



VR-Forces 4.4 Release Notes

This release note contains the following release-specific information for VR-Forces Release 4.4:

Systems Supported.....	2
3D Video Cards Supported.....	2
Disk Space Requirements.....	3
Compiler Compatibility on Windows.....	3
Required Software for Toolkit Users.....	3
MÄK Product Compatibility.....	4
License Manager.....	4
Third Party Library Support.....	4
Using MÄK Products on Virtual Machines.....	5
Network Compatibility.....	5
Backwards Compatibility.....	5
RPR FOM Versions Supported.....	6
New Features and Updates.....	9
Simulation Changes.....	11
Visualization Changes.....	14
API Changes.....	15
Documentation Updates.....	17
Fixed Bugs.....	17

Copyright © 2016 VT MÄK, 150 Cambridge Park Drive, Third Floor, Cambridge, MA 02140 All rights reserved. VR-Exchange™, VR-TheWorld™, VR-Vantage™, DI-Guy™, and DI-Guy Scenario™ are trademarks of VT MÄK. MÄK Technologies®, VR-Forces®, RTIsPy®, B-HAVE®, and VR-Link® are registered trademarks of VT MÄK.

Document ID: VRF-4.4-3-160315

Systems Supported

VR-Forces 4.4 is available for the platforms listed in [Table 1](#). For toolkit users, application code must be built with the indicated compilers in order to link to VR-Forces libraries.

Table 1: Platforms supported

Operating System	Compiler
Windows Vista, Windows 7, Windows 8	Microsoft Visual C++ 10.0 SP 1 (64 bit)
Red Hat Enterprise Linux 6 (64 bit libraries)	default compiler

3D Video Cards Supported

VR-Forces 4.4 supports NVIDIA graphics cards that support OpenGL 3.3 VR-Forces does not support AMD (formerly ATI) graphics hardware.



You must use NVidia driver 340.29 or later.



VR-Forces strives to provide the best 3D scene quality possible and cutting edge visualization effects. To provide this functionality, VR-Forces takes advantage of OpenGL features that may not be supported on older video cards. When this is the case, you can disable these features, allowing VR-Forces to run in 3D modes.

OpenGL 3.3 is the minimum requirement for rendering a 3D scene in VR-Forces. For the best visual quality and highest performance you should use the most recent NVIDIA GeForce card.

Some laptops have an NVIDIA graphics card and an Intel graphics card. You must configure your computer to use the NVIDIA card. To do so:

1. Open the NVIDIA Control Panel (in the System Tray).
2. Select Manage 3D Settings.
3. Select Global Settings.
4. Select Preferred Graphics Processor.
5. Select High-performance NVIDIA Processor.



You should always try to use the latest drivers available for your video board.

Using Kepler Series Quadro Cards

There is a known issue with the Quadro drivers on Kepler series Quadro cards. NVIDIA says its related to the fact that the default profile optimizes for a fixed eyepoint (for applications like CAD). To run on these cards, you must change the global profile for the driver, as follows:

1. Open the NVIDIA Control Panel.
2. Select the Manage 3D Settings page.
3. Select the Global Settings tab.
4. In the list of driver profiles, select Workstation App - Dynamic Streaming.

We are working with NVIDIA to have their drivers recognize our applications and apply this profile automatically. In the meantime, Kepler series Quadro you must make this change manually.

Disk Space Requirements

A full installation of VR-Forces requires approximately nine GB of disk space.

Compiler Compatibility on Windows

MÄK provides versions of product releases that have been compiled with different versions of Microsoft Visual C++ (for example, 10.0, 11.0, and 12.0). When you run MÄK products together on the same computer, for example, VR-Forces and VR-Vantage Stealth, we strongly recommend that you run versions compiled with the same compiler. Mixing products compiled with different versions of the compiler can result in program instability.

Required Software for Toolkit Users

If you plan to build VR-Forces applications using the VR-Forces Toolkit, you must download and install VR-Link 5.2.1. You can download it from:

- ♦ <ftp://ftp.mak.com/out/products/vrlink5.2.1-win-vc10.exe>
- ♦ <ftp://ftp.mak.com/out/products/vrlink5.2.1-win-vc12.exe>
- ♦ <ftp://ftp.mak.com/out/products/vrlink5.2.1-linux-rhe6.tar.gz>

If you plan to build the VR-Forces GUI, you must download and install QT 4.7.4 (VC 10) or QT 4.8.7 (VC 12). You can download it at:

<http://www.mak.com/support/resources/bonus-material>.

Qt is also available as a free download under the LGPL version 2.1 license at qt.digia.com. This version should be satisfactory for most VR-Forces customers. If you need a Qt commercial license, you must purchase the license from Digia.

MÄK Product Compatibility

To build VR-Forces applications, you must link with VR-Link 5.2.1. If you are building for HLA and want to link with the MÄK RTI, use MÄK RTI 3.x or later, although we recommend use of the latest release. If you are building for HLA Evolved, you must use MÄK RTI 4.0.4 or later.

License Manager

VR-Forces 4.4 uses FLEXlm 11.13.0.2.

The License Manager files are not part of the VR-Forces installer. You can download them at:

- ♦ Windows:
<ftp://ftp.mak.com/out/MAKLicenseManager-win-setup.exe> (for 32-bit systems)
<ftp://ftp.mak.com/out/MAKLicenseManager-win64-setup.exe>
- ♦ Linux:
<ftp://ftp.mak.com/out/MAKLicenseManager-linux-setup.tar.gz> (for 32-bit systems)
<ftp://ftp.mak.com/out/MAKLicenseManager-linux64-setup.tar.gz>

A license manager FAQ is available at:

<http://www.mak.com/license-support.html>

Third Party Library Support

VR-Forces 4.4 uses the indicated versions of the following libraries:

- ♦ VR-Link 5.2.1
- ♦ Qt 4.7.4 (VC 10), 4.8.7 (VC 12)
- ♦ OpenSceneGraph 3.1.4
- ♦ Boost 1.46.1
- ♦ FreeGlut 2.4.0
- ♦ GL-Studio 5.0.2
- ♦ DI-Guy 13.2
- ♦ Triton 3.51
- ♦ Silverlining 4.0741
- ♦ SpeedTree 6.3.2
- ♦ osgEarth 3.0.

Using MÄK Products on Virtual Machines

In general, MÄK products run on virtual machines without any problems. The exception to this statement is that applications that use 3D graphics, such as VR-Vantage and the VR-Forces front-end, require a graphics card. Standard graphics cards typically do not work with VMs. To use a 3D enabled environment in a VM, please contact your VM vendor for options. NVidia produces the NVidia GRID card, which is designed to solve this problem. Your VM vendor may have additional information. For more information on NVidia GRID, please see <http://www.nvidia.com/object/virtual-gpus.html>

Network Compatibility

HLA only

VR-Forces 4.4 was built against VR-Link 5.2.1 and is compliant with:

- RPR-FOM 1.0 and a subset of 2.0 (draft 6, 17, and 20).
- MÄK RTI 3.x, 4.x, and other RTIs that meet the HLA 1.3, IEEE 1516 SISO DLC API, or HLA Evolved specifications and are built with the same compilers as VR-Forces.

VR-Forces 4.4 is compatible with applications that use earlier versions of VR-Link if they support versions of the RPR FOM listed.

The VR-Forces 4.4 front-end is not compatible with RTI-NG Pro because it uses a non-standard version of *xerces_c_2_7.dll*.

DIS only

VR-Forces 4.4 supports DIS 4, 5, 6, and 7, and can therefore interoperate with DIS applications of any of these versions.

Backwards Compatibility

The VR-Forces 4.4 APIs are not compatible with VR-Forces 3.x or earlier. It is largely compatible with the 4.3.x API. Small changes may be necessary to migrate from 4.3.

Scenarios created with VR-Forces 3.12 and 4.x are forward compatible.

Support for MÄK Data Logger Recordings

The recordings provided with MÄK Data Logger 5.4 or later use the new FOM naming conventions. These should work seamlessly with the default HLA configurations in VR-Forces 4.4. If you want to connect VR-Forces 4.4 to recordings made with previous versions of the Logger, you must change the connection configuration to use the old FOM naming convention.

RPR FOM Versions Supported

VR-Forces 4.4 has built-in support for versions 1.0 and 2.0 (drafts 6, 17, and 20) of the RPR FOM and RPR FOM 2.0, draft 17 with DI-Guy extensions. It also supports VR-Link's ability to support alternative FOMs through the FOM Mapper. By default, VR-Forces 4.4 uses RPR FOM 2.0.

If you are building an application with the VR-Forces toolkit and you want to specify a version of the RPR FOM through code, pass the version number (for example, 2.0006) to the *DtRprFomMapper* constructor and pass the resulting object to the *DtExerciseConn* constructor. Also, make sure you are using a federation execution name that corresponds to the right FED file. For example:

```
DtExerciseConn conn("VR-Link20006", "MyApp", new DtRprFomMapper(2.0006));
```

In order to support RPR FOM 1.0, we have added the following extensions (which are supported in later versions of the RPR FOM):

- ♦ EnvironmentProcess
- ♦ GriddedData
- ♦ EmbeddedSystem.IFF
- ♦ EmbeddedSystem.IFF.NatoIFF
- ♦ EmbeddedSystem.IFF.NatoIFF.NatoIFFTransponder
- ♦ EmbeddedSystem.IFF.NatoIFF.NatoIFFInterrogator
- ♦ EmbeddedSystem.IFF.SovietIFF
- ♦ EmbeddedSystem.IFF.SovietIFF.SovietIFFTransponder
- ♦ EmbeddedSystem.IFF.SovietIFF.SovietIFFInterrogator
- ♦ EmbeddedSystem.IFF.RRB
- ♦ BaseEntity.AggregateEntity
- ♦ ObjectRoot.BaseEntity.PhysicalEntity.Lifeform.

For both RPR FOM 1.0 and 2.0 VR-Forces 4.4 relies on the LgrControl and View-Control custom MÄK extensions.

New FED File Naming Convention

VR-Forces 4.4 uses the FOM naming conventions introduced in VR-Link 5.2. This section describes these changes.

Historically, VR-Link has named FED files using the convention VR-Link<RPR FOM version>.fed or VR-Link<RPR FOM version>.xml. For example, the FED file for RPR FOM 2, draft 17 was *VR-Link20017-1.fed*. This naming scheme has two problems. First, it is confusing for end-users of applications that use VR-Link (but who are not themselves VR-Link customers) to be using a FOM named that way. Second, the file-name was not versioned, so any updates that we would make to the existing FOMs could cause compatibility problems. In fact for this latter reason, we had refrained from fixing a few minor bugs that we found over the years until now.

Effective with this release, VR-Link introduces a new naming convention: MAK-*<FOM name><FOM revision><FOM version>.fed* (or *.xml*, or *_evolved.xml*, as appropriate). For example, *VR-Link20017-1.fed* has been renamed *MAK-RPR20017-1-1.fed*.

Additionally, given the power of HLA Evolved FOM Modules, starting with RPR 2.0 we are now providing the RPR standard FOM for HLA Evolved, currently named *RPR_FOM_v2.0_1516-2010*. MAK's extensions, are in FOM Modules that our software will automatically add to an exercise. This means that if you have another RPR-based FOM, such as NETN or NASMP, you will have a much easier time using it without needing to modify the FOM.

VR-Link continues to include FED files and XML files using the old convention, so you do not have to change any existing scripts or code.

[Table 1-1](#) lists legacy FED files and how they map to new FED files. Many of the FED files delivered in previous releases do not have updated versions, usually because they are rarely used.

Table 1-1: Old FED file to new FED file mappings

Old FOM	New FOM
HLA 1.3	
DI-Guy-13_0.fed	
DI-Guy.fed	
VR-Link.fed	MAK-RPR1-1-1.fed
VR-Link05.fed	
VR-Link07.fed	
VR-Link08.fed	
VR-Link10.fed	MAK-RPR1-1-1.fed
VR-Link16.fed	
VR-Link20006.fed	
VR-Link20014.fed	
VR-Link20017-1.fed	MAK-RPR20017-1-1.fed
VR-Link20017-6.fed	MAK-RPR20017-6-1.fed
VR-Link20017.fed	
VR-Link20020.fed	MAK-RPR2-1-1.fed
VR-LinkBOML16.fed	
VR-LinkDDM.fed	
VR-LinkMC02.fed	MAK-RPR1-MC02-1-1.fed
VR-LinkTM.fed	MAK-RPR1-TimeManagement-1-1.fed
VRL-20017-1-Link16.fed	

Table 1-1: Old FED file to new FED file mappings

Old FOM	New FOM
VRL-DIGuy20017.fed	
VRL-DIGuy20017DDM.fed	
1516-2000	
DI-Guy-13_0.xml	
DI-Guy.xml	
VR-Link.xml	MAK-RPR1-1-1.xml
VR-Link20014.xml	
VR-Link20017-1.xml	MAK-RPR20017-1-1.xml
VR-Link20017-6.xml	MAK-RPR20017-6-1.xml
VR-Link20020.xml	MAK-RPR2-1-1.xml
VR-LinkBOML16.xml	
VR-LinkDDM.xml	
VR-LinkTM.xml	MAK-RPR1-TimeManagement-1-1.xml
VRL-DIGuy20017.xml	
VRL-DIGuy20017DDM.xml	
HLA Evolved	
DIGuyModule_1300.xml	MAK-DIGuy-1_evolved.xml
DIGuyModule_1302.xml	
RPR_FOM_v2.0_draft20_1516-2010.xml	RPR_FOM_v2.0_1516-2010.xml
VR-Link20014_base_evolved.xml	
VR-Link20014_evolved.xml	
VR-Link20017-1_base_evolved.xml	MAK-RPR20017-Base-1-1_evolved.xml
VR-Link20017-1_base_evolved_with-UpdateRate.xml	
VR-Link20017-1_evolved.xml	MAK-RPR20017-1-1_evolved.xml
VR-Link20017-1_evolved_withUpdateRate.xml	
VR-Link20017-6_base_evolved.xml	MAK-RPR20017-Base-6-1_evolved.xml
VR-Link20017-6_evolved.xml	MAK-RPR20017-6-1_evolved.xml
VR-LinkDDM_base_evolved.xml	
VR-LinkDDM_evolved.xml	
VR-Link_LgrControl_evolved.xml	MAK-LgrControl-1_evolved.xml

Table 1-1: Old FED file to new FED file mappings

Old FOM	New FOM
VR-Link_ViewControl_evolved.xml	
VR-Link_VrExt_evolved.xml	MAK-VRFExt-1_evolved.xml
VR-Link_base_evolved.xml	MAK-RPR1-Base-1-1_evolved.xml
VR-Link_evolved.xml	MAK-RPR1-1-1_evolved.xml
VRL-DIGuy20017DDM_evolved.xml	
VRL-DIGuy20017_evolved.xml	
updateRate.xml	MAK-updateRate-1_evolved.xml
VR-LinkBOML16_module.xml	MAK-BOML16-1_evolved.xml
	MAK-VRFAggregate-1_evolved.xml

New Features and Updates

VR-Forces 4.4 has the following new features and updates.

- You can now specify additional command line arguments for the front-end, back-end, or both in the Launcher. You can add any supported argument.
- Support for the FOM naming conventions introduced in VR-Link 5.2. For details, please see [“New FED File Naming Convention,”](#) on page 1-6
- The list of preconfigured simulation connections listed in the Launcher has changed. DIS 7 is now the default version. Only two HLA configuration are provided: HLA 1.3 RPR FOM 2.0 with MAK extensions and HLA Evolved RPR FOM 2.0 with MAK extensions. The configurations for other MAK products will match these ones as they are updated.

These configurations are not compatible with the configurations in previous releases. VR-Forces still supports the configurations used in previous versions. You can add them if you want to. Copy the files in *./factory/settings/vrfLauncher-legacy* to *./appData/settings/vrfLauncher*.

- Support for universal unique IDs (UUIDs) as a way to identify each simulation object, tactical graphic, and global plan. Legacy scenarios are supported. If you open a legacy scenario and save it, VR-Forces replaces object names with UUIDs in all scenario files. For more information, please see [“UUID Replaces Object Name for Identifying Objects,”](#) on page 1-15.
- The features of the B-HAVE Module for VR-Forces have been integrated into VR-Forces and it is no longer a separate plug-in. It now uses AutoDesk Gameware Navigation software instead of Kynapse. For an introduction to advanced navigation, please see [“Advanced Terrain Navigation,”](#) on page 3-12 in *VR-Forces Users Guide*. Features have been implemented as follows:
 - Create Multiple Entities is on the Create menu [“Creating Multiple Simulation Objects,”](#) on page 16-21 in *VR-Forces Users Guide*.

- The Spawn Pattern Manager is on the Simulation menu Chapter 31, [Generating Traffic](#) in *VR-Forces Users Guide*.
- Generation of PathData, now called navigation data, is managed on a page in the Terrain Settings dialog box Chapter 52, [Generating Navigation Data](#) in *VR-Forces Users Guide*. To generate navigation data you must purchase an additional license.
- You can generate navigation data for entities that have object geometry “[Generating Navigation Data for Entities](#),” on page 52-8 in *VR-Forces Users Guide*.
- The Hide task has been dropped.
- A new type of simulation graphic, pedestrian path, creates a preferred route on navigation areas that humans will automatically use when path planning.
- ♦ Support for importing scenarios. This feature replaces the Scenario Merge tool.
- ♦ There has been a significant change in terminology. The term entity is now used to refer only to individual entities. The term unit is used to refer to all aggregates. The term simulation object refers to entities and units collectively. This change is reflected in documentation and various GUI menus and dialog boxes. The Entity Editor has been renamed the Simulation Object Editor. Class names and parameter names have not been changed.
- ♦ Changes to command line arguments. The following command line arguments have been removed from vrfSim:
 - --dynamicFeaturesEnabled
 - --frameRateStatisticsLogFile
 - --disableParallelProcessing

The --highestPriority command-line option has been added to vrfGui.

The following command line arguments have been added to vrfSim:

- --disableParallelTick
- --waitQueue
- --numCallbackThreads
- --disableCallbackQueue
- --nonVrfObjectsUUIDScheme
- --enableChannel
- --startMinimized.
- --logFileName
- --fileNotifyLevel
- --noRtiCompilerCheck

Simulation Changes

- ♦ Joystick control has been completely rewritten. You can now map joystick and keyboard keys to movement and weapon control. Configuration is done in the front-end. You do not have to install the joystick on the computer running the back-end any more. All joystick control is from the front-end. For details, please see “[Using a Joystick to Control Entities](#),” on page 19-23 in *VR-Forces Users Guide*.

i

Red Hat 6 does not support joysticks by default, but it can be enabled in the operating system. For details, go to this web site:
<https://access.redhat.com/solutions/64114>

- ♦ Support for automatic embarkation. When you create an entity, you can automatically embark it by placing it on a parent entity. For details, please see “[Embarking Simulation Objects Using Automatic Embarkation](#),” on page 19-11 in *VR-Forces Users Guide*.
- ♦ Support for simulation object groups (SOGs). SOGs allow you to create a configuration of simulation objects and tactical graphics that can be added to scenarios like individual simulation objects. They are not tied to a terrain location, so they are available to any scenario. For details, please see Section 16.10, “[Simulation Object Groups](#)” and Section 63.8, “[Creating a Simulation Object Group](#),” in *VR-Forces Users Guide*.
- ♦ Task and Set menus for global plans. The principal benefit of this is that you can add Wait statements to a global plan and you can write scripted tasks and reactive tasks for execution by global plans. For details, please see “[Creating Global Plans](#),” on page 30-28 in *VR-Forces Users Guide*. Global plans can be launched from the Quick Launch toolbar and from other plans.
- ♦ As part of the updates to global plans, the global plan file has been removed. Global plans are now saved in the plan file (.pln). If you save a scenario created with a previous release of VR-Forces, the plan file is updated and the global plan file is removed.
- ♦ The Quick Launch toolbar allows you to start Events and run global plans by clicking a toolbar icon. For details, please see “[Starting a Scenario Event](#),” on page 8-12 and “[Controlling When Global Plans Run \(Quick Launch\)](#),” on page 30-30 in *VR-Forces Users Guide*.
- ♦ Scenario events have a new end mode – Permanent. A permanent event does not end. For details, please see “[Creating a Scenario Event](#),” on page 8-3 and “[Ending a Scenario Event](#),” on page 8-14 in *VR-Forces Users Guide*.

- ♦ The Simulation Object Editor (formerly called the Entity Editor) can now try to set the support points for an entity based on its model. Bounding boxes will be more accurate. You can still set these manually. The option to set support points from the bounding volume has been removed. For details, please see Section 63.2.2, ["Editing a Simulation Object's Bounding Volume \(Size\)"](#) and Section 63.2.4, ["Specifying Support Points,"](#) in *VR-Forces Users Guide*.
- ♦ The Simulation Object Editor can display the switch node states for entity models that have them. For details, please see ["Displaying Details about the Visual Model,"](#) on page 63-24 in *VR-Forces Users Guide*.
- ♦ UAVs support gimbaled sensors. For details, please see ["Displaying Sensor Views \(3D Only\),"](#) on page 45-10.
- ♦ If an entity has object geometry and entities are embarked on an external surface, such as humans on a carrier deck or the deck of a destroyer, they are included in line-of-sight calculations and can fire at opposing entities and be targeted by them. Embarked entities that are inside a host entity are not included in line-of-sight calculations.
- ♦ The first experience and embarkexample example scenarios have been updated.

Tactical Graphics Changes

- ♦ Support for extruded (volumetric) lines (corridors), areas, boxes, circles, and ellipses. Extruded tactical graphics can specify a base altitude and height. Lines have a width. They are particularly useful for air corridors and airspace management. For details, please see Section 34.1.4, ["Creating Volumetric Tactical Graphics,"](#) in *VR-Forces Users Guide*.
- ♦ A new type of obstacle, linear obstacle, has been added. It works like an areal obstacle, except it is drawn as a line. It is on the Hazards/Obstacles Palette. Supported for entity-level simulation only.
- ♦ The obstacle control object is now on the Hazards/Obstacles Palette instead of on the Tactical Graphics Palette for entity-level scenarios.
- ♦ Tactical graphics can be active and inactive. You can change their activation state based on simulation time or time of day. For details, please see Section 29.2, ["Activation,"](#) [Tactical Graphic Active](#), and Section 34.5, ["Active and Inactive Tactical Graphics,"](#) in *VR-Forces Users Guide*.

Terrain Changes

- ♦ Support for the CDB terrain database format. For details, please see Section 25.8, ["Loading CDB Databases,"](#) in *VR-Vantage Users Guide*.
- ♦ Support for procedurally generated terrain, using either osgEarth or Blueberry 3D. This release includes several example terrains that demonstrate this feature. VR-Forces also supports generation of airports runways, taxiways, signage, and markings.

- ♦ The earth file format now supports inclusion of XML formatted files. For details, please see Section 51.2.1, “[Including External Files into Earth Files](#),” in *VR-Forces Users Guide*.
- ♦ The earth file attribute prefix has changed from `mak_vrf_` to `vrfsim:`. For example, `mak_vrf_layer` is now `vrfsim:layer`.
- ♦ Exporting terrains to GDB is no longer supported.
- ♦ If a model uses the `feature_geom` driver, VR-Forces can now process features differently for the front-end and back-end if the `vrfsim:enabled` attribute is set to true. For more information, please see “[Specifying Different Feature Names for the Front-End and Back-end](#),” on page 54-2 in *VR-Forces Users Guide*.
- ♦ The Nellis, Berkeley, HarrisAfghanVillage, and TerraSim_SampleUrban terrains have been removed.

Task and Set Changes

- ♦ The move to location and move to waypoint tasks have been renamed, as follows:
 - Move to Location (Direct) and Move to Waypoint (Direct) are now called Move to Location and Move to Waypoint. If a terrain has navigation data, and a ground entity given one of these tasks has to pass through a navigation area to reach its destination, VR-Forces uses the navigation data to plan movement within the navigation area.
 - Move to Location (Use Roads) and Move to Waypoint (Use Roads) have been removed from the Task menu. The underlying C++ task is still supported, since it is used by the plan along roads tasks.
 - Move to Location (Plan Path) and Move to Waypoint (Plan Path) for ground entities have been renamed Move to Location (Plan Along Roads) and Move to Waypoint (Plan Along Roads). These tasks work identically to the tasks as previously named.
 - Move to Location (Plan Path) and Move to Waypoint (Plan Path) for surface and subsurface entities continues to be available with the same name.

These changes should not affect scenarios that used the tasks with the old names.

- ♦ The behavior of human entities moving along a route has changed slightly. If they are moving outside of a navigation area and you move the entity away from the route or change the route, the entity plans a path to the route vertex it is moving to rather than moving to the nearest point on the route and then proceeding to the target vertex.
- ♦ Support for fire suppression. For details, please see “[Suppressive Fire Tasks](#),” on page 25-53 in *VR-Forces Users Guide*.
- ♦ New Aiming set data request has been added. The Weapon State set data request has been removed.
- ♦ The Intercept and Destroy task is no longer provided.

- ♦ The Heading set data request and Turn to Heading task now support turning relative to the current position in addition to setting an absolute heading. For details, please see Section 25.63, “[Turn to Heading](#),” and Section 29.30, “[Heading](#),” in *VR-Forces Users Guide*

Visualization Changes

- ♦ Simulation of camera jitter. This is an observer-specific setting. For details, please see Section 43.9, “[Simulating Camera Jitter](#),” in *VR-Force Users Guide*.
- ♦ Sensor contact lines display an arrow between a sensing simulation object and the objects that it has detected. For details, please see “[Displaying Sensor Contact Lines](#),” on page 36-16 in *VR-Forces Users Guide*.
- ♦ Visualization of tasks. VR-Forces can display graphics that help you understand the task an entity is executing. For example, it can display a line showing a Move to Location task. For details, please see “[Visualizing Tasks](#),” on page 36-13.
- ♦ Display of tracers for tracer munitions. This is a 3D-only, observer-specific setting. New Tracer Use set data request to enable use of tracer munitions for weapons that support them. For details, please see “[Tracer Use](#),” on page 29-46 in *VR-Forces Users Guide*.
- ♦ Support for Mimic Track and Tether Track attach modes on a location. For details, please see Section 15.2.5, “[Mimic Location Track Mode](#)” and Section 15.2.9, “[Tether Location Track Mode](#)” in *VR-Vantage Users Guide*.
- ♦ Support for Fresnel lighting for scene reflections.
- ♦ Upgrade to DI-Guy 13.2. Many new characters.
- ♦ Many improvements to performance, including faster and smoother SpeedTree loading and an improved ability to handle long frames.
- ♦ Vegetation can now show light cast from street lights and other light sources.
- ♦ Many new and upgraded entity models, including:
 - Upgraded F18 and Mirage.
 - KC 135 tanker, with boom.
 - Patriot battery models.
 - Coast Guard patrol boat.
 - Passenger ferry
 - MQ-8 Firescout
 - Cruise ship.
 - Gimballed sensor on MQ-9 Reaper.
 - Tugboat.
 - V-280 Valor.
- ♦ Support for touch screens. VR-Forces supports touch screens to the extent that your operating system and monitor support them. There is no VR-Forces-specific configuration required.

Changes to the Graphical User Interface

- The Attach menu option is now Observer Attach.
- The Selection Groups Panel has been integrated into the Objects List Panel as a new tab. Importing and exporting selection groups is no longer supported.
- The Observer Settings toolbar has been reduced to show the settings that we think will be most useful for VR-Forces users. All of the observer settings continue to be available on the Observer menu and you can edit the toolbar if you want to, as described in Section 74.5, “[Configuring Toolbars](#),” in *VR-Forces Users Guide*.
- The Scenario Startup screen has been updated. You can now select previously opened scenarios and can select the terrain and simulation model set for new scenarios. It also includes a description of the selected terrain. For details, please see Section 4.1.1, “[Starting VR-Forces from the VR-Forces Launcher](#),” in *VR-Forces Users Guide*.

API Changes

UUID Replaces Object Name for Identifying Objects

VR-Forces 4.4 now uses a VR-Forces UUID as a unique identifier. Simulation objects and tactical graphics can now have the same names, but can still be identified as unique objects. This makes collaborative creation of scenarios and merging scenarios an easy process. As part of this transition, the Scenario Merge tool has been dropped. You can now merge scenarios simply by importing them into an open scenario in the VR-Forces front-end.

Loading Legacy Scenarios

When you load or import a scenario created in VR-Forces 4.3.x or earlier, all simulation objects are given new unique IDs. These unique IDs replace simulation object names in all locations where they are used, such as plans and the order of battle file.

When a legacy scenario is loaded (or imported) the DtUUID, DtRwUUID, and DtUUIDMarkingTextResolution managers convert object names to new UUIDs. A marking text to UUID mapping is created in the marking text, and all the marking text entries are then remapped to their new UUIDs. This will occur in any place that an object name is used as an identifier (plans, tasks, sets, and so on).

This process is transparent to the end user. Simulation objects are still be referenced by object name and all operations still work on object names from the users perspective.

API Changes

Wherever marking text used to be used to address objects, the object's UUID will be used. If you are sending messages to objects and are using the `objectName()` in a message, for example:

```
msg.setObjectName(obj->objectName());
```

you will now use the object's UUID:

```
msg.setUUID(obj->uuid());
```



It is still possible to use the marking text of the object, for example:

```
msg.setUUID(obj->markingText());
```

However, using old style code is not recommended. If you do not use the UUID, scenarios cannot use duplicate simulation object names and end users will lose the benefit of this change. For example, importing scenarios will not be guaranteed to work.

If you have written controllers, process state repositories, tasks, sets, and so on, that use `DtRwString` as an identifier for an object to apply an operation to, all you need to do to migrate your code is change `DtRwString` to `DtRwUUID`.

If you make this change, when you load a scenario, VR-Forces remaps what used to be the marking text to the new UUID representation. In the debugger you will see that the UUID also has a text pointer to the object (via marking text) that this represents.

If you have created methods that take an object name, such as:

```
void MyClass::setObjectName(const DtString&);
```

when you change your member to a `DtRwUUID` you should also change this prototype:

```
void MyClass::setUUID(const DtUUID&);
```

When looking up *DtVrfObjects* in the object manager you can now use:

```
lookupVrfObjectByUUID
```



If you pass in marking text, this call also tries to look up the object by marking text. However, this is not recommended, because it will not allow the controller to work with similarly named objects.

Changes to VR-Link's DtVector

In VR-Link 5.2, *DtVector* has been split into *DtVector32* and *DtVector64*. (*DtVector64* is aliased to *DtVector*.) All functions that take velocity, acceleration, and embedded position information have been converted into *DtVector32*. This was done because DIS and the RPR FOM use 32-bit values for those fields and customers using our standard *DtVector* were seeing small rounding issues due to conversion to and from 32-bits when the data was passed over the network. Unfortunately, this also affects a large number of classes in VR-Link.

This change is unlikely to require changes to VR-Forces code. However, it is pointed out here just in case. For more information, please see *VR-Link 5.2 Release Notes* and class documentation.

Documentation Updates

Documentation has been updated for this release. *VR-Forces Users Guide*, *VR-Forces Scenario Management Guide*, and *VR-Forces Configuration Guide* have been combined into one manual – *VR-Forces Users Guide*. We believe this change will simplify searching for information. *VR-Forces Documentation Center* is no longer provided, since *VR-Forces Users Guide* effectively provides a complete table of contents and index to VR-Forces documentation.

i

The video tutorials for terrain composition and model creation distributed with VR-Forces were created using VR-Vantage. Although there are obvious differences in the overall GUI, the menu options and procedures for these activities are the same in VR-Forces as in VR-Vantage. The Breaching Scenario and Creating Conditional Expression videos were created using VR-Forces 4.2. Again, although there are some changes in the GUI, these procedures have not changed.

VR-Forces Users Guide incorrectly states that Collada terrains are supported. They are not.

Fixed Bugs

The following problems identified in previous releases have been fixed:

- VR-Forces allowed you to exit without warning you to save the scenario after you changed the environment. 56837
- On Linux, the Entity Palette would not stay open when the window was maximized. 54305
- On Linux, a variety of GUI bugs related to resizing windows and dialog boxes were resolved. 54500
- On Linux, example makefiles did not work. 54706

- The GUI crashed on exit if an aggregate was embarked. 54784
- Some incomplete GUI API documentation was improved. 56894
- The Show Other Stealth setting did not persist. 55891
- On Linux, the Help menu link did not work. 55737
- The Location3D:isInsidePolygon Lua function did not always work as expected. 56898
- Double clicking on the title bar of pinned dock widgets caused unexpected behavior. 56921
- The move-along controller for ships was not handling the case where there are two (or more) points close together at the end of the route. 55623
- It was not possible to turn off vertex labels for unpublished tactical graphics. 56887
- The OPD Editor did not save changes to the structure of sensor geometry. 54947
- On Linux, the feature_custom osgEarth driver did not load. 54803
- In the scripted task editor, deleted folders were not properly deleted. 55654
- The ballistic gun controller did not preload until it found a target. 56015
- Improved default naming scheme for scenarios. 54918
- In Lua scripts, DtAnimationModel did not execute the last frame once before DtAnimationModel::isFinished() returned true. 55754
- Cruise missile surface skimming did not work correctly. 56667
- The graph search algorithm for finding initial edge did not always work. 54814
- Instrument panels were not supported. 56676
- Color by force did not work for spot report labels. 56682
- Documentation has been updated for many examples.
- Dragon Warrior UAV exploded when placed on the ground. 54860
- The widget used to create a route or an area called the coordinate display function convertPositionToStringList with the altitude unit instead of the distance unit. 56655
- If you attached a weapon system to a remote entity, its resources were not provided. 56206
- Improved editing formations in the Simulation Object Editor. 56183
- If you changed the main window from embedded to another window type, the overlay bar disappeared. 56184
- If a tactical graphic had its color changed to white and the scenario was saved, when the scenario opened again, the graphic was no longer white. 55629
- If two plan windows were open and one of the plans was abandoned, the other was also marked as abandoned. 55080
- Improved how aircraft embarked on a carrier. 55406
- The checkpoint timer affected the observer view. It as changed to a toolbar. 55736
- After creating a new model in the Simulation Object Editor, the new model was not visible. 55748

- ♦ Missiles were not given an initial velocity when launched. 55818
- ♦ Aggregates abandoned their plan when switching model resolution. 55685
- ♦ If you set some appearance of a human, the Compliance always got set to Detained. 55430
- ♦ DtFixedWingMoveAlongController was not using the atGoal function as it should have been - it was just taking current speed and comparing it to distance, effectively looking ahead 1s instead of using at-time. 55419
- ♦ Fire-for-Effect tasks did not work. 55392
- ♦ MSDL did not correctly save or restore environmental objects. 54968
- ♦ Load balancing did not get applied after a rewind. 55030
- ♦ The embarked on list in the Information dialog box was sometimes incomplete. 54785
- ♦ The OPD Editor could not load an SMS after one was already loaded. 54940
- ♦ In the Simulation Object Editor, the search bar did not work with a reloaded SMS. 54948
- ♦ When loading an aggregate-level scenario that has a disaggregated aggregate with saved state data this state data got lost. 55164
- ♦ The Patrol Route task did not save task status correctly. 55157
- ♦ If you tried to change environment conditions without loading a scenario, the back-end crashed. 55130
- ♦ DtComponentSystem::getAllAttributeNames() did not go through the list of attribute "set" functions and report the attribute names. 55035

