerNotOnBitVal, 0, AfterBurner)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtAfterBurnerOnBitVal, 1, AfterBurner)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLaunchFlashNotPresentBitVal, 0, LaunchFlash)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLaunchFlashPresentBitVal, 1, LaunchFlash)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformNABitVal, 0, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformStandingBitVal, 1, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformWalkingBitVal, 2, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformRunningBitVal, 3, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformKneelingBitVal, 4, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformProneBitVal, 5, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformCrawlingBitVal, 6, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformSwimmingBitVal, 7, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformParachutingBitVal, 8, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformJumpingBitVal, 9, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformSittingBitVal, 10, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformSquattingBitVal, 11, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformCrouchingBitVal, 12, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtLifeformWadingBitVal, 13, LifeformState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtWeapon1None, 0, Weapon1)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtWeapon1Stowed, 1, Weapon1)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtWeapon1Deployed, 2, Weapon1)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtWeapon1InFirePosition, 3, Weapon1)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtWeapon2None, 0, Weapon2)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtWeapon2Stowed, 1, Weapon2)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtWeapon2Deployed, 2, Weapon2)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtWeapon2InFirePosition, 3, Weapon2)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtDensityClearBitVal, 0, Density)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtDensityHazyBitVal, 1, Density)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtDensityDenseBitVal, 2, Density)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtDensityVeryDenseBitVal, 3, Density)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtDensityOpaqueBitVal, 4, Density)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtMixedMinefield, 0, MinefieldType)
Minefield appearance
[DtEXTEND_ATTR_VAL_ENUM](#)(DtPureAntiPersonnelMinefield, 1, MinefieldType)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtPureAntiTankMinefield, 2, MinefieldType)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtActiveMinefield, 0, MinefieldActiveStatus)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtInactiveMinefield, 1, MinefieldActiveStatus)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtMinefieldHasActiveLane, 0, MinefieldLane)

[DtEXTEND_ATTR_VAL_ENUM](#)(DtMinefieldHasInactiveLane, 1, MinefieldLane)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtActiveMinefieldState, 0, MinefieldState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtDeactivatedMinefieldState, 1, MinefieldState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectNoDamage, 0, ObjectDamage)
Point, Linear, and Areal Object appearance
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectDamaged, 1, ObjectDamage)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectDestroyed, 2, ObjectDamage)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectCreatedDuringExercise, 0, ObjectPredistributed)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectPredistributed, 1, ObjectPredistributed)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectActive, 0, ObjectState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectDeactivated, 1, ObjectState)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectNotSmoking, 0, ObjectSmoking)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectSmoking, 1, ObjectSmoking)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectNotFlaming, 0, ObjectFlaming)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectFlaming, 1, ObjectFlaming)

Point object specific appearance

[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectAbatisNoBreaching, 0, ObjectAbatisBreach)
This is for the log crib, abatis, vehicle defilade, and infantry fighting position breach appearance
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectAbatisBreached, 1, ObjectAbatisBreach)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectAbatisCleared, 2, ObjectAbatisBreach)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectBurstOtherSmoke, 0, ObjectBurstChemicalSmoke)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectBurstHydrochloricSmoke, 1, ObjectBurstChemicalSmoke)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectBurstWhitePhosphorousSmoke, 2, ObjectBurstChemicalSmoke)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectBurstRedPhosphorousSmoke, 3, ObjectBurstChemicalSmoke)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectTankDitchBreachNoBreaching, 0, ObjectTankDitchBreach)
Linear object specific appearance
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectTankDitchBreachSlightBreaching, 1, ObjectTankDitchBreach)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectTankDitchBreachModerateBreaching, 2, ObjectTankDitchBreach)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectTankDitchBreachCleared, 3, ObjectTankDitchBreach)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectExhaustSmokeNotAttached, 0, ObjectExhaustSmokeAttached)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectExhaustSmokeIsAttached, 1, ObjectExhaustSmokeAttached)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectExhaustOtherSmoke, 0, ObjectExhaustSmokeChemical)
[DtEXTEND_ATTR_VAL_ENUM](#)(DtObjectExhaustHydrochloricSmoke, 1, ObjectExhaustSmokeChemical)

	<u>DtEXTEND_ATTR_VAL_ENUM</u> (DtObjectExhaustWhitePhosphorousSmoke, 2, ObjectExhaustSmokeChemical)
	<u>DtEXTEND_ATTR_VAL_ENUM</u> (DtObjectExhaustRedPhosphorousSmoke, 3, ObjectExhaustSmokeChemical)
	<u>DtEXTEND_ATTR_VAL_ENUM</u> (DtObjectMinefieldLeftSideVisible, 0, ObjectMinefieldLaneMarkerVisibleSide)
	<u>DtEXTEND_ATTR_VAL_ENUM</u> (DtObjectMinefieldRightSideVisible, 1, ObjectMinefieldLaneMarkerVisibleSide)
	<u>DtEXTEND_ATTR_VAL_ENUM</u> (DtObjectMinefieldBothSidesVisible, 2, ObjectMinefieldLaneMarkerVisibleSide)
	<u>DtEXTEND_ATTR_VAL_ENUM</u> (DtObjectMinefieldNoBreaching, 0, ObjectMinefieldBreach)
	<u>DtEXTEND_ATTR_VAL_ENUM</u> (DtObjectMinefieldBreached, 1, ObjectMinefieldBreach)
	<u>DtEXTEND_ATTR_VAL_ENUM</u> (DtObjectMinefieldCleared, 2, ObjectMinefieldBreach)
#define	DtDisPduFactory_H_
#define	DtReflectedEmitterSystemList_H_
#define	DtElectromagneticEmissionPdu
#define	DtReflectedAggregateList_H_
#define	DtReflectedAggregate_H_
typedef void	<u>(*DtAggregateCallbackFunction)</u> (DtReflectedAggregate* obj, void* userData)
#define	DtReflectedEntity_H_
typedef void	<u>(*DtEntityCallbackFunction)</u> (DtReflectedEntity* obj, void* userData)
#define	<u>DtReflectedObject_H_</u>
#define	DtReflectedEmitterSystem_H_
typedef void	<u>(*DtEmitterSystemCallbackFunction)</u> (DtReflectedEmitterSystem* obj, void* userData)
#define	DtReflectedEntityList_H_
#define	DtReflectedObjectList_H_
#define	DtDiscoverObjectCbInfo_H_
#define	dDtStateMsg_H_
#define	DtElectromagneticEmissionPdu_H_
#define	DtElectromagneticEmissionStaticSize
#define	DtEMISSION_SUCCESS
#define	DtBAD_SYS_NUM
#define	DtBAD_BEAM_NUM
#define	DtBAD_TARGET_NUM
#define	DtBAD_SYS_ADDR
#define	DtBAD_BEAM_ADDR
#define	DtBAD_TARGET_ADDR
#define	DtINCONSISTENT_SYS_LENGTHS_PDU_LENGTH
#define	DtINCONSISTENT_BEAM_LENGTHS_SYS_LENGTH
#define	DtINCONSISTENT_NUM_TARGETS_BEAM_LENGTH

typedef void

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

typedef void

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

typedef void

#define

#define

#define

#define

[\(*DtElectromagneticEmissionPduCb\)](#)(DtElectromagneticEmissionPdu *pdu,
void *usr)

DtEmissionPdu_H_

DtEmissionStaticSize

DtEMISSION_SUCCESS

DtBAD_SYS_NUM

DtBAD_BEAM_NUM

DtBAD_TARGET_NUM

DtBAD_SYS_ADDR

DtBAD_BEAM_ADDR

DtBAD_TARGET_ADDR

DtINCONSISTENT_SYS_LENGTHS_PDU_LENGTH

DtINCONSISTENT_BEAM_LENGTHS_SYS_LENGTH

DtINCONSISTENT_NUM_TARGETS_BEAM_LENGTH

[\(*DtEmissionPduCb\)](#)(DtEmissionPdu *pdu, void *usr)

DtEmitterBeamRepository_H_

DtEmptyFomMapper_H_

DtStateEncoderFactory_H_

ENTITY_PUB_H_

DtEntityStateRepository_H_

This file provides protocol independence for Local Entities

[ENTITYTYPES_H](#)

DtSameEntityKind(a,b)

DtlsLandPlatform(a)

DtlsLandLifeForm(a)

DtlsAirPlatform(a)

DtlsTank(a)

DtlsRotaryWing(a)

DtlsFixedWing(a)

antiPersonell

DtEntityIsAntiPersonnel(a)

DtEnvironmentEntityDecoder_H_

DtEnvironmentEntityEncoder_H_

DtEnvironmentTypeRecord_H_

DtEnvironmentalProcessPdu_H_

[\(*DtEnvironmentalProcessPduCb\)](#)(DtEnvironmentalProcessPdu* pdu, void*
usr)

DtEnvironmentalProcessPdu callback function signature

DtEntityStatePdu_H_

DtEntityStateStaticSize

DtINCREMENT

DtNO_SUCH_ART_PART

```
#define
typedef void
#define
typedef void

#define
typedef DtEventId
#define
#define
#define
typedef DtEventReportPdu
typedef void
#define
typedef void

#define
#define
#define
#define
typedef DtFirePdu
typedef void
#define
#define
#define
#define
#define
#define
#define
#define
#define
#define
#define
typedef DtEntityIdentifier
typedef DtObjectId
#define
#define
#define
```

START_PART_SUCCESS

[**\(*DtEntityStatePduCb\)**](#)(DtEntityStatePdu *pdu, void *usr)

DtEntityStateUpdatePdu_H_

[**\(*DtEntityStateUpdatePduCb\)**](#)(DtEntityStateUpdatePdu* pdu, void* usr)
DtEntityStateUpdatePdu callback function signature

DtEventId_H_

[**DtEventID**](#)

EVENTREPORT_H_

DtEventReportPdu_H_
This file provides protocol independence for EventReport Interactions

DtEventReportStaticSize

[**DtEventReportInteraction**](#)

[**\(*DtEventReportPduCb\)**](#)(DtEventReportPdu *pdu, void *usr)

DtEventReptRPdu_H_

[**\(*DtEventReptRPduCb\)**](#)(DtEventReptRPdu* pdu, void* usr)
DtEventReptRPdu callback function signature

[**EXERCISE_CONNECTION_H_**](#)

FIRE_INTERACTION_H_
This file provides protocol independence for exercise connections

DtFirePdu_H_
This file provides protocol independence for Fire Interactions

DtFireStaticSize

[**DtFireInteraction**](#)

[**\(*DtFireInteractionCb\)**](#)(DtFireInteraction *inter, void *usr)

DtFom_H_

DtFomMapper_H_

DtBasicFixedWingAircraftRecord_H_

DtBasicRotorWingAircraftRecord_H_

DtBasicGroundCombatSoldierRecord_H_

DtBasicGroundCombatVehicleRecord_H_

DtEnhancedFixedWingAircraftRecord_H_

DtEnhancedRotorWingAircraftRecord_H_

DtEnhancedGroundCombatSoldierRecord_H_

DtEnhancedGroundCombatVehicleRecord_H_

DtGroundLogisticsVehicleRecord_H_

DtGenericObjectPublisher_H_

DtGenericStateRepository_H_

DtGlobalObjectDesignator_H_

[**DtObjectId**](#)

[**DtGlobalObjectDesignator**](#)

DtGridAxisRecordFixed_H_

DtGridAxisRecordVariable_H_

DtGriddedDataPdu_H_

typedef void

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

[\(*DtGriddedDataPduCb\)](#)(DtGriddedDataPdu* pdu, void* usr)

DtGriddedDataPdu callback function signature

DtGridDataRecord_H_

DtGridDataRecordType0_H_

DtGridDataRecordType1_H_

DtGridDataRecordType2_H_

DtAcknowledgeDecoder_H_

DtAcknowledgeEncoder_H_

DtAcknowledgeInteraction_H_

DtActionRequestDecoder_H_

DtActionRequestEncoder_H_

DtActionRequestInteraction_H_

DtActionRequestToObjDecoder_H_

DtActionRequestToObjEncoder_H_

DtActionRequestToObjInter_H_

DtActionResponseDecoder_H_

DtActionResponseEncoder_H_

DtActionResponseInteraction_H_

DtActionResponseFromObjDecoder_H_

DtActionResponseFromObjEncoder_H_

DtActionResponseFromObjInter_H_

DtAggregatePublisher_H_

DtAppSpecificRadioSignalDecoder_H_

DtAppSpecificRadioSignalEncoder_H_

DtAttributeChangeReqDec_H_

DtAttributeChangeReqEnc_H_

DtAttributeChangeReqInter_H_

DtAttributeChangeResDec_H_

DtAttributeChangeResEnc_H_

DtAttributeChangeResInter_H_

DtCollisionDecoder_H_

DtCollisionEncoder_H_

DtCollisionInteraction_H_

DtCommentDecoder_H_

DtCommentEncoder_H_

DtCommentInteraction_H_

DtCreateEntityDecoder_H_

DtCreateEntityEncoder_H_

DtCreateEntityInteraction_H_

DtCreateObjReqDecoder_H_

DtCreateObjReqEncoder_H_

#define	DtCreateObjReqInter_H_
#define	DtCreateObjectResultDecoder_H_
#define	DtCreateObjectResultEncoder_H_
#define	DtCreateObjectResultInter_H_
#define	DtDataDecoder_H_
#define	DtDataEncoder_H_
#define	DtDataInteraction_H_
#define	DtDataQueryDecoder_H_
#define	DtDataQueryEncoder_H_
#define	DtDataQueryInteraction_H_
#define	DtDatabaseIndexRadioSignalDecoder_H_
#define	DtDatabaseIndexRadioSignalEncoder_H_
#define	DtDesignatorPublisher_H_
#define	DtDetonationDecoder_H_
#define	DtDetonationEncoder_H_
#define	DtDetonationInteraction_H_
#define	DtEncodedAudioRadioSignalDecoder_H_
#define	DtEncodedAudioRadioSignalEncoder_H_
#define	DtEntityPublisher_H_
#define	DtEventReportDecoder_H_
#define	DtEventReportEncoder_H_
#define	DtEventReportInteraction_H_
#define	DtExerciseConn_H_
#define	DtFireDecoder_H_
#define	DtFireEncoder_H_
#define	DtFireInteraction_H_
#define	DtHInteraction_H_
#define	DtInteractionFactory_H_
#define	DtInteractionWithData_H_
#define	DtInteractionWithEncDec_H_
#define	DtHlaObject_H_
#define	DtHlaObjectFactory_H_
#define	DtHlaObjectWithStateRep_H_
#define	DtLgrControlDecoder_H_
#define	DtLgrControlEncoder_H_
#define	DtLgrControlInteraction_H_
#define	_HOSTSTRUCTS_H_
typedef DtU16	DtExerciseld
DtU16	DtHost
	DtSimulationAddress

[illegible]

DtDeferedEntityId

DtDeferedEntityId

DtObjectId

DtAny

DtRawBinaryRadioSignalDecoder_H_

DtRawBinaryRadioSignalEncoder_H_

DtReflectedEmitterBeamList_H_

DtReflectedDesignatorList_H_

DtReflectedEmitterSystemList_H_

DtReflectedAggregateList_H_

DtReflectedDesignator_H_

DtReflectedAggregate_H_

DtReflectedEntity_H_

DtReflectedObject_H_

DtReflectedEmitterSystem_H_

DtReflectedEntityList_H_

DtReflectedRadioTransmitter_H_

DtReflectedObjectList_H_

DtReflectedRadioReceiverList_H_

DtReflectedRadioTransmitterList_H_

DtRemoveObjectRequestDecoder_H_

DtRemoveObjectRequestEncoder_H_

DtRemoveObjectRequestInter_H_

DtRemoveObjectResultDecoder_H_

DtRemoveObjectResultEncoder_H_

DtRemoveObjectResultInter_H_

DtRemoveEntityDecoder_H_

DtRemoveEntityEncoder_H_

DtRemoveEntityInteraction_H_

DtRepairCompleteDecoder_H_

DtRepairCompleteEncoder_H

DtRepairCompleteInteraction_H

DtRepairResponseDecoder_H

DtRepairResponseEncoder H

DtRepairResponseInteraction H

DtResupplyCancelDecoder H

DtResupplyCancelEncoder H

DtResupplyCancelInteraction H

DtResupplyOfferDecoder H

#define	DtResupplyOfferEncoder_H_
#define	DtResupplyOfferInteraction_H_
#define	DtResupplyReceivedDecoder_H_
#define	DtResupplyReceivedEncoder_H_
#define	DtResupplyRecInteraction_H_
#define	DtServiceRequestDecoder_H_
#define	DtServiceRequestEncoder_H_
#define	DtServiceRequestInteraction_H_
#define	DtSetDataDecoder_H_
#define	DtSetDataEncoder_H_
#define	DtSetDataInteraction_H_
#define	DtSignalInteraction_H_
#define	DtStartResumeDecoder_H_
#define	DtStartResumeEncoder_H_
#define	DtStartResumeInteraction_H_
#define	DtHlaStateDecoder_H_
#define	DtHlaStateEncoder_H_
#define	hDtStateMsg_H_
#define	DtStopFreezeDecoder_H_
#define	DtStopFreezeEncoder_H_
#define	DtStopFreezeInteraction_H_
#define	DtUnknownInteraction_H_
#define	DtViewControlDecoder_H_
#define	DtViewControlEncoder_H_
#define	DtViewControlInteraction_H_
#define	DtIffAtcNav aidsBeamData_H_
#define	DtIffAtcNav aidsLayerHeader_H_
#define	DtIffAtcNav aidsOperational_H_
#define	DtIffAtcNav aidsSecondOperational_H_
#define	DtIffAtcNav aidsParameter_H_
#define	DtIffAtcNav aidsPdu_H_
typedef void	(*DtIffAtcNav aidsPduCb) (DtIffAtcNav aidsPdu* pdu, void* usr) <i>DtIffAtcNav aidsPdu callback function signature</i>
#define	DtIffAtcNav aidsSystemID_H_
#define	DtInterClassDesc_H_
#define	DtInteractionDecoderFactory_H_
#define	DtInteractionEncoderFactory_H_
#define	DtIntercomControlPdu_H_
typedef void	(*DtIntercomControlPduCb) (DtIntercomControlPdu* pdu, void* usr) <i>DtIntercomControlPdu callback function signature</i>
#define	DtIntercomEntityDestRec_H_

#define	DtIntercomGroupAsgmtRec_H_
#define	DtIntercomGroupDestRec_H_
#define	DtIntercomParmRec_H_
#define	DtIntercomSignalPdu_H_
typedef void	(*DtIntercomSignalPduCb) (DtIntercomSignalPdu* pdu, void* usr) <i>DtIntercomSignalPdu callback function signature</i>
#define	DtInteractionDecoder_H_
#define	DtInteractionEncoder_H_
#define	DtInteractionManager_H_
#define	DtIsGroupOfPdu_H_
typedef void	(*DtIsGroupOfPduCb) (DtIsGroupOfPdu* pdu, void* usr) <i>DtIsGroupOfPdu callback function signature</i>
#define	DtIsPartOfPdu_H_
typedef void	(*DtIsPartOfPduCb) (DtIsPartOfPdu* pdu, void* usr) <i>DtIsPartOfPdu callback function signature</i>
#define	DtLaserPdu_H_
#define	DtLaserStaticSize
typedef void	(*DtLaserPduCb) (DtLaserPdu *pdu, void *usr)
#define	LGR_CONTROL_INTERACTION_H_
#define	DtLgrControlPdu_H_ <i>This file provides protocol independence for LgrControl Interactions</i>
typedef DtLgrControlPdu	DtLgrControlInteraction
typedef void	(*DtLgrControlInteractionCb) (DtLgrControlInteraction* inter, void* usr)
#define	DtLeAppearancePdu_H_
typedef void	(*DtLeAppearancePduCb) (DtLeAppearancePdu* pdu, void* usr) <i>DtLeAppearancePdu callback function signature</i>
#define	DtLeArticulatedPartsPdu_H_
typedef void	(*DtLeArticulatedPartsPduCb) (DtLeArticulatedPartsPdu* pdu, void* usr) <i>DtLeArticulatedPartsPdu callback function signature</i>
#define	DtLeDeadReckoningParameters_H_
#define	DtLeDetonationPdu_H_
 class DtLeDetonationPdu:	 (*DtLeDetonationPduCb) (DtLeDetonationPdu* pdu, void* usr) <i>DtLeDetonationPdu callback function signature</i>
typedef void	DtLeEntityID_H_
#define	DtLeEntityOrientation_H_
#define	DtLeEntityLinearVelocity_H_
#define	DtLeFirePdu_H_
 class DtLeFirePdu:	 (*DtLeFirePduCb) (DtLeFirePdu* pdu, void* usr) <i>DtLeFirePdu callback function signature</i>
typedef void	DtLeEntityLocationEntityCoordinates_H_
#define	DtLeEntityLocationWorldCoordinates_H_
#define	DtLeOrientationError_H_

```

#define
#define
typedef void
#define
typedef void
#define
#define
#define
#define
typedef void
#define
#define
#define
#define
#define
typedef void
#define
#define
typedef void
#define
typedef void
#define
typedef void
#define
#define
#define
#define
For backwards compatibility
typedef DtNetU8
typedef DtNetU8
typedef DtNetU8
typedef DtNetU8
typedef DtNetEntityIdentifier
typedef DtNetEntityIdentifier
typedef DtVector
typedef struct DtNetRadioEntityType DtNetU8
DtNetU8

```

```

DtLePositionError_H_
DtLeTspiPdu_H_
(\*DtLeTspiPduCb)(DtLeTspiPdu* pdu, void* usr)
DtLeTspiPdu callback function signature
DtLinearObjectStatePdu_H_
(\*DtLinearObjectStatePduCb)( DtLinearObjectStatePdu* pdu, void* usr)
DtLinearObjectStatePdu callback function signature
DtLinearSegmentParameter_H_
DtLocalHlaObjectManager_H_
DtMessagePdu_H_
DtMessageStaticSize
(\*DtMessagePduCb)(DtMessagePdu *pdu, void *usr)
DtMilitaryEntityDecoder_H_
DtMilitaryEntityEncoder_H_
DtMilitaryPlatformEntityDecoder_H_
DtMilitaryPlatformEntityEncoder_H_
DtMinefieldDataPdu_H_
(\*DtMinefieldDataPduCb)(DtMinefieldDataPdu* pdu, void* usr)
DtMinefieldDataPdu callback function signature
DtMinefieldPerimeterRecord_H_
DtMinefieldQueryPdu_H_
(\*DtMinefieldQueryPduCb)(DtMinefieldQueryPdu* pdu, void* usr)
DtMinefieldQueryPdu callback function signature
DtMinefieldResponseNackPdu_H_
(\*DtMinefieldResponseNackPduCb)(DtMinefieldResponseNackPdu* pdu,
void* usr)
DtMinefieldResponseNackPdu callback function signature
DtMinefieldStatePdu_H_
(\*DtMinefieldStatePduCb)(DtMinefieldStatePdu* pdu, void* usr)
DtMinefieldStatePdu callback function signature
DtMunitionEntityDecoder_H_
DtMunitionEntityEncoder_H_
DtNetBoolean_H_
_NETSTRUCTS_H_

DtNet8Bits
DtNet16Bits[2]
DtNet32Bits[4]
DtNet64Bits[8]
DtNetObjectId
DtNetGlobalObjectDesignator
DtNet64Vector
kind
domain

```

DtNetU16	country
DtNetU8	category
DtNetU8	nomenclatureVersion
DtNetU16	nomenclature
	DtNetRadioEntityType
typedef struct DtNetEntityType DtNetU8	DtEntityKind
DtNetU8	DtDomain
DtNetU16	DtCountry
DtNetU8	DtCategory
DtNetU8	DtSubCategory
DtNetU8	DtSpecific
DtNetU8	DtExtra
	DtNetEntityType
typedef struct DtNetBurstDescriptor DtNetEntityType	DtMunition
DtNetU16	DtWarhead
DtNetU16	DtFuze
DtNetU16	DtQuantity
DtNetU16	DtRate
	DtNetBurstDescriptor
typedef struct DtNetSimulationAddress DtNetU16	DtSite
DtNetU16	DtHost
	DtNetSimulationAddress
typedef struct DtNetEntityIdentifier DtNetSimulationAddress	DtSimulator
DtNetU16	DtEntity
	DtNetEntityIdentifier
typedef struct DtNetEventId DtNetSimulationAddress	DtSimulator
DtNetU16	DtEvent
	DtNetEventId
typedef struct DtNetEulerAngles DtNetFloat32	psi
DtNetFloat32	theta
DtNetFloat32	phi
	DtNetEulerAngles
typedef DtNetFloat32	DtNet32Vector[3]
typedef DtNetFloat64	DtNet64Vector[3]
typedef struct DtNetArtParamRecord DtNetU8	artOrAttached
DtNetU8	DtChange
DtNetU16	DtChange
DtNetU16	DtAttachedTo
DtNetU32	DtType
DtNet64Bits	DtValue
	DtNetArtParamRecord

 typedef DtNetArtParamRecord	DtArticulatedParts
 typedef DtNetArtParamRecord	DtNetArtPartRecord
 typedef struct DtNetAttachedPart DtNetU8	myArtOrAttached
 DtNetU8	myChange
 DtNetU16	myAttachedTo
 DtNetU32	myStation
 DtNetEntityType	myType
	DtNetAttachedPart
 typedef struct DtNetAngularVelocity DtNetFloat32	wx
 DtNetFloat32	wy
 DtNetFloat32	wz
	DtNetAngularVelocity
 typedef struct DtNetLinearVelocity DtNetFloat32	DtX
 DtNetFloat32	DtY
 DtNetFloat32	DtZ
	DtNetLinearVelocity
 typedef struct DtNetWorldCoordinates DtNetFloat64	DtX
 DtNetFloat64	DtY
 DtNetFloat64	DtZ
	DtNetWorldCoordinates
 typedef struct DtNetEntityCoordinates DtNetFloat32	DtX
 DtNetFloat32	DtY
 DtNetFloat32	DtZ
	DtNetEntityCoordinates
 typedef struct DtNetTopoUpCoord DtNetFloat32	East
 DtNetFloat32	North
 DtNetFloat32	Up
	DtNetTopoUpCoord
 typedef struct DtNetInt32	hour
 DtNetU32	hourFrac
	DtNetClockTime
 DtTime	DtClockTime2AbsTime (const DtNetClockTime *clock)
 void	DtAbsTime2ClockTime (DtTime absTime, DtNetClockTime *clock)
 #define	DtObjClassDesc_H_
 #define	DtObjectType_H_
 #define	DtObjectReflectAttributeValuesCbInfo_H_
 #define	_parseEnum_h_
 char*	DtEnumString (DtDetonatorFuze param)
 char*	DtEnumString (DtWarheadType param)
 char*	DtEnumString (DtRepairType param)
 char*	DtEnumString (DtCollisionType param)

char*	<u>DtEnumString</u> (DtForceType param)
char*	<u>DtEnumString</u> (DtDetonationResult param)
char*	<u>DtEnumString</u> (DtProtocolFamily param)
char*	<u>DtEnumString</u> (DtRepairResult param)
char*	<u>DtEnumString</u> (DtServiceType param)
char*	<u>DtEnumString</u> (DtStopReason param)
char*	<u>DtEnumString</u> (DtAcknowledgeFlag param)
char*	<u>DtEnumString</u> (DtResponseFlag param)
char*	<u>DtEnumString</u> (DtActionId param)
char*	<u>DtEnumString</u> (DtRequestStatus param)
char*	<u>DtEnumString</u> (DtEventType param)
char*	<u>DtEnumString</u> (DtDatumParam param)
char*	<u>DtEnumString</u> (DtEmissionPduType param)
char*	<u>DtEnumString</u> (DtEmitterSysName param)
char*	<u>DtEnumString</u> (DtEmissionFunc param)
char*	<u>DtEnumString</u> (DtBeamFunc param)
char*	<u>DtEnumString</u> (DtTrackingJamming param)
char*	<u>DtEnumString</u> (DtReferenceSys param)
char*	<u>DtEnumString</u> (DtDesignatorCodeName param)
char*	<u>DtEnumString</u> (DtDesignatorCode param)
char*	<u>DtEnumString</u> (DtRadioTransmitState param)
char*	<u>DtEnumString</u> (DtRadioInputSource param)
char*	<u>DtEnumString</u> (DtAntennaPatternType param)
char*	<u>DtEnumString</u> (DtRadioSpreadSpectrum param)
char*	<u>DtEnumString</u> (DtRadioMajorModulation param)
char*	<u>DtEnumString</u> (DtRadioDetailedModulationAmplitude param)
char*	<u>DtEnumString</u> (DtDetailedModulationAmplitudeAngle param)
char*	<u>DtEnumString</u> (DtDetailedModulationAngle param)
char*	<u>DtEnumString</u> (DtDetailedModulationCombination param)
char*	<u>DtEnumString</u> (DtDetailedModulationUnmodulated param)
char*	<u>DtEnumString</u> (DtDetailedModulationType param)
char*	<u>DtEnumString</u> (DtRadioSystem param)
char*	<u>DtEnumString</u> (DtCryptoMode param)
char*	<u>DtEnumString</u> (DtRadioCryptoSystem param)
char*	<u>DtEnumString</u> (DtEncodingClass param)
char*	<u>DtEnumString</u> (DtRadioReceiverState param)
char*	<u>DtEnumString</u> (DtTacticalDataLinkType param)
char*	<u>DtEnumString</u> (DtUserProtocol param)
char*	<u>DtEnumString</u> (DtEncodingType param)
char*	<u>DtEnumString</u> (DtCreateObjectResult param)

char*	<u>DtEnumString</u> (DtRemoveObjectResult param)
char*	<u>DtEnumString</u> (DtAttributeChangeResult param)
char*	<u>DtEnumString</u> (DtDeadReckonTypes param)
char*	<u>DtEnumString</u> (DtViewerCommand param)
char*	<u>DtEnumString</u> (DtLgrCommand param)
char*	<u>DtEnumString</u> (DtAppearAttributeBitShift param)
char*	<u>DtEnumString</u> (DtAppearAttribute param)
char*	<u>DtEnumString</u> (DtAppearAttrVal param)
char*	<u>DtEnumString</u> (DtDamageState param)
char*	<u>DtEnumString</u> (DtTrailState param)
char*	<u>DtEnumString</u> (DtHatchState param)
char*	<u>DtEnumString</u> (DtLightState param)
char*	<u>DtEnumString</u> (DtLifeformState param)
char*	<u>DtEnumString</u> (DtCamouflageType param)
char*	<u>DtEnumString</u> (DtWeaponState param)
char*	<u>DtEnumString</u> (DtCapability param)
char*	<u>DtEnumString</u> (DtDensityType param)
char*	<u>DtEnumString</u> (DtArtParamType param)
char*	<u>DtEnumString</u> (DtArtPartType param)
char*	<u>DtEnumString</u> (DtIntercomCtlType param)
char*	<u>DtEnumString</u> (DtIntercomCommClass param)
char*	<u>DtEnumString</u> (DtIntercomCommType param)
char*	<u>DtEnumString</u> (DtIntercomCmdType param)
char*	<u>DtEnumString</u> (DtIntercomTransLineState param)
char*	<u>DtEnumString</u> (DtIntercomDestLineStateCmd param)
char*	<u>DtEnumString</u> (DtIntercomCtlRecType param)
char*	<u>DtEnumString</u> (DtEnvironmentalModelType param)
char*	<u>DtEnumString</u> (DtEnvironmentalStatusBit param)
char*	<u>DtEnumString</u> (DtGridDataRecType param)
char*	<u>DtEnumString</u> (DtGridAxisType param)
char*	<u>DtEnumString</u> (DtIffAtcNav aidsSysName param)
char*	<u>DtEnumString</u> (DtIffAtcNav aidsSysType param)
char*	<u>DtEnumString</u> (DtIffAtcNav aidsStatusBit param)
char*	<u>DtEnumString</u> (DtIffAtcNav aidsSysMode param)
char*	<u>DtEnumString</u> (DtIffAtcNav aidsChangeOptionsBit param)
char*	<u>DtEnumString</u> (DtIffAtcNav aidsParameterBit param)
char*	<u>DtEnumString</u> (DtIffAtcNav aidsAltParm4 param)
char*	<u>DtEnumString</u> (DtIffAtcNav aidsModifier param)
char*	<u>DtEnumString</u> (DtIffAtcNav aidsLayerSpecificInfo param)
char*	<u>DtEnumString</u> (DtIffAtcNav aidsParm1 param)

char*	<u>DtEnumString</u> (DtIffAtcNav aidsParm2 param)
char*	<u>DtEnumString</u> (DtIffAtcNav aidsApplicableModes param)
char*	<u>DtEnumString</u> (DtUnderwaterAcousticUpdate param)
char*	<u>DtEnumString</u> (DtUnderwaterAcousticSystemName param)
char*	<u>DtEnumString</u> (DtUnderwaterAcousticFunction param)
char*	<u>DtEnumString</u> (DtUnderwaterAcousticActiveEmissionParmIndex param)
char*	<u>DtEnumString</u> (DtUnderwaterAcousticScanPattern param)
char*	<u>DtEnumString</u> (DtUnderwaterAcousticPassiveParmIndex param)
char*	<u>DtEnumString</u> (DtUnderwaterAcousticPropulsionPlantConfig param)
char*	<u>DtEnumString</u> (DtUnderwaterAcousticHullMountedMaskerStatus param)
char*	<u>DtEnumString</u> (DtUnderwaterAcousticAdditionalPassiveActivityParmIndex param)
char*	<u>DtEnumString</u> (DtTransferType param)
char*	<u>DtEnumString</u> (DtRequiredReliabilityService param)
char*	<u>DtEnumString</u> (DtNature param)
char*	<u>DtEnumString</u> (DtPosition param)
char*	<u>DtEnumString</u> (DtStationName param)
char*	<u>DtEnumString</u> (DtObjectKind param)
char*	<u>DtEnumString</u> (DtModificationField param)
char*	<u>DtEnumString</u> (DtMinefieldDataFilterField param)
char*	<u>DtEnumString</u> (DtMinefieldFusing param)
char*	<u>DtEnumString</u> (DtMinefieldAntiHandlingDevice param)
char*	<u>DtEnumString</u> (DtSensorTypeCategory param)
char*	<u>DtEnumString</u> (DtOpticalSensorTypeSubcategory param)
char*	<u>DtEnumString</u> (DtFLIR_SensorTypeSubcategory param)
char*	<u>DtEnumString</u> (DtRADAR_SensorTypeSubcategory param)
char*	<u>DtEnumString</u> (DtMagneticSensorTypeSubcategory param)
char*	<u>DtEnumString</u> (DtLaserSensorTypeSubcategory param)
char*	<u>DtEnumString</u> (DtSONARSensorTypeSubcategory param)
char*	<u>DtEnumString</u> (DtPhysicalSensorTypeSubcategory param)
char*	<u>DtEnumString</u> (DtMultispectralSensorTypeSubcategory param)
char*	<u>DtEnumString</u> (DtMinefieldAlgae param)
char*	<u>DtEnumString</u> (DtMinefieldPaintScheme param)
char*	<u>DtEnumString</u> (DtMinefieldProtocol param)
char*	<u>DtEnumString</u> (DtSubProtocol param)
char*	<u>DtEnumString</u> (DtCharacterSet param)
char*	<u>DtEnumString</u> (DtLE_TSPI_FieldFlag param)
char*	<u>DtEnumString</u> (DtLE_AppearanceField1Flag param)
char*	<u>DtEnumString</u> (DtLE_AppearanceField2Flag param)
char*	<u>DtEnumString</u> (DtLE_FireFieldFlag param)
char*	<u>DtEnumString</u> (DtLE_DetonFieldFlag param)

```

char*
char*
char*
char*
char*
#define
#define
#define
extern int
extern int
extern int
#define
#define
#define
typedef void
#define
#define
typedef DtPdu*

```

class DtPduFactory: Instances of DtPduFactory are used to create PDUs (protocol data units) or packets, typically for DIS (as opposed to HLA)

```

#define
#define
#define
#define
#define
typedef void
#define
#define
#define
typedef DtDataQueryPdu
typedef void
#define
#define
typedef void
#define
#define
typedef void
#define

```

```

DtEnumString(DtLE_DetonField2Flag param)
DtEnumString(DtGroupedEntityCategory param)
DtEnumString(DtRestStatus param)
DtEnumString(DtAggregateState param)
DtEnumString(DtAggregateFormation param)
DtPdu_H_
DtSTATUS_OK
DtSTATUS_NOT_OK
DtProtocolVersionToSend
DtProtocolVersionToRecvMin
DtProtocolVersionToRecvMax
netPdu()
DtPduCallbackInfo_H_
DtPduCallbackManager_H_
(*DtPduCallbackFcn)(DtPdu* pdu, void* usr)
PDU callback function signature
DtPduDatumRecordSet_H_
DtPduFactory_H_
(*DtPduCreator)(const DtNetPduHeader* packet, DtBufferPtr buffer)
Pdu creation function signature

PDUS_H_
DtDataStaticSize
DtPhysicalEntityDecoder_H_
DtPhysicalEntityEncoder_H_
DtPointObjectStatePdu_H_
(*DtPointObjectStatePduCb)(DtPointObjectStatePdu* pdu, void* usr)
DtPointObjectStatePdu callback function signature
DtDataQueryPdu_H_
DtQueryStaticSize
DtQueryStaticSize
DtDataQueryInteraction
(*DtDataQueryPduCb)(DtDataQueryPdu *pdu, void *usr)
DtReceiverPdu_H_
DtReceiverStaticSize
(*DtReceiverPduCb)(DtReceiverPdu *pdu, void *usr)
DtReceiveInteractionCbInfo_H_
DtRecordPdu_H_
(*DtRecordPduCb)(DtRecordPdu* pdu, void* usr)
DtRecordPdu callback function signature
DtRecordSet_H_

```

	#define	DtRecordSetPdu_H_
	#define	DtRecordQueryPdu_H_
	typedef void	(*DtRecordQueryPduCb) (DtRecordQueryPdu* pdu, void* usr) <i>DtRecordQueryPdu callback function signature</i>
	#define	DtReflectAttributeValuesCbInfo_H_
	#define	DtReflectAttributeValuesCbList_H_
	#define	REFLECTED_EMITTER_LIST_H_
	#define	DtReflectedGenericObject_H_ <i>This file provides protocol independence for reflected emitter system lists</i>
	#define	DtReflectedGenericObjectList_H_
	#define	REMOTE_AGGREGATE_LIST_H_
	#define	REMOTE_AGGREGATE_H_ <i>This file provides protocol independence for remote aggregate lists</i>
	#define	REMOTE_ENTITY_H_ <i>This file provides protocol independence for reflected aggregates</i>
	#define	REMOTE_OBJ_H_ <i>This file provides protocol independence for Reflected Entities</i>
	#define	REFLECTED_EMITTER_H_ <i>This file provides protocol independence for Reflected Objects</i>
	#define	REMOTE_ENTITY_LIST_H_ <i>This file provides protocol independence for reflected emitter systems</i>
	#define	REMOTE_OBJ_LIST_H_ <i>This file provides protocol independence for Remote Entity Lists</i>
	#define	DtReflectedHlaObjectManager_H_ <i>This file provides protocol independence for Reflected Object Lists</i>
	#define	DtRegControlCbInfo_H_
	#define	DtRegControlCbList_H_
	#define	REMOVE_ENTITY_H_
	#define	DtRemoveObjectCbInfo_H_ <i>This file provides protocol independence for RemoveEntity Interactions</i>
	#define	DtRemoveEntityRPdu_H_
	typedef void	(*DtRemoveEntityRPduCb) (DtRemoveEntityRPdu* pdu, void* usr) <i>DtRemoveEntityRPdu callback function signature</i>
	#define	DtRemoveEntityPdu_H_
	#define	DtRemoveStaticSize
	typedef DtRemoveEntityPdu	DtRemoveEntityInteraction
	typedef void	(*DtRemoveEntityPduCb) (DtRemoveEntityPdu *pdu, void *usr)
	#define	DtRepairResponsePdu_H_
	#define	DtRepairResponseStaticSize
	typedef DtRepairResponsePdu	DtRepairResponseInteraction
	typedef void	(*DtRepairResponsePduCb) (DtRepairResponsePdu *pdu, void *usr)
	#define	DtRepairCompleteStaticSize
	typedef void	(*DtRepairCompletePduCb) (DtRepairCompletePdu *pdu, void *usr)
	#define	REPAIR_COMPLETE_H_
	#define	REPAIR_RESPONSE_H_ <i>This file provides protocol independence for ActionResponse Interactions</i>
	#define	RESUPPLY_H_ <i>This file provides protocol independence for ActionResponse Interactions</i>

#define	DtResupplyCancelPdu_H_ <i>This file provides protocol independence for the Resupply Interactions: ResupplyCancel, ResupplyOffer, ResupplyReceived</i>
#define	DtResupplyCancelStaticSize
typedef DtResupplyCancelPdu	<u>DtResupplyCancelInteraction</u>
typedef void	<u>(*DtResupplyCancelPduCb)</u> (DtResupplyCancelPdu *pdu, void *usr)
#define	DtResupplyOfferStaticSize
#define	DtSUPPLY_VALID
#define	DtSUPPLY_BAD_INDEX
typedef DtResupplyOfferPdu	<u>DtResupplyOfferInteraction</u>
typedef void	<u>(*DtResupplyOfferPduCb)</u> (DtResupplyOfferPdu *pdu, void *usr)
#define	DtResupplyReceivedStaticSize
typedef DtResupplyReceivedPdu	<u>DtResupplyRecInteraction</u>
typedef void	<u>(*DtResupplyReceivedPduCb)</u> (DtResupplyReceivedPdu *pdu, void *usr)
#define	DtRprStateDecoderFactory_H_
#define	DtRprStateEncoderFactory_H_
#define	DtRprFomMapper_H_
#define	DtRprInteractionDecoderFactory_H_
#define	DtRprInteractionEncoderFactory_H_
#define	DtRprInteractionFactory_H_
#define	DtRprHlaObjectFactory_H_
#define	SERVICE_REQUEST_H_
#define	DtServiceRequestPdu_H_ <i>This file provides protocol independence for ActionResponse Interactions</i>
#define	DtServiceRequestStaticSize
#define	DtSUPPLY_VALID
#define	DtSUPPLY_BAD_INDEX
typedef DtServiceRequestPdu	<u>DtServiceRequestInteraction</u>
typedef void	<u>(*DtServiceRequestPduCb)</u> (DtServiceRequestPdu *pdu, void *usr)
#define	SET_DATA_H_
#define	DtSetDataPdu_H_ <i>This file provides protocol independence for ActionResponse Interactions</i>
#define	DtSetDataStaticSize
typedef DtSetDataPdu	<u>DtSetDataInteraction</u>
typedef void	<u>(*DtSetDataPduCb)</u> (DtSetDataPdu *pdu, void *usr)
#define	DtSetDataRPdu_H_
typedef void	<u>(*DtSetDataRPduCb)</u> (DtSetDataRPdu* pdu, void* usr) <i>DtSetDataRPdu callback function signature</i>
#define	<u>DtSetRecordPdu_H_</u>
typedef void	<u>(*DtSetRecordPduCb)</u> (DtSetRecordPdu* pdu, void* usr) <i>DtSetRecordPdu callback function signature</i>
#define	DtSignalPdu_H_
#define	DtSignalStaticSize
typedef DtSignalPdu	<u>DtSignalInteraction</u>

typedef void	<u>(*DtSignalInteractionCb)</u> (DtSignalInteraction *pdu, void *usr)
#define	DtSimanPdu_H_
#define	DtSimpleRprFomMapper_H_
#define	_ACKPRED_H_
int	<u>DtAcknowledgePredicate</u> (DtPdu *pdu, void *arg)
	<i>for use with Connection::DtReadUntil</i>
#define	DtSoldierDecoder_H_
#define	DtSoldierEncoder_H_
#define	START_INTERACTION_H_
#define	DtStartResumePdu_H_
	<i>This file provides protocol independence for StartResume Interactions</i>
#define	DtStartStaticSize
typedef DtStartResumePdu	<u>DtStartResumeInteraction</u>
typedef void	<u>(*DtStartResumePduCb)</u> (DtStartResumePdu *pdu, void *usr)
#define	DtStartResumeRPdu_H_
typedef void	<u>(*DtStartResumeRPduCb)</u> (DtStartResumeRPdu* pdu, void* usr)
#define	<u>STATE_MESSAGE_H_</u>
#define	DtStateRepository_H_
	<i>This file provides protocol independence for state messages</i>
#define	DtStartStopInterCbInfo_H_
#define	DtStartStopInterCbList_H_
#define	DtStopFrzRPdu_H_
typedef void	<u>(*DtStopFreezeRPduCb)</u> (DtStopFreezeRPdu* pdu, void* usr)
	<i>DtStopFreezeRPdu callback function signature</i>
#define	<u>STOP_INTERACTION_H_</u>
#define	DtStopPdu_H_
	<i>This file provides protocol independence for StopFreeze Interactions</i>
#define	DtStopStaticSize
typedef DtStopFreezePdu	<u>DtStopFreezeInteraction</u>
typedef void	<u>(*DtStopFreezePduCb)</u> (DtStopFreezePdu *pdu, void *usr)
#define	DtUpdateControlCbInfo103_H_
#define	DtUpdateControlCbList103_H_
#define	DtVectoringNozzleSystem_H_
#define	DtSuppEmisEntStatePdu_H_
typedef void	<u>(*DtSuppEmisEntStatePduCb)</u> (DtSuppEmisEntStatePdu* pdu, void* usr)
	<i>DtSuppEmisEntStatePdu callback function signature</i>
#define	DtPropulsionSystem_H_
#define	DtSystemHashKey_H_
#define	DtTrackJamDesignatorList_H_
<i>class DtTrackJamDesignatorList: Instances of DtTrackJamDesignatorList are list of globally significant object identifiers</i>	
#define	DtTrackJamDesignator_H_
#define	DtTransferControlRequestPdu_H_

typedef void

#define

#define

typedef void

#define

#define

#define

#define

#define

#define

typedef void

#define

#define

#define

#define

typedef void

#define

#define

#define

#define

typedef DtViewControlPdu

typedef void

DtViewControlPdu*

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

#define

[\(*DtTransferControlRequestPduCb\)](#)(DtTransferControlRequestPdu* pdu,
void* usr)

DtTransferControlRequestPdu callback function signature

DtTransmitterPdu_H_

DtTransmitterStaticSize

[\(*DtTransmitterPduCb\)](#)(DtTransmitterPdu *pdu, void *usr)

DtFREQ_UPPER

DtFREQ_LOWER

DtUnderwaterAcousticBeam_H_

DtUnderwaterAcousticEmitterSystem_H_

DtUnderwaterAcousticAddtnlPassiveAct_H_

DtUnderwaterAcousticPdu_H_

[\(*DtUnderwaterAcousticPduCb\)](#)(DtUnderwaterAcousticPdu* pdu, void* usr)
DtUnderwaterAcousticPdu callback function signature

DtUnderwaterAcousticShaftSpeed_H_

DtUnknownHlaObject_H_

DtUnknownPdu_H_

DtUnknownStaticSize

[\(*DtUnknownPduCb\)](#)(DtUnknownPdu *pdu, void *usr)

DtUpdateControlCbInfo_H_

DtUpdateControlCbList_H_

VIEW_CONTROL_INTERACTION_H_

DtViewControlPdu_H_

This file provides protocol independence for ViewControl Interactions

[DtViewControllInteraction](#)

[\(*DtViewControllInteractionCb\)](#)(DtViewControllInteraction *inter, void *usr)

[DtNewViewControlPdu](#)(DtViewerCommand command)

DtIGNORE_EAST(1<<0)

DtIGNORE_NORTH(1<<1)

DtIGNORE_HEIGHT(1<<2)

DtIGNORE_HEADING(1<<3)

DtIGNORE_PITCH(1<<4)

DtIGNORE_ROLL(1<<5)

DtIGNORE_EAST_OFFSET(1<<6)

DtIGNORE_NORTH_OFFSET(1<<7)

DtIGNORE_HEIGHT_OFFSET(1<<8)

DtIGNORE_RADIUS(1<<9)

DtIGNORE_BODY_HEADING(1<<10)

DtIGNORE_HEADING_OFFSET(1<<11)

DtIGNORE_PITCH_OFFSET(1<<12)

DtIGNORE_FORWARD(1<<13)

DtIGNORE_RIGHT(1<<14)

● #define
● #define
● #define
● #define
● #define
● #define
● #define
● #define
● #define

DtIGNORE_UP(1<<15)
DtIGNORE_BODY_PITCH(1<<16)
DtIGNORE_BODY_ROLL(1<<17)
DtIGNORE_AZIMUTH_RATE(1<<18)
DtIGNORE_RADIAL_OFFSET(1<<19)
DtIGNORE_PRIMARY_ID(1<<20)
DtIGNORE_SECONDARY_ID(1<<21)
DtIGNORE_GROUP_IDS(1<<22)
DtVrIFederateAmbassador_H_

[Alphabetic index](#) [Hierarchy of classes](#)



VR-Link Class Documentation

MÄK Technologies: www.mak.com

In file baseexconn.h:

typedef void ([*DtPostDrainCb](#))(void* usr)

class DtBaseExerciseConn: DtBaseExerciseConn is an abstract base class

Documentation

class DtBaseExerciseConn:

DtBaseExerciseConn is an abstract base class. It provides an application's interface or connection to a network.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vlprint.h:

void DtAbort(void)

DtAbort() exits the program

Documentation

DtAbort() exits the program. The user can provide for different behaviour [e.g. calling abort()] by linking with a another definition DtAbort [i.e. It is in its own object file.], but the user's DtAbort() should not return.

[Alphabetic index](#) [Hierarchy of classes](#)

In file smutil.h:

**int DtAcknowledgePredicate(DtAcknowledgePdu
pdu, DtAckPredOpt *opt)*

Return 1 if to, from, and ackFlag match between pkt and arg and positive response

Documentation

Return 1 if to, from, and ackFlag match between pkt and arg and positive response. Return negAckRetVal if match w/ neg Reponse. Return 0 otherwise. If (from.DtEntity == DtDeferredEntityId and hosts match), then from.DtEntity will be set to from.DtEntity in PDU.

In file libmatrix.h:

void [DtDcmCopy](#)(DtConstDcm src, DtDcmRef dest)

make orthonormal, pick most important

Documentation

make orthonormal, pick most important. axis

[Alphabetic index](#) [Hierarchy of classes](#)

In file libmatrix.h:

**double [DtDcmDcmRotTrace](#)(DtConstDcm a,
DtConstDcm b)**

Returns the trace of the rotation matrix, R, between between matrices A and B

Documentation

Returns the trace of the rotation matrix, R, between between matrices A and B. If A is a DCM between frame body and world, and B is that same frame rotated through angle theta about an arbitrary axis, then trace of R [where $B = A * R$, or $R = \text{inv}(A) * B$] is $2 * \cos(\text{theta}) + 1$.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vlprint.h:

void [DtFatalError](#)(const char *s, ...)

These functions print messages using stdio depending on the notify level

Documentation

These functions print messages using stdio depending on the notify level. Fatal and Warn print to stderr. Info prints to stdout. Fatal funcs call DtAbort at end, even if (notifyLevel < DtNIFatal) Fatal prints are preceded by "***FATAL ERROR**:" the DtFatalError print is followed by \n Perror prints are followed by a call to perror()

In file vutil.h:

int DtIsInList(const int *lst, int num)

Returns -1 if num is not in lst

Documentation

Returns -1 if num is not in lst. Otherwise, returns the index in the list at which num is first found. lst must be a zero-terminated array of integers.

[Alphabetic index](#) [Hierarchy of classes](#)

In file topocoord.h:

```
void DtLatLon\_to\_TopoToGeoc(const  
DtGeodeticCoord& g, DtDcmRef topoToGeoc)
```

From latitude and longitude, we can produce the rotation matrices that rotate between geocentric and topographic-down orientations

Documentation

From latitude and longitude, we can produce the rotation matrices that rotate between geocentric and topographic-down orientations. VR-Link uses post-multiplication for its rotation matrices. For example, the matrix produced by `DtLatLon_to_TopoToGeoc` can be used to rotate a topographic vector to a geocentric vector like this: `DtDcmVecMul(topoToGeoc, Vtopo, Vgeoc);`

In file vlttime.h:

void DtSetSimTime(DtTime NewSimTime)

Set the current value of VR-Link simulation time

Documentation

Set the current value of VR-Link simulation time. Should be called once per simulation frame to let VR-Link know the value of simulation time for the frame.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vlutil.h:

char* **DtStrdup**(const char* str)

Strdup wrapper

Documentation

Strdup wrapper. Returns NULL if str is NULL. Otherwise returns a newed char* containing a copy of str.
Caller deletes using delete[].

[Alphabetic index](#) [Hierarchy of classes](#)

In file vlttime.h:

DtTime DtTimeNow(void)

Obsolete

Documentation

Obsolete. This function exists for backwards compatibility with older versions of VR-Link. Use DtGetElapsedRealTime instead.

[Alphabetic index](#) [Hierarchy of classes](#)

In file utmcoord.h:

```
void DtUtmInit(const DtDegMinSec& latRef, const DtDegMinSec& lonRef, int standardOffsets=1, int verbose=1)
```

Sets reference point for all DtUtmCoords Vrlink presently supports only one UTM reference at a time, but the reference point may be reinitialized at any point

Documentation

Sets reference point for all DtUtmCoords Vrlink presently supports only one UTM reference at a time, but the reference point may be reinitialized at any point.

If standardOffsets is true, the standard false easting (500,000 m) and false northing (10,000,000 m only when south of equator) will be applied.

DtUtmInit expects lonRef between +/- 180. The UTM zone will be set by the zone that the reference point falls in, where zone N extends from (N*6 - 186) to (N*6 - 180).

In file entitytypes.h:

enum DtMunitionUSAntiArmorUnguided

 Dt25mm

 Dt105m

 Dt120mm

Documentation

 Dt25mm

 Dt105m

 Dt120mm

In file entitytypes.h:

enum DtMunitionUSAntiPersonnelUnguided

 Dt7_62mm

Documentation

 Dt7_62mm

In file entitytypes.h:

enum DtMunitionUKAntiPersonnelUnguided

UK

 Dt7_62mmUK

Documentation

UK

 Dt7_62mmUK

In file entitytypes.h:

enum DtMunitionUSSRAntiPersonnelUnguided

 Dt7_62mmUSSR

Documentation

 Dt7_62mmUSSR

In file entitytypes.h:

enum DtPlatformAirUSAttack

 DtA10

Documentation

 DtA10

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtUserProtocol

-
-  DtUserProtocolOther
 -  DtCCSIL
 -  DtA2ATD_SINGARS_ERF
 -  DtA2ATD_CAC2
 -  DtBattle_Command
 -  DtModSAF_Text_Radio
 -  DtCCTT_SINGARS_ERF_LOCKOUT
 -  DtCCTT_SINGARS_ERF_HOPSET
 -  DtCCTT_SINGARS_OTAR
 -  DtCCTT_SINGARS_DATA
 -  DtModSAF_FWA_Forward_Air_Controller
 -  DtModSAF_Threat_ADA_C3
 -  DtModSAF_Artillery_Fire_Control
 -  DtAGTS
 -  DtUserGC3
 -  DtWNCP_data
-

Documentation

-  DtUserProtocolOther
-  DtCCSIL
-  DtA2ATD_SINGARS_ERF
-  DtA2ATD_CAC2
-  DtBattle_Command
-  DtModSAF_Text_Radio
-  DtCCTT_SINGARS_ERF_LOCKOUT

- DtCCTT_SINGARS_ERF_HOPSET
- DtCCTT_SINGARS_OTAR
- DtCCTT_SINGARS_DATA
- DtModSAF_FWA_Forward_Air_Controller
- DtModSAF_Threat_ADA_C3
- DtModSAF_Artillery_Fire_Control
- DtAGTS
- DtUserGC3
- DtWNCP_data

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtMunitionUSSRAntiAirGuided

USSR

-  DtAA3Anan
 -  DtAA9Amos
 -  DtSA2Guideline
 -  DtSA3Goa
 -  DtSA4Ganef
 -  DtSA6Gainful
 -  DtSA7Grail
 -  DtSA8Gecko
 -  DtSA9Gaskin
 -  DtSA10Grumble
-

Documentation

USSR

-  DtAA3Anan
-  DtAA9Amos
-  DtSA2Guideline
-  DtSA3Goa
-  DtSA4Ganef
-  DtSA6Gainful
-  DtSA7Grail
-  DtSA8Gecko
-  DtSA9Gaskin
-  DtSA10Grumble

In file disenums.h:

enum DtEncodingType

-  DtEncodingTypeOther
 -  DtEncoding_8_bit_mu_law
 -  DtCVSD_per_MIL_STD_188_113
 -  DtADPCM_per_CCITT_G721
 -  DtEncoding_16_bit_linear_PCM
 -  DtEncoding_8_bit_linear_PCM
 -  DtVQ_Vector_Quantization
-

Documentation

-  DtEncodingTypeOther
-  DtEncoding_8_bit_mu_law
-  DtCVSD_per_MIL_STD_188_113
-  DtADPCM_per_CCITT_G721
-  DtEncoding_16_bit_linear_PCM
-  DtEncoding_8_bit_linear_PCM
-  DtVQ_Vector_Quantization

In file entitytypes.h:

enum DtMunitionUSAntiRadarGuided

 DtAGM88HARM

Documentation

 DtAGM88HARM

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtPlatformAirUSAttackHelicopter

-
-  DtAH64
 -  DtAH1W
 -  DtRAH66
-

Documentation

-  DtAH64
-  DtAH1W
-  DtRAH66

In file entitytypes.h:

enum DtMunitionUSAntiAirGuided

US

-
-  DtSidewinder
 -  DtAMRAAM
 -  DtHawk
 -  DtPhoenix
 -  DtRoland
 -  DtSeaSparrow
 -  DtSparrow
 -  DtStinger
 -  DtPatriot
-

Documentation

US

-  DtSidewinder
-  DtAMRAAM
-  DtHawk
-  DtPhoenix
-  DtRoland
-  DtSeaSparrow
-  DtSparrow
-  DtStinger
-  DtPatriot

In file disenums.h:

enum DtTacticalDataLinkType

-
-  DtTacticalDataLinkOther
 -  DtAbbreviated_Command_And_Control
 -  DtTdlGC3
-

Documentation

-  DtTacticalDataLinkOther
-  DtAbbreviated_Command_And_Control
-  DtTdlGC3

In file disenums.h:

enum DtActionId

- DtActionOther
- DtActionLocalStorage
- DtActionOutOfAmmo
- DtActionKilledInAction
- DtActionDamage
- DtActionMobilityDisabled
- DtActionFireDisabled
- DtActionOutOfFuel
- DtActionRecallCheckpointData
- DtActionRecallInitialParameters
- DtActionInitiateTetherLead
- DtActionInitiateTetherFollow
- DtActionUntether
- DtActionInitiateServiceStationResupply
- DtActionInitiateTailgateResupply
- DtActionInitiateHitchLead
- DtActionInitiateHitchFollow
- DtActionUnhitch
- DtActionMount
- DtActionDismount
- DtActionStartDRC
- DtActionStopDRC
- DtActionDataQuery
- DtActionStatusRequest
- DtActionSendObjectStateData
- DtActionReconstitute

- DtActionLockSiteConfiguration
 - DtActionUnlockSiteConfiguration
 - DtActionUpdateSiteConfiguration
 - DtActionQuerySiteConfiguration
 - DtActionTetheringInformation
 - DtActionMountIntent
 - DtActionOther2
 - DtActionAcceptSubscription
 - DtActionUnsubscribe
 - DtActionTeleportEntity
 - DtActionChangeAggregateState
 - DtActionRequestStartPDU
 - DtActionWakeUpGetReadyForInitialization
 - DtActionInitializeInternalParameters
 - DtActionSendPlanData
 - DtActionSynchronizeInternalClocks
 - DtActionRun
 - DtActionSaveInternalParameters
 - DtActionSimulateMalfunction
 - DtActionJoinExercise
 - DtActionResignExercise
 - DtActionTimeAdvance
-

Documentation

- DtActionOther
- DtActionLocalStorage
- DtActionOutOfAmmo
- DtActionKilledInAction
- DtActionDamage
- DtActionMobilityDisabled
- DtActionFireDisabled

- DtActionOutOfFuel
- DtActionRecallCheckpointData
- DtActionRecallInitialParameters
- DtActionInitiateTetherLead
- DtActionInitiateTetherFollow
- DtActionUntether
- DtActionInitiateServiceStationResupply
- DtActionInitiateTailgateResupply
- DtActionInitiateHitchLead
- DtActionInitiateHitchFollow
- DtActionUnhitch
- DtActionMount
- DtActionDismount
- DtActionStartDRC
- DtActionStopDRC
- DtActionDataQuery
- DtActionStatusRequest
- DtActionSendObjectStateData
- DtActionReconstitute
- DtActionLockSiteConfiguration
- DtActionUnlockSiteConfiguration
- DtActionUpdateSiteConfiguration
- DtActionQuerySiteConfiguration
- DtActionTetheringInformation
- DtActionMountIntent
- DtActionOther2
- DtActionAcceptSubscription
- DtActionUnsubscribe
- DtActionTeleportEntity
- DtActionChangeAggregateState
- DtActionRequestStartPDU
- DtActionWakeUpGetReadyForInitialization

- DtActionInitializeInternalParameters
- DtActionSendPlanData
- DtActionSynchronizeInternalClocks
- DtActionRun
- DtActionSaveInternalParameters
- DtActionSimulateMalfunction
- DtActionJoinExercise
- DtActionResignExercise
- DtActionTimeAdvance

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtRadioDetailedModulationAmplitude

-  DtRadioDetailModOther
 -  DtAfsk
 -  DtAm
 -  DtCw
 -  DtDsb
 -  DtIsb
 -  DtLsb
 -  DtSsbFull
 -  DtSsbReduc
 -  DtUsb
 -  DtVsb
-

Documentation

-  DtRadioDetailModOther
-  DtAfsk
-  DtAm
-  DtCw
-  DtDsb
-  DtIsb
-  DtLsb
-  DtSsbFull
-  DtSsbReduc
-  DtUsb
-  DtVsb

In file entitytypes.h:

enum DtPlatformSurface

-  DtCarrier
 -  DtCommandShip
 -  DtGuidedMissileCruiser
 -  DtGuidedMissileDestroyer
 -  DtDestroyer
 -  DtGuidedMissileFrigate
 -  DtPatrolCraft
 -  DtMineCounterMeasureShip
 -  DtDockLandingShip
 -  DtTankLandingShip
 -  DtLandingCraft
 -  DtFrigate
 -  DtBattleship
 -  DtHeavyCruiser
 -  DtDestroyerTender
 -  DtAmphibiousAssaultShip
 -  DtAmphibiousCargoShip
 -  DtAmphibiousTransportDock
 -  DtAmmunitionShip
 -  DtCombatStoresShip
-

Documentation

-  DtCarrier
-  DtCommandShip
-  DtGuidedMissileCruiser

- DtGuidedMissileDestroyer
- DtDestroyer
- DtGuidedMissileFrigate
- DtPatrolCraft
- DtMineCounterMeasureShip
- DtDockLandingShip
- DtTankLandingShip
- DtLandingCraft
- DtFrigate
- DtBattleship
- DtHeavyCruiser
- DtDestroyerTender
- DtAmphibiousAssaultShip
- DtAmphibiousCargoShip
- DtAmphibiousTransportDock
- DtAmmunitionShip
- DtCombatStoresShip

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtDetailedModulationCombination

 DtRadioDetailModCombo

 DtAmplitudeAnglePulse

Documentation

 DtRadioDetailModCombo

 DtAmplitudeAnglePulse

In file entitytypes.h:

enum DtMunitionDomain

-
-  DtMunitionDomainOther
 -  DtAntiAir
 -  DtAntiArmor
 -  DtAntiGuidedMunition
 -  DtAntiRadar
 -  DtAntiSatellite
 -  DtAntiShip
 -  DtAntiSubmarine
 -  DtAntiPersonnel
 -  DtBattlefieldSupport
 -  DtStrategic
-

Documentation

-  DtMunitionDomainOther
-  DtAntiAir
-  DtAntiArmor
-  DtAntiGuidedMunition
-  DtAntiRadar
-  DtAntiSatellite
-  DtAntiShip
-  DtAntiSubmarine
-  DtAntiPersonnel
-  DtBattlefieldSupport
-  DtStrategic

In file entitytypes.h:

enum DtMunitionAntiArmor

 DtAntiArmorGuided

 DtAntiArmorUnguided

Documentation

 DtAntiArmorGuided

 DtAntiArmorUnguided

In file entitytypes.h:

enum DtMunitionAntiPersonnel

 DtAntiPersonnelGuided

 DtAntiPersonnelUnguided

Documentation

 DtAntiPersonnelGuided

 DtAntiPersonnelUnguided

In file entitytypes.h:

enum DtMunitionAntiRadar

-
-  DtAntiRadarGuided
 -  DtAntiRadarUnguided
-

Documentation

-  DtAntiRadarGuided
-  DtAntiRadarUnguided

In file entitytypes.h:

enum DtMunitionAntiShip

-
-  DtAntiShipGuided
 -  DtAntiShipUnguided
-

Documentation

-  DtAntiShipGuided
-  DtAntiShipUnguided

In file entitytypes.h:

enum DtMunitionAntiSubmarine

 DtAntiSubmarineGuided

 DtAntiSubmarineUnguided

Documentation

 DtAntiSubmarineGuided

 DtAntiSubmarineUnguided

In file entitytypes.h:

enum DtPlatformAir

-  DtFighter
 -  DtAttack
 -  DtBomber
 -  DtCargo
 -  DtAntiSubmarinePatrol
 -  DtAttackHelicopter
 -  DtUtilityHelicopter
 -  DtSeaGoingHelicopter
 -  DtElectronicWarfare
 -  DtUnmannedAirVehicle
 -  DtReconnaissance
 -  DtSpecialOperations
-

Documentation

-  DtFighter
-  DtAttack
-  DtBomber
-  DtCargo
-  DtAntiSubmarinePatrol
-  DtAttackHelicopter
-  DtUtilityHelicopter
-  DtSeaGoingHelicopter
-  DtElectronicWarfare
-  DtUnmannedAirVehicle
-  DtReconnaissance

In file disenums.h:

enum DtArtParamType

Articulated parts

-  DtApPosition
 -  DtApPositionRate
 -  DtApExtension
 -  DtApExtensionRate
 -  DtApX
 -  DtApXRate
 -  DtApY
 -  DtApYRate
 -  DtApZ
 -  DtApZRate
 -  DtApAzimuth
 -  DtApAzimuthRate
 -  DtApElevation
 -  DtApElevationRate
 -  DtApRotation
 -  DtApRotationRate
-

Documentation

Articulated parts

-  DtApPosition
-  DtApPositionRate
-  DtApExtension
-  DtApExtensionRate
-  DtApX
-  DtApXRate

- DtApY
- DtApYRate
- DtApZ
- DtApZRate
- DtApAzimuth
- DtApAzimuthRate
- DtApElevation
- DtApElevationRate
- DtApRotation
- DtApRotationRate

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtAppearAttributeBitShift

-  DtAppPaintSchemeBitShift
 -  DtAppMobilityKillBitShift
 -  DtAppFirePowerKillBitShift
 -  DtAppDamageBitShift
 -  DtAppSmokeBitShift
 -  DtAppTrailingEffectsBitShift
 -  DtAppHatchBitShift
 -  DtAppLightsBitShift
 -  DtAppFlamingBitShift
 -  DtAppFrozenBitShift
 -  DtAppPowerPlantStatusBitShift
 -  DtAppFinalPduBitShift
 -  DtAppDummyBitShift
-

Documentation

-  DtAppPaintSchemeBitShift
-  DtAppMobilityKillBitShift
-  DtAppFirePowerKillBitShift
-  DtAppDamageBitShift
-  DtAppSmokeBitShift
-  DtAppTrailingEffectsBitShift
-  DtAppHatchBitShift
-  DtAppLightsBitShift
-  DtAppFlamingBitShift
-  DtAppFrozenBitShift

- DtAppPowerPlantStatusBitShift
- DtAppFinalPduBitShift
- DtAppDummyBitShift

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtPlatformLand

-
-  DtTank
 -  DtArmoredFightingVehicle
 -  DtArmoredUtilityVehicle
 -  DtSelfPropelledArtillery
 -  DtTowedArtillery
 -  DtSmallWheeledUtilityVehicle
 -  DtLargeWheeledUtilityVehicle
 -  DtSmallTrackedUtilityVehicle
 -  DtLargeTrackedUtilityVehicle
 -  DtMortar
-

Documentation

-  DtTank
-  DtArmoredFightingVehicle
-  DtArmoredUtilityVehicle
-  DtSelfPropelledArtillery
-  DtTowedArtillery
-  DtSmallWheeledUtilityVehicle
-  DtLargeWheeledUtilityVehicle
-  DtSmallTrackedUtilityVehicle
-  DtLargeTrackedUtilityVehicle
-  DtMortar

In file entitytypes.h:

enum DtNumberOfIndividualLifeForms

-
-  DtFireTeam
 -  DtSquad
 -  DtPlatoon
 -  DtCompany
 -  DtBattalion
 -  DtRegiment
 -  DtDivision
 -  DtCorps
 -  DtArmy
-

Documentation

-  DtFireTeam
-  DtSquad
-  DtPlatoon
-  DtCompany
-  DtBattalion
-  DtRegiment
-  DtDivision
-  DtCorps
-  DtArmy

In file disenums.h:

enum DtAttributeChangeResult

 DtAttributeChangeResultOther

Documentation

 DtAttributeChangeResultOther

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtMunitionUSBattlefieldSupportGuided

 DtBGM109Tomahawk

Documentation

 DtBGM109Tomahawk

In file entitytypes.h:

enum DtMunitionUSAntiArmorGuided

 DtBGM71TOW

 DtHellfire

 DtMaverick

Documentation

 DtBGM71TOW

 DtHellfire

 DtMaverick

In file entitytypes.h:

enum

DtPlatformLandUSSRArmoredFightingVehicle

-
-  DtBMP1
 -  DtBMP2
 -  DtBRDM1
 -  DtMTLB
-

Documentation

-  DtBMP1
-  DtBMP2
-  DtBRDM1
-  DtMTLB

In file entitytypes.h:

enum DtF16

 DtBasicF16

 DtF16C

Documentation

 DtBasicF16

 DtF16C

In file entitytypes.h:

enum DtT72Specific

 DtBasicT72

Documentation

 DtBasicT72

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtMunitionUSBattlefieldSupportUnguided

-
-  DtBattlefield20mm
 -  DtBattlefield30mm
 -  DtBattlefield105mm
 -  DtBattlefield120mm
-

Documentation

-  DtBattlefield20mm
-  DtBattlefield30mm
-  DtBattlefield105mm
-  DtBattlefield120mm

In file entitytypes.h:

enum [DtMunitionUKBattlefieldSupportUnguided](#)

 DtBattlefield105mmUK

Documentation

 DtBattlefield105mmUK

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtMunitionBattlefieldSupport

 DtBattlefieldSupportGuided

 DtBattlefieldSupportUnguided

Documentation

 DtBattlefieldSupportGuided

 DtBattlefieldSupportUnguided

In file entitytypes.h:

enum DtLHASpecific

-
-  DtTarawa
 -  DtSaipan
 -  DtBelleauWood
 -  DtNassau
 -  DtPeleliu
-

Documentation

-  DtTarawa
-  DtSaipan
-  DtBelleauWood
-  DtNassau
-  DtPeleliu

In file entitytypes.h:

enum DtCulturalFeatureLand

-
-  DtBridgeConcreteTwoLane
 -  DtBuildingOneStory
 -  DtBuildingTwoStory
 -  DtBuildingThreeStory
 -  DtBuildingFourStory
 -  DtDragonsTeeth
-

Documentation

-  DtBridgeConcreteTwoLane
-  DtBuildingOneStory
-  DtBuildingTwoStory
-  DtBuildingThreeStory
-  DtBuildingFourStory
-  DtDragonsTeeth

In file entitytypes.h:

enum DtPlatformAirUSCargo

 DtC130

 DtKC130

Documentation

 DtC130

 DtKC130

In file entitytypes.h:

enum DtPlatformSurfaceUSCarrier

 DtCV59

Documentation

 DtCV59

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtPlatformLandUKTank

UK

 DtChallengerMBT

Documentation

UK

 DtChallengerMBT

In file entitytypes.h:

enum

DtPlatformLandAnySmallWheeledUtilityVehicle

Any Country

 DtCivilianCar

Documentation

Any Country

 DtCivilianCar

In file entitytypes.h:

enum DtEnvironmentalAir

-
-  DtEnvironmentalDomainAirOther
 -  DtSmoke
 -  DtFog
 -  DtCloud
 -  DtCloudWithRainFalling
 -  DtCloudWithSnowFalling
-

Documentation

-  DtEnvironmentalDomainAirOther
-  DtSmoke
-  DtFog
-  DtCloud
-  DtCloudWithRainFalling
-  DtCloudWithSnowFalling

In file disenums.h:

enum DtDetailedModulationUnmodulated

 DtRadioDetailModUnmod

 DtContinuous

Documentation

 DtRadioDetailModUnmod

 DtContinuous

In file entitytypes.h:

enum DtCountryCode

-
-  DtCountryCodeOther
 -  DtFrance
 -  DtIraq
 -  DtUnionOfSovietSocialistRepublics
 -  DtUnitedKingdom
 -  DtUnitedStates
-

Documentation

-  DtCountryCodeOther
-  DtFrance
-  DtIraq
-  DtUnionOfSovietSocialistRepublics
-  DtUnitedKingdom
-  DtUnitedStates

In file disenums.h:

enum DtCreateObjectResult

 DtCreateObjectResultOther

Documentation

 DtCreateObjectResultOther

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtCulturalFeatureDomain

-
-  DtCulturalFeatureDomainOther
 -  DtCulturalFeatureDomainLand
 -  DtCulturalFeatureDomainAir
 -  DtCulturalFeatureDomainSurface
 -  DtCulturalFeatureDomainSubsurface
 -  DtCulturalFeatureDomainSpace
-

Documentation

-  DtCulturalFeatureDomainOther
-  DtCulturalFeatureDomainLand
-  DtCulturalFeatureDomainAir
-  DtCulturalFeatureDomainSurface
-  DtCulturalFeatureDomainSubsurface
-  DtCulturalFeatureDomainSpace

In file entitytypes.h:

enum DtEntityKind

-
-  DtEntityKindOther
 -  DtPlatform
 -  DtMunition
 -  DtLifeForm
 -  DtEnvironmental
 -  DtCulturalFeature
 -  DtSupply
-

Documentation

-  DtEntityKindOther
-  DtPlatform
-  DtMunition
-  DtLifeForm
-  DtEnvironmental
-  DtCulturalFeature
-  DtSupply

In file entitytypes.h:

enum DtPlatformLandUSSRSelfTowedArtillery

 DtD20

Documentation

 DtD20

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtDetailedModulationAmplitudeAngle

-
-  DtRadioModAmpAngOther
 -  DtDetailedAmplitudeAndAngle
-

Documentation

-  DtRadioModAmpAngOther
-  DtDetailedAmplitudeAndAngle

In file entitytypes.h:

enum DtLifeFormLand

 DtDismountedInfantry

Documentation

 DtDismountedInfantry

[Alphabetic index](#) [Hierarchy of classes](#)

In file utmcoord.h:

enum DtPolarDirection

-  DtEast
 -  DtWest
 -  DtNorth
 -  DtSouth
-

Documentation

-  DtEast
-  DtWest
-  DtNorth
-  DtSouth

In file disenums.h:

enum DtArtPartType

- DtEntityBase
- DtRudder
- DtLeftFlap
- DtRightFlap
- DtLeftAileron
- DtRightAileron
- DtPeriscope
- DtGenericAntenna
- DtSnorkel
- DtLandingGear
- DtTailHook
- DtSpeedBrake
- DtLeftWeaponBayDoor
- DtRightWeaponBayDoor
- DtTankAPCHatch
- DtWingSweep
- DtPrimaryTurret1
- DtPrimaryTurret2
- DtPrimaryTurret3
- DtPrimaryTurret4
- DtPrimaryTurret5
- DtPrimaryTurret6
- DtPrimaryTurret7
- DtPrimaryTurret8
- DtPrimaryTurret9
- DtPrimaryTurret10

- DtPrimaryGun1
- DtPrimaryGun2
- DtPrimaryGun3
- DtPrimaryGun4
- DtPrimaryGun5
- DtPrimaryGun6
- DtPrimaryGun7
- DtPrimaryGun8
- DtPrimaryGun9
- DtPrimaryGun10
- DtPrimaryLauncher1
- DtPrimaryLauncher2
- DtPrimaryLauncher3
- DtPrimaryLauncher4
- DtPrimaryLauncher5
- DtPrimaryLauncher6
- DtPrimaryLauncher7
- DtPrimaryLauncher8
- DtPrimaryLauncher9
- DtPrimaryLauncher10
- DtPrimaryDefenseSys1
- DtPrimaryDefenseSys2
- DtPrimaryDefenseSys3
- DtPrimaryDefenseSys4
- DtPrimaryDefenseSys5
- DtPrimaryDefenseSys6
- DtPrimaryDefenseSys7
- DtPrimaryDefenseSys8
- DtPrimaryDefenseSys9
- DtPrimaryDefenseSys10
- DtPrimaryRadar1
- DtPrimaryRadar2

- DtPrimaryRadar3
- DtPrimaryRadar4
- DtPrimaryRadar5
- DtPrimaryRadar6
- DtPrimaryRadar7
- DtPrimaryRadar8
- DtPrimaryRadar9
- DtPrimaryRadar10
- DtSecondaryTurret1
- DtSecondaryTurret2
- DtSecondaryTurret3
- DtSecondaryTurret4
- DtSecondaryTurret5
- DtSecondaryTurret6
- DtSecondaryTurret7
- DtSecondaryTurret8
- DtSecondaryTurret9
- DtSecondaryTurret10
- DtSecondaryGun1
- DtSecondaryGun2
- DtSecondaryGun3
- DtSecondaryGun4
- DtSecondaryGun5
- DtSecondaryGun6
- DtSecondaryGun7
- DtSecondaryGun8
- DtSecondaryGun9
- DtSecondaryGun10
- DtSecondaryLauncher1
- DtSecondaryLauncher2
- DtSecondaryLauncher3
- DtSecondaryLauncher4

- DtSecondaryLauncher5
 - DtSecondaryLauncher6
 - DtSecondaryLauncher7
 - DtSecondaryLauncher8
 - DtSecondaryLauncher9
 - DtSecondaryLauncher10
 - DtSecondaryDefenseSys1
 - DtSecondaryDefenseSys2
 - DtSecondaryDefenseSys3
 - DtSecondaryDefenseSys4
 - DtSecondaryDefenseSys5
 - DtSecondaryDefenseSys6
 - DtSecondaryDefenseSys7
 - DtSecondaryDefenseSys8
 - DtSecondaryDefenseSys9
 - DtSecondaryDefenseSys10
 - DtSecondaryRadar1
 - DtSecondaryRadar2
 - DtSecondaryRadar3
 - DtSecondaryRadar4
 - DtSecondaryRadar5
 - DtSecondaryRadar6
 - DtSecondaryRadar7
 - DtSecondaryRadar8
 - DtSecondaryRadar9
 - DtSecondaryRadar10
-

Documentation

- DtEntityBase
- DtRudder
- DtLeftFlap

- DtRightFlap
- DtLeftAileron
- DtRightAileron
- DtPeriscope
- DtGenericAntenna
- DtSnorkel
- DtLandingGear
- DtTailHook
- DtSpeedBrake
- DtLeftWeaponBayDoor
- DtRightWeaponBayDoor
- DtTankAPCHatch
- DtWingSweep
- DtPrimaryTurret1
- DtPrimaryTurret2
- DtPrimaryTurret3
- DtPrimaryTurret4
- DtPrimaryTurret5
- DtPrimaryTurret6
- DtPrimaryTurret7
- DtPrimaryTurret8
- DtPrimaryTurret9
- DtPrimaryTurret10
- DtPrimaryGun1
- DtPrimaryGun2
- DtPrimaryGun3
- DtPrimaryGun4
- DtPrimaryGun5
- DtPrimaryGun6
- DtPrimaryGun7
- DtPrimaryGun8
- DtPrimaryGun9

- DtPrimaryGun10
- DtPrimaryLauncher1
- DtPrimaryLauncher2
- DtPrimaryLauncher3
- DtPrimaryLauncher4
- DtPrimaryLauncher5
- DtPrimaryLauncher6
- DtPrimaryLauncher7
- DtPrimaryLauncher8
- DtPrimaryLauncher9
- DtPrimaryLauncher10
- DtPrimaryDefenseSys1
- DtPrimaryDefenseSys2
- DtPrimaryDefenseSys3
- DtPrimaryDefenseSys4
- DtPrimaryDefenseSys5
- DtPrimaryDefenseSys6
- DtPrimaryDefenseSys7
- DtPrimaryDefenseSys8
- DtPrimaryDefenseSys9
- DtPrimaryDefenseSys10
- DtPrimaryRadar1
- DtPrimaryRadar2
- DtPrimaryRadar3
- DtPrimaryRadar4
- DtPrimaryRadar5
- DtPrimaryRadar6
- DtPrimaryRadar7
- DtPrimaryRadar8
- DtPrimaryRadar9
- DtPrimaryRadar10
- DtSecondaryTurret1

- DtSecondaryTurret2
- DtSecondaryTurret3
- DtSecondaryTurret4
- DtSecondaryTurret5
- DtSecondaryTurret6
- DtSecondaryTurret7
- DtSecondaryTurret8
- DtSecondaryTurret9
- DtSecondaryTurret10
- DtSecondaryGun1
- DtSecondaryGun2
- DtSecondaryGun3
- DtSecondaryGun4
- DtSecondaryGun5
- DtSecondaryGun6
- DtSecondaryGun7
- DtSecondaryGun8
- DtSecondaryGun9
- DtSecondaryGun10
- DtSecondaryLauncher1
- DtSecondaryLauncher2
- DtSecondaryLauncher3
- DtSecondaryLauncher4
- DtSecondaryLauncher5
- DtSecondaryLauncher6
- DtSecondaryLauncher7
- DtSecondaryLauncher8
- DtSecondaryLauncher9
- DtSecondaryLauncher10
- DtSecondaryDefenseSys1
- DtSecondaryDefenseSys2
- DtSecondaryDefenseSys3

- DtSecondaryDefenseSys4
- DtSecondaryDefenseSys5
- DtSecondaryDefenseSys6
- DtSecondaryDefenseSys7
- DtSecondaryDefenseSys8
- DtSecondaryDefenseSys9
- DtSecondaryDefenseSys10
- DtSecondaryRadar1
- DtSecondaryRadar2
- DtSecondaryRadar3
- DtSecondaryRadar4
- DtSecondaryRadar5
- DtSecondaryRadar6
- DtSecondaryRadar7
- DtSecondaryRadar8
- DtSecondaryRadar9
- DtSecondaryRadar10

In file entitytypes.h:

enum DtEnvironmentalDomain

-
-  DtEnvironmentalDomainOther
 -  DtEnvironmentalDomainLand
 -  DtEnvironmentalDomainAir
 -  DtEnvironmentalDomainSurface
 -  DtEnvironmentalDomainSubsurface
 -  DtEnvironmentalDomainSpace
-

Documentation

-  DtEnvironmentalDomainOther
-  DtEnvironmentalDomainLand
-  DtEnvironmentalDomainAir
-  DtEnvironmentalDomainSurface
-  DtEnvironmentalDomainSubsurface
-  DtEnvironmentalDomainSpace

In file entitytypes.h:

enum DtPlatformAirUSFighter

 DtF117

 DtF16

 DtF15

 DtF15E

 DtF18

Documentation

 DtF117

 DtF16

 DtF15

 DtF15E

 DtF18

In file entitytypes.h:

enum DtPlatformSurfaceUSGuidedMissileFrigate

 DtFFG7

Documentation

 DtFFG7

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtDetailedModulationAngle

 DtRadioModeAngleOther

 DtFm

 DtFsk

 DtPm

Documentation

 DtRadioModeAngleOther

 DtFm

 DtFsk

 DtPm

In file disenums.h:

enum DtRadioSpreadSpectrum

-  DtRadioSpreadOther
 -  DtFrequencyHopping
 -  DtPseudoNoise
 -  DtTimeHopping
 -  DtPseudoNoiseFrequencyHopping
 -  DtFrequencyHoppingTimeHopping
 -  DtPseudoNoiseTimeHopping
 -  DtPseudoNoiseFrequencyHoppingTimeHopping
-

Documentation

-  DtRadioSpreadOther
-  DtFrequencyHopping
-  DtPseudoNoise
-  DtTimeHopping
-  DtPseudoNoiseFrequencyHopping
-  DtFrequencyHoppingTimeHopping
-  DtPseudoNoiseTimeHopping
-  DtPseudoNoiseFrequencyHoppingTimeHopping

In file entitytypes.h:

enum

DtPlatformLandUSSRLargeWheeledUtilityVehicle

 DtZIL157

 DtGAZ66

 DtUral375

Documentation

 DtZIL157

 DtGAZ66

 DtUral375

In file entitytypes.h:

enum DtPlatformAirFranceUtilityHelicopter

France

 DtGazelle

Documentation

France

 DtGazelle

In file entitytypes.h:

enum

DtPlatformLandUSSmallWheeledUtilityVehicle

 DtUSCivilianCar

 DtHMMWV

Documentation

 DtUSCivilianCar

 DtHMMWV

[Alphabetic index](#) [Hierarchy of classes](#)

In file lcpdu.h:

enum DtLgrProcess

Receiving application may filter lcpdu based on this start-up flag

 DtLGR_NONE

 DtLGR_MINE

 DtLGR_ALL

Documentation

Receiving application may filter lcpdu based on this start-up flag

 DtLGR_NONE

 DtLGR_MINE

 DtLGR_ALL

In file lcpdu.h:

enum DtLgrDirection

Play forwards or backwards when in LGR_PLAY mode

-  DtLGR_STOPPED
 -  DtLGR_NOPKT
 -  DtLGR_FORWARDS
 -  DtLGR_BACKWARDS
-

Documentation

Play forwards or backwards when in LGR_PLAY mode

-  DtLGR_STOPPED
-  DtLGR_NOPKT
-  DtLGR_FORWARDS
-  DtLGR_BACKWARDS

In file lcpdu.h:

enum DtLgrFilePosition

Location status

-  DtLGR_NOFILE
 -  DtLGR_START
 -  DtLGR_MIDDLE
 -  DtLGR_END
-

Documentation

Location status

-  DtLGR_NOFILE
-  DtLGR_START
-  DtLGR_MIDDLE
-  DtLGR_END

In file lcpdu.h:

enum DtLgrStyle

 DtLGR_PRISTINE

 DtLGR_RVA

Documentation

 DtLGR_PRISTINE

 DtLGR_RVA

In file entitytypes.h:

enum DtSizeQualifier

-  DtOther
 -  DtVerySmall
 -  DtSmall
 -  DtMedium
 -  DtLarge
 -  DtVeryLarge
-

Documentation

-  DtOther
-  DtVerySmall
-  DtSmall
-  DtMedium
-  DtLarge
-  DtVeryLarge

In file entitytypes.h:

enum DtLifeFormDomain

-
-  DtLifeFormDomainOther
 -  DtLifeFormDomainLand
 -  DtLifeFormDomainAir
 -  DtLifeFormDomainSurface
 -  DtLifeFormDomainSubsurface
 -  DtLifeFormDomainSpace
-

Documentation

-  DtLifeFormDomainOther
-  DtLifeFormDomainLand
-  DtLifeFormDomainAir
-  DtLifeFormDomainSurface
-  DtLifeFormDomainSubsurface
-  DtLifeFormDomainSpace

In file entitytypes.h:

enum DtPlatformLandUSSelfPropelledArtillery

 DtM109

Documentation

 DtM109

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtPlatformLandUSArmoredFightingVehicle

-
-  DtM2
 -  DtM3
 -  DtM113
-

Documentation

-  DtM2
-  DtM3
-  DtM113

In file entitytypes.h:

enum DtPlatformLandUSTank

US

 DtM1

Documentation

US

 DtM1

In file entitytypes.h:

enum [DtPlatformLandUSArmoredUtilityVehicle](#)

 DtM88

Documentation

 DtM88

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtPlatformAirUSSpecialOperations

 DtMH60

 DtMH53

Documentation

 DtMH60

 DtMH53

In file entitytypes.h:

enum DtPlatformAirUSSRAttackHelicopter

 DtMI28

 DtMI24

Documentation

 DtMI28

 DtMI24

In file procontrol.h:

extern DtMaximumSocketNumber

the following is used as the first argument to the PC version of select

Documentation

the following is used as the first argument to the PC version of select. It cannot take a 0 like unix versions.
I should be able to get this number programatically.

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtPlatformAirUSSRFighter

 DtMiG29

 DtMiG23

Documentation

 DtMiG29

 DtMiG23

In file entitytypes.h:

enum DtMunitionUSAntiSubmarineGuided

 DtMk48AntiSubmarine

Documentation

 DtMk48AntiSubmarine

In file entitytypes.h:

enum DtMunitionAntiAir

-
-  DtMunitionGuided
 -  DtMunitionUnguided
-

Documentation

-  DtMunitionGuided
-  DtMunitionUnguided

In file entitytypes.h:

enum DtPlatformAirUSObservationHelicopter

 DtOH58

Documentation

 DtOH58

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum DtObjectKind

Environmental process PDU enums

-  DtOtherObjectKind
 -  DtObstacle
 -  DtPreparedPosition
 -  DtObjectCulturalFeature
 -  DtPassageway
 -  DtTacticalSmoke
 -  DtObstacleMarker
 -  DtObstacleBreach
-

Documentation

Environmental process PDU enums

-  DtOtherObjectKind
-  DtObstacle
-  DtPreparedPosition
-  DtObjectCulturalFeature
-  DtPassageway
-  DtTacticalSmoke
-  DtObstacleMarker
-  DtObstacleBreach

In file entitytypes.h:

enum DtPlatformDomain

-
-  DtPlatformDomainOther
 -  DtPlatformDomainLand
 -  DtPlatformDomainAir
 -  DtPlatformDomainSurface
 -  DtPlatformDomainSubsurface
 -  DtPlatformDomainSpace
-

Documentation

-  DtPlatformDomainOther
-  DtPlatformDomainLand
-  DtPlatformDomainAir
-  DtPlatformDomainSurface
-  DtPlatformDomainSubsurface
-  DtPlatformDomainSpace

In file disenums.h:

enum DtRemoveObjectResult

 DtRemoveObjectResultOther

Documentation

 DtRemoveObjectResultOther

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtPlatformLandUSSRSelfPropelledArtillery

 DtSS1ScudMissileLauncher

 DtZSU23_4

 DtSA6Launcher

 DtSAMSite

 DtSA10FlapLidRadar

 DtSA8LauncherRadar

Documentation

 DtSS1ScudMissileLauncher

 DtZSU23_4

 DtSA6Launcher

 DtSAMSite

 DtSA10FlapLidRadar

 DtSA8LauncherRadar

In file entitytypes.h:

enum DtPlatformSubsurfaceUSStrategicMissileSubmarine

 DtSBBNOhio

Documentation

 DtSBBNOhio

In file entitytypes.h:

enum DtPlatformAirUSSRAttack

 DtSU25

Documentation

 DtSU25

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtAH1WSpecific

 DtSpecificAH1W

Documentation

 DtSpecificAH1W

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtCV59Specific

 DtSpecificCV59

Documentation

 DtSpecificCV59

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtFFG7Specific

 DtSpecificFFG7

Documentation

 DtSpecificFFG7

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtSA10Specific

 DtSpecificSA10

Documentation

 DtSpecificSA10

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtSA8Specific

 DtSpecificSA8

Documentation

 DtSpecificSA8

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtSupplyCategories

-  DtSupplyOther
 -  DtSupplyFuels
 -  DtSupplyOils
 -  DtSupplyLubricants
 -  DtSupplyFood
 -  DtSupplySpareParts
 -  DtSupplyPersonnel
-

Documentation

-  DtSupplyOther
-  DtSupplyFuels
-  DtSupplyOils
-  DtSupplyLubricants
-  DtSupplyFood
-  DtSupplySpareParts
-  DtSupplyPersonnel

In file entitytypes.h:

enum DtPlatformLandUSSRTank

USSR

-  DtT72
 -  DtT64
-

Documentation

USSR

-  DtT72
-  DtT64

In file entitytypes.h:

enum DtPlatformAirUSSRBomber

 DtTU26

 DtTU95

Documentation

 DtTU26

 DtTU95

In file entitytypes.h:

enum

DtPlatformSurfaceUSAmphibiousAssaultShip

 **DtTarawaClass**

Documentation

 **DtTarawaClass**

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtAegisCruiserSpecific

-
-  DtTiconderoga
 -  DtYorktown
 -  DtVincennes
 -  DtValleyForge
 -  DtThomasSGates
-

Documentation

-  DtTiconderoga
-  DtYorktown
-  DtVincennes
-  DtValleyForge
-  DtThomasSGates

In file entitytypes.h:

enum [DtPlatformSurfaceUSGuidedMissileCruiser](#)

 **DtTiconderogaClass**

Documentation

 **DtTiconderogaClass**

[Alphabetic index](#) [Hierarchy of classes](#)

In file entitytypes.h:

enum DtPlatformAirUSUnmannedAirVehicle

 DtUAV

Documentation

 DtUAV

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

enum

DtUnderwaterAcousticHullMountedMaskerStatus

 DtUnderwaterAcousticHullMountedMaskerOff

 DtUnderwaterAcousticHullMountedMaskerOn

Documentation

 DtUnderwaterAcousticHullMountedMaskerOff

 DtUnderwaterAcousticHullMountedMaskerOn

In file utmcoord.h:

const DtDegMinSec DtUtmLatRefZone10

The standard UTM reference for a zone is the point along the zone's central meridian at the equator

Documentation

The standard UTM reference for a zone is the point along the zone's central meridian at the equator. Standard UTM positions also use the standard false easting and false easting (see DtUtmInit)

This is the standard UTM reference position for UTM zone 10

In file geodcoord.h:

const DtMapDatum [DtWGS84](#)

Documentation

...and Mapping Datums

[Alphabetic index](#) [Hierarchy of classes](#)

In file hoststructs.h:

typedef struct DtSimulationAddress DtU16 [DtSite](#)

for backwards compatibility

 DtObjectIsUnknown

 DtObjectIsDis

 DtObjectIsHla

Documentation

for backwards compatibility

 DtObjectIsUnknown

 DtObjectIsDis

 DtObjectIsHla

In file libmatrix.h:

enum generic_mm_multflag

Generic DcmDcm Multiply routine

-  TRANSPOSE_NONE
 -  TRANSPOSE_M1
 -  TRANSPOSE_M2
 -  TRANSPOSE_M1_AND_M2
-

Documentation

Generic DcmDcm Multiply routine

-  TRANSPOSE_NONE
-  TRANSPOSE_M1
-  TRANSPOSE_M2
-  TRANSPOSE_M1_AND_M2

In file vlconfig.h:

#define DT_NEED_DLL_DECLARATIONS

If building a DLL MachineTypes

Documentation

If building a DLL MachineTypes.h turn on __declspec(...)

[Alphabetic index](#) [Hierarchy of classes](#)

In file pdufactory.h:

#define DtADD_PDU_CREATOR(x)

DtADD_PDU_CREATOR adds a PDU classes static create function to a PDU factory

Documentation

DtADD_PDU_CREATOR adds a PDU classes static create function to a PDU factory. For example, DtADD_PDU_CREATOR(EntityState) produces: addCreator(DtEntityStatePduKind, DtEntityStatePdu::create);

In file vlutil.h:

#define DtATOI(s)

some versions of strtol returns INT_MAX for input larger than 0x7ffffff

Documentation

some versions of strtol returns INT_MAX for input larger than 0x7ffffff. As long as strtoul accepts and properly processes negative input, this macro will work for both unsigned and signed ints (as other strtol do).

In file libmatrix.h:

#define DtAbs(x)

Returns the absolute value of x

Documentation

Returns the absolute value of x.

[Alphabetic index](#) [Hierarchy of classes](#)

In file netsocket.h:

#define DtBindToDestAddr((DtU8)

setting this bit in the flags constructor argument will cause the NetSocket to ignore all traffic (incl

Documentation

setting this bit in the flags constructor argument will cause the NetSocket to ignore all traffic (incl.
broadcast when dest_addr is multicast and point-to-point) besides that to dest_addr.

[Alphabetic index](#) [Hierarchy of classes](#)

In file disenums.h:

#define DtEXTEND_ENUM(enumType, name, val)

This file should contain only enumeration definitions

Documentation

This file should contain only enumeration definitions. This file is processed by a program that generates the files parseEnum.cc and parseEnum.h. Therefore all enums must conform to the format of the ones already in this file. No comments on the same line as a member of the enum, and all enumeration names and member names must start with Dt. -Len

[Alphabetic index](#) [Hierarchy of classes](#)

In file socket.h:

#define DtMaxPacketSize

IP + UDP headers are 20 + 8

Documentation

IP + UDP headers are 20 + 8. Ethernet MTU is 1500 (or 1492 for 802.3 encapsulation), thus PDU limits will typically be 1472 or 1464 FDDI MTU is 4352 (PDU limit 4324) IEEE 802.5 token ring MTU is 4464 (PDU limit of 4436)

In file libmatrix.h:

#define DtNeg(x)

Returns x if x is less than zero

Documentation

Returns x if x is less than zero.

[Alphabetic index](#) [Hierarchy of classes](#)

In file nullfedamb.h:

#define DtNullFederateAmbassador_H_

Adapted from the DMSO RTI 1

Documentation

Adapted from the DMSO RTI 1.3.4 file NullFederateAmbassador.hh. This class provides empty definitions for the pure virtual functions declared as members of RTI::FederateAmbassador.

[Alphabetic index](#) [Hierarchy of classes](#)

In file pduwithdata.h:

#define [DtPduWithData_H_](#)

DMO 2

Documentation

DMO 2.1.4 PDUs here

[Alphabetic index](#) [Hierarchy of classes](#)

In file libmatrix.h:

#define DtPos(x)

Returns x if greater than zero

Documentation

Returns x if greater than zero.

[Alphabetic index](#) [Hierarchy of classes](#)

In file libmatrix.h:

#define DtSgn(x)

Returns the sign of x

Documentation

Returns the sign of x.

[Alphabetic index](#) [Hierarchy of classes](#)

In file libmatrix.h:

#define DtSquare(x)

Returns the square of x

Documentation

Returns the square of x.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vrlfedtime.h:

#define DtVrlFedTime_H_

This file is based on MyFedTime

Documentation

This file is based on MyFedTime.h, which appears in the DMSO RTI MyFedTime example in RTI_HOME/lang/C++/demo/MyFedTime.

[Alphabetic index](#) [Hierarchy of classes](#)

In file vlconfig.h:

#define VLCONFIG_INCLUDED

File - libsrc/md/c-NT

Documentation

File - libsrc/md/c-NT.h (vlConfig.h for win32)

[Alphabetic index](#) [Hierarchy of classes](#)

In file netsocket.h:

#define __NetSocket_h

class NetSocket NetSocket provides several overloaded constructors that allow the user to specify UDP port, device, and destination address in any combination except port and destination address together

Documentation

class NetSocket

NetSocket provides several overloaded constructors that allow the user to specify UDP port, device, and destination address in any combination except port and destination address together.

In addition, each constructor has an optional flags argument, which may be set to DtRead (read only), DtWrite (write only) or both (default, DtRead | DtWrite).

The default destination address is broadcast. The default network interface device is the first appropriate one located by the constructor.

In file vllist.h:

#define [looplist](#)(theData, list)

Macro to loop through the list

Documentation

Macro to loop through the list.

[Alphabetic index](#) [Hierarchy of classes](#)
