



VR-Forces 4.3 Release Notes

This release note contains the following release-specific information for VR-Forces Release 4.3:

Systems Supported	2
3D Video Cards Supported	2
Disk Space Requirements	2
Compiler Compatibility on Windows	2
MÄK Product Compatibility	3
Qt Release Compatibility	3
FLEXlm Support	3
Third Party Library Support	3
Running the 32-bit Version of VR-Forces	4
Using MÄK Products on Virtual Machines	4
Network Compatibility	4
Backwards Compatibility	4
RPR FOM Versions Supported	5
New Features and Updates	6
Simulation Features	6
Visualization (Front-End) Features	9
API Changes	11
Documentation Updates	12
Fixed Bugs	14

Copyright © 2015 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-4.3-3-150311

Systems Supported

VR-Forces 4.3 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 (32 bit and 64 bit)
Red Hat Enterprise Linux 6 (64 bit libraries)	default compiler

3D Video Cards Supported

VR-Forces 4.3 supports NVIDIA graphics cards that support OpenGL 3.3 VR-Forces does not support AMD (formerly ATI) graphics hardware.

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.

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 and 11.0). 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.

MÄK Product Compatibility

To build VR-Forces applications, you must link with VR-Link 5.1.2. 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.

Qt Release Compatibility

If you want to do development using the VR-Forces GUI API, you need to use Qt, a cross-platform GUI toolkit from Digia. The VR-Forces GUI was built with Qt release 4.7.4. Qt is 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.

FLEXlm Support

VR-Forces 4.3 uses FLEXlm 11.11. If you have a previous version of FLEXlm installed, you can install the MÄK License Manager and run FLEXlm from that directory, or you can copy the files installed by MÄK License Manager to your previous FLEXlm directory on your license server. You do not need to change license files (unless your maintenance agreement has expired.)

Third Party Library Support

VR-Forces 4.3 uses the indicated versions of the following libraries:

- ♦ VR-Link 5.1.2
- ♦ Qt 4.7.4
- ♦ OpenSceneGraph 3.1.4
- ♦ Boost 1.46.1
- ♦ FreeGlut 2.4.0
- ♦ GL-Studio 4.2.3
- ♦ DI-Guy 13.0.1
- ♦ Triton 3.10
- ♦ Silverlining 3.040
- ♦ SpeedTree 6.3.2
- ♦ osgEarth 2.6_rev150902.

MÄK will provide modified source code for OSG and osgEarth upon request, as per the OSGPL license. Please contact support@mak.com for download links.

Running the 32-bit Version of VR-Forces

While VR-Forces is supported on 32-bit operating systems and processors, it is not recommended. Modern applications like VR-Forces take advantage of large pools of memory to load complicated and beautiful content. Using a 32-bit operating system or machine, will mean some content may not be available, and VR-Forces may perform worse than with a 64-bit environment.

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 4.3 was built against VR-Link 5.1.2 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 4.3 is compatible with applications that use earlier versions of VR-Link if they support versions of the RPR FOM listed.

The VR-Forces 4.3 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 4.3 supports DIS 4, 5, 6, and 7, and can therefore interoperate with DIS applications of any of these versions.

Backwards Compatibility

The VR-Forces 4.3 APIs are not compatible with VR-Forces 3.x or earlier. It is largely compatible with the 4.2.x API. Small changes may be necessary to migrate from 4.2.

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

RPR FOM Versions Supported

VR-Forces 4.3 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 4.3 uses RPR FOM 1.0.

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 4.3 relies on the LgrControl and View-Control custom MÄK extensions.

New Features and Updates

VR-Forces 4.3 has the following new features and updates.

Simulation Features

- ♦ New distinction between entity-level modeling (the modeling approach used in VR-Forces 4.2.x and earlier) and aggregate-level modeling. To support these two approaches, we provide *EntityLevel.sms* (the former *default.sms*) and *AggregateLevel.sms*. For introductory conceptual details, please see Section 1.1.1, “Entity-Level and Aggregate-Level Simulation,” and Section 3.7, “Aggregate Modeling” in *VR-Forces Users Guide*.
- ♦ Aggregate warfare model, which supports many new tasks and set data requests. For details, please see Chapter 5, *Simulating Aggregate-Level Entities*, in *VR-Forces Scenario Management Guide*.
- ♦ Refactoring of SMSs and the ability to include one SMS in another SMS. SMSs from VR-Forces 4.2 should still work without being upgraded to the new format. For details, please see Section 5.3.1, “Including Simulation Model Sets in other Simulation Model Sets” and its following sections in *VR-Forces Configuration Guide*.
- ♦ The Entity Editor has been updated to support configuration of entities for entity-level modeling and aggregate-level modeling. For details, please see *VR-Forces Configuration Guide*.
- ♦ Scenario events. Scenario events allow you to inject information into a scenario for scenario participants to respond to. For details, please see Chapter 12, *Scenario Events*, in *VR-Forces Scenario Management Guide*.
- ♦ Localized weather areas. For details, please see Section 15.5, “Creating Local Weather Zones,” in *VR-Forces Scenario Management Guide*.
- ♦ Support for VR-Link 5.1.2 and, therefore, for DIS 7.
- ♦ In Stealth observer mode, if you are creating multiple human characters of a type that randomizes the appearance of the character that gets created, such as Civilian Male, after each character you create, the 3D model attached to the cursor changes to show you the appearance of the next character that will be created. The appearance is synchronized with the front-end and back-end.
- ♦ The names of DI-Guy animations are more readable. For example `angry_chant_1_arm` is now called Chant with One Arm. This name change should not affect legacy scenarios.
- ♦ Ability to specify use of Start/Resume and Stop/Freeze messages when running and pausing scenarios instead of VR-Forces remote control messages.
- ♦ Naval guns can now lead targets. This is an automatic behavior.
- ♦ Improved support for IFF.

Terrain

- ♦ The Boundary Tool has been updated. The executable has been renamed. For details, please see Section 15.6, “[Using Cut-in Sites for High Resolution Insets](#),” in *VR-Forces Configuration Guide*.
- ♦ The TDB Tool is no longer part of the VR-Forces installer. It is available as a separate download. Tutorials that used the TDB Tool have been removed.
- ♦ Support for dynamic features. Dynamic features simulated objects (entities) that get inserted into a feature layer and which the simulation engine then treats like any other feature for the purpose of path planning, obstacle avoidance, and so on. Example of dynamic features could include cultural features such as bridges, and combat engineering objects. For details, please see Section H.9, “[Dynamic Features](#),” in *VR-Forces Configuration Guide*.
- ♦ New and updated terrains to demonstrate streaming features, such as S-57 data and point buildings. (The terrain *VR-TheWorld Online - Mak Earth - with point feature buildings.mtf* is considered to be an experimental terrain. Experimental terrains may cause degraded performance or may have visual problems.) *VR-TheWorld Online - MAK Earth - No Streaming SpeedTrees.mtf* and *MaklandNoSpeedtrees.mtf* have been removed.

Command Line Options and Configuration Parameters

- ♦ New command-line options:
 - `-I | --batchedRendering`. Improves performance by using instancing for multiple instances of identical models. (Front-end only.)
 - `--pvd`. Starts the front-end with just the 2D display modes. (Front-end only.)
 - `--stealth`. Starts the front-end with just the 3D display modes. (Front-end only.)
 - `--defaultObjectTimeout`. Sets the DIS object timeout interval. (back-end only.)
 - `--dynamicFeaturesDisabled`. Turns dynamic features on and off. (back-end only)
 - `--setMainThreadToHighPriority`. Sets the VR-Forces thread to the high priority. (back-end only)
 - `--disableParallelPreprocess`. Disables the ability of components that implement the preprocess call to run them in parallel.
 - `--fowPerspective force`. Shows the ground truth and spot reports perspective of the specified force. Currently supports Friendly, Opposing, Neutral, None. Default: None. (Front-end only.)
 - `--fileTransporterReceivePort port`. Port for the file transporter.
 - `-O | --hasNTPSync`. Synchronizes ocean and speedtrees between different applications, such as the VR-Forces front-end and VR-Vantage Stealth. (Front-end only.)

- `-S` | `--suppressHatIntersectOnAttach`. Suppress the periodic HAT intersection when attached. This option improves performance if you are attaching the observer to ground entities, but not air entities. (Front-end only.)
- ♦ `publishSimulationTimeForStealth` has been removed from *vrfSim.mtl*. This behavior happens by default.
- ♦ The *vrfSim.mtl* configuration file has the following additions:
 - `disableParallelPreprocess`. If set to 0, the default, use parallel preprocessing for objects. If set to 1, use serial preprocessing.
 - `setMainThreadToHighPriority`. If set to 1, sets the main thread to `THREAD_PRIORITY_TIME_CRITICAL` in Windows. Using this option on a single core processor will probably cause deadlock. (Windows only)
 - `enableDebugDrawer`. Enables (1) or disables (0) the simulation terrain debug drawer. Default: 1.
 - `dynamicFeaturesDisabled`. Set to 0, the default, to enable dynamic features. Set to 1 to disable them. Dynamic features are features that get integrated into the vector network for path planning.
 - `frameRateStatisticsLogFile` specifies a log file for frame rate statistics.
 - `publishZeroVelocityWhenPaused`. When set to 1, the default, VR-Forces sets entity velocities to 0 when it is paused so that entities do not continue to dead reckon. To publish the current velocity, set to 0.
- ♦ In batch scenarios, the `parameter-filename` parameter, which specified an OPD file has been replaced with `sms-filename`, which specifies an SMS file.

Tasks and Plan Changes

- ♦ Support for sending view control messages from a plan. For details, please see Section 11.6.6, “[Sending View Control Messages](#),” in *VR-Forces Scenario Management Guide*.
- ♦ The Fire-for-Effect on Entity task has changed. Artillery entities will now track a moving target and fire at the dead-reckoned location of the entity rather than its location at the time the task is assigned.
- ♦ New Lua function `getLifeformPosture()`.
- ♦ Many new tasks and set data requests for aggregate-level simulation, including a new category of scripted tasks called background processes.
- ♦ The following new commands for entity-level simulation:
 - Set Superior. Allows you to add entities to an aggregate or remove them from an aggregate.
 - Strafe Ground Target. Support for aircraft with guns, such as the A-10, to strafe ground targets. Also supported through an updated Execute Close Air Support task.
 - Set Jam Targets. Orders an aircraft to jam specific targets or all entities within its jamming beam.

- Attack Aircraft with Guns. Orders an aircraft to attack another aircraft using its guns (as opposed to missiles).

Visualization (Front-End) Features

VR-Vantage 4.3 has the following new features:

- ♦ Attempt to force NVidia card use on laptops. VR-Vantage now exports a global symbol used by Optimus (NVidia's laptop graphics management system) to always use the high-performance graphics card.
- ♦ The location of the title for tactical graphics boxes has changed. Titles for boxes created in previous versions of VR-Forces will keep their old location.
- ♦ Revised Information dialog box. The dialog box now has a summary panel that displays key object information. You can now select multiple objects and display summary information for all of them in one window. For details, please see Section 6.5, “[Displaying Information About an Object](#),” in *VR-Forces Users Guide*.
- ♦ The Terrain Settings Dialog Box, Earth Layers page has been added. It lets you turn visualization of data on and off by feature layer and by observer mode. For details, please see Section 18.7, “[Filtering Earth File Layers](#),” in *VR-Forces Users Guide*.
- ♦ You can now use environment variables for file names anywhere that APP_DIR, DATA_DIR, and USER_DIR are supported, provided the environment variable begins with “VRVDIR_”. Any path that matches that variable will have the variable substituted into whatever file is being written and will be resolved when the file is read.
- ♦ Upgraded to OSG 3.1.4.
- ♦ Upgraded to VR-Link 5.1.2, which includes support for DIS 7.
- ♦ VR-Forces now uses DI-Guy 13.0.
- ♦ VR-Forces now uses SpeedTree 6.3.2.

Lighting, Shadows, and Other Shader Improvements

- ♦ Support for Forward+ lighting. This improvement removes the restriction on the number of lights in the scene.
- ♦ Cascaded, shadow map-based shadows. Shadowing is significantly improved. Entities, props, and terrain can all cast shadows and shadows can be cast on them as well. Shadow configuration is managed on the Display Settings dialog box, Shadow Settings page. This page will probably be simplified prior to final release and therefore is not fully documented. Generic descriptions of most parameters are available on the world wide web. For more information, please see Section 17.8, “[Displaying Shadows \(3D Only\)](#),” in *VR-Forces Users Guide*.

- ♦ Ambient Material Value and Diffuse Material Value. In conjunction with shadows, it is important that the ambient response of modeled materials is compatible. You can now allow VR-Vantage to set the ambient material value and diffuse material value to be used for all objects in a scene and to scale the current ambient value(s) in use, whether ours or the values on the modeled materials. For more information, please see Section 17.6, “[Configuring the Material Ambient and Diffuse Values](#),” in *VR-Forces Users Guide*.
- ♦ Emissive Maps for Day/Night. VR-Vantage now supports Emissive Maps that can be used to control the emissivity of whatever is applied to based on the current ambient light values. These maps use a naming scheme similar to the way our specular and normal maps work – texture names that end with `_EMM` are recognized as Emissive Maps. For more information, please see Section 13.7, “[Using Shader-based Effect Maps](#),” in *VR-Forces Configuration Guide*.
- ♦ Dynamic reloading of shaders. Developers who are updating shaders can reload them while VR-Forces is running. For details, please see Section 13.7.1, “[Reloading Shaders](#),” in *VR-Forces Configuration Guide*.
- ♦ Anti-aliasing is supported when rendering to a texture buffer rather than the screen. This means it is now supported on post-processor effects such as rain splash and depth of field.
- ♦ Ability to set the regeneration threshold for the skybox cube map. For details, please see Section 17.7, “[Configuring the Skybox Cube Map](#),” in *VR-Forces Users Guide*.
- ♦ Ability to change which shaders are visualized for a 3D model.

Marine

- ♦ Improved visualization of maritime effects.
- ♦ Support for S-57 data, as follows:
 - Streaming buoys and beacons. VR-Forces can display buoys, beacons, and navigation lights based on data in shape files or S-57 data. To view buoys, load *VRTW - Massachusetts S-57 buoys.mtf* and zoom into Boston harbor. Turning on model scaling may help you to see them more easily. For more information, please see Section 15.8, “[Streaming Buoys and Beacons](#),” in *VR-Forces Configuration Guide*.
 - Signal sequence generation. If S-57 data for beacons and buoys does not include signal sequence information, VR-Forces tries to generate one. For details, please see Section 15.8.6, “[Generating Signal Sequences](#),” in *VR-Forces Configuration Guide*.
 - Multicolor (alternating) navigational aid lights are now supported.
 - Streaming daymarks for buoys are supported. For details, please see Section 15.8.5, “[Streaming Daymarks](#),” in *VR-Forces Configuration Guide*.
- ♦ The maximum ocean height that can be specified from the GUI has been increased to 5000 m.

- ♦ Tidal stream wakes. Tidal stream wakes are wake effects around buoys and beacons caused by movement of the tides around them. For more information, please see Section 17.8.1, “[Configuring Tidal Stream Wakes](#),” in *VR-Forces Configuration Guide*.
- ♦ Ability to set the ocean height map size, texture, and regeneration threshold. For details, please see Section 13.5.4, “[Configuring the Ocean Height Map](#),” in *VR-Forces Configuration Guide*.
- ♦ Buoyancy model. You can now turn off the buoyancy model for watercraft entities. The parameter `autoBuoyancy` has been added to the watercraft model definition schema. The default (unspecified) is `true`. Add `autoBuoyancy false` to disable the buoyancy model from entities created using that model definition.

Modeling Features

- ♦ Depth of Field. When enabled, the Stealth view is focused only within the focal depth and range. For details, please see Section 8.17, “[Using the Depth of Field Effect](#),” in *VR-Forces Users Guide*.
- ♦ Display of laser designators. For more information, please see Section 8.10, “[Displaying Laser Designators](#),” in *VR-Forces Users Guide*.
- ♦ Particle systems are affected by wind.
- ♦ Support for animated flags and windsocks.
- ♦ New render settings for high quality rotor wash, and sea spray particle effects.

Performance

- ♦ Improved performance and visual quality. In addition to all the specific features listed in this section, VR-Forces 4.3 has significantly improved performance through changes to underlying APIs and rendering approaches.
- ♦ Batched instance rendering (instancing). This can greatly reduce the GPU time to render multiple entities of the same type. Instancing is disabled using the `-I` or `--batchedRendering` command line argument. Default: disabled.
- ♦ Updated performance statistics display and new function profiler. For information about using the function profiler, please see Section 11 in *VR-Vantage Developers Guide*.

API Changes

VR-Forces has the following changes to its APIs. For occasional updates to migration information, please check <http://www.mak.com/support/migration-support.html>.

- ♦ We have renamed a lot of header files for clarity, so that they will better match class names and make things easier to find. This release includes tools to help make this part of the upgrade process easy:

- In the *./compatibility* folder, there is a command line tool you can run to upgrade the header file references in your code. Instructions for how to do this are in *./compatibility/readme.txt*.
- Alternatively, header files with the old names have been added to *./include/compatibility*. They point to the new files. You can add this directory to your project's include paths.
- ♦ In HLA, VR-Forces can change the transport type Data interactions at run time based on the content type of the message. Before sending a message, VR-Forces checks to see if it should be best-effort or reliable. If the current setting does not match what is required for the content type of the message, it makes an RTI service call to update the transport type.
- ♦ *DtVrfMessageInterface* has new methods to allow a plug-in to change transport types as well.
- ♦ The *DtIfServerStatus* message now contains, by back-end, the number of entities (by kind and domain) that are being simulated on that back-end. When a scenario is loaded, the *signal_allScenarioObjectsDiscovered* signal is sent from the *DtVrfScenarioManager*.
- ♦ By default, the following content types are sent reliable. Everything else is best-effort.
 - *DtRemoveFromOrganizationType*
 - *DtSetSubordinateOrderType*
 - *DtAddToOrganizationType*
 - *DtIfObjectIndexMessageType*
 - *DtScriptedTaskResponseMessageType*.

Documentation Updates

Documentation has been updated for this release.



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. Most of the VR-Forces-specific videos were created using VR-Forces 4.2. Again, although there are some changes in the GUI, these procedures have not changed.

The following section is an update and replacement for Section 17.9, “[Adding Wind-based Controls to Models](#),” in *VR-Forces Configuration Guide*.

Adding Wind-based Controls to Models

VR-Vantage can simulate the movement of flags and windsocks based on the current simulated wind speed and direction on terrains. It can also simulate movement of flags and windsocks on models added through connections or programmatically (this is not supported for props or external references). These features require models with appropriate switches and articulations, as well as model-specific or terrain-specific meta files.

A meta file is associated with a terrain or model in one of the following ways:

- ♦ Place it in the same directory as the terrain or model's MEDF file. The meta file must have the same name as the MEDF file, but with a *.meta* extension.
- ♦ Specify it in the `metaFileName` parameter of the model definition used to import the model.

Meta files are user-editable, but require understanding of the annotations and naming conventions in the model file itself.

The meta files are JSON configuration files. They have sections that configure wind control of nodes, as follows:

- ♦ Control switch nodes based on wind speed.
- ♦ Control the heading (Z-Axis) of articulating (degree of freedom) nodes based on wind direction.

Each section has a `Linkage` entry and a `FeatureData` entry. The `Linkage` entry specifies an OSG node that should have the feature specified in the `FeatureData` entry.

Linkages can be specified by node name or WRM Specification comment (for example, `@dis articulated_part 7872`). When a model or terrain file is loaded and a meta file is associated with that file, the model or terrain file is searched for any node or nodes specified in any `Linkage` entries in that meta file and the necessary instances to implement the `FeatureData` indicated in that `Linkage/FeatureData` pair are created.

`FeatureData` sections indicate a `FeatureName` (which is mapped to a creator that will instantiate the appropriate object to manage that feature) and any additional parameters needed to manage the feature. The meta file used for wind features supports two features - `WindSwitchPart` and `WindDirPart`.

WindSwitchPart. The `WindSwitchPart` feature is used to change the state of a switch node based on the current wind speed. The flag models used by MÄK include a switch with three states. The `WindSpeed` parameter is a list of wind speeds (in km/h) and the `WindState` parameter is a list of the switch state values that correspond to the values in the `WindSpeed` parameter. The number of items in each list should be the same (and only contain switch states supported by the node specified in the `Linkage` entry).

WindDirPart. The `WindDirPart` feature is used to modify the heading value (Z-Axis) of an articulation (degree of freedom) node based on wind direction. It has the following parameters:

- ♦ `RotOffsetDeg`. Allows a rotation offset to be applied.

- ♦ **RotDir.** Indicates if the node should be rotated in a clockwise or counterclockwise direction.

For information about adding new meta files or new meta file features, please see section 8.6, Transforming Nodes Programmatically, in *VR-Forces Front-End Developers Guide*.

Fixed Bugs

The following problems identified in previous releases have been fixed:

- ♦ Ballistic missiles in saved scenarios ignored the launch time. 52709
- ♦ The Plan dialog box did not disable the Embedded Entities option for entities that did not have embedded entities. 52715
- ♦ It was not possible to disable laser designator lines and cross hairs. 42849
- ♦ The simulation engine did not process model layers with the simple driver. 49495
- ♦ The Apply Environment Conditions list did not show the selected condition. 51263
- ♦ The Underwater Acoustics PDU offset was not updated. 51316
- ♦ Double-clicking to select an SMS in the New SMS from Existing dialog box did not work. 51343
- ♦ If B-HAVE was installed, but the user did not have a B-HAVE license, the back-end failed without saying what the problem was. 51440
- ♦ The missile launched by the Apache sometimes blocked LOS from the laser designator (since they are both configured to be at the origin) so the missile sometimes did not track its target. 51960
- ♦ VR-Forces crashed if an entity had fewer formation slots than subordinates. 52079
- ♦ The value specified in initial-weapon-state was never applied, because of the order of initialization of the human state repository. 52085
- ♦ The Lua follow task did not work when B-HAVE was installed. 52137
- ♦ The OPD Editor did not write out changes to the damage model. 53817
- ♦ Surface entities kept making small changes to their heading when following a straight route. 51153
- ♦ The entity icon on a track history terrain profile was not in the correct location. 51175
- ♦ When editing embarkation slots in the Entity Editor, the Up position did not update after changes. 52186
- ♦ The Linux version did not run without an RTI in its path. 51575
- ♦ Task and Set dialog boxes for embedded entities showed information for their parent instead of themselves. 52141
- ♦ vrfGui crashed when running against an SMS that had not been upgraded. 52329

- ♦ An entity given a backwards movement task did not return to forward movement. 52512
- ♦ If you opened the OPD Editor from the Entity Editor, it did not save changes. 52708
- ♦ If you tried to move a circle by right-clicking and selecting move, the dialog box did not have an OK button. 52777
- ♦ Collision avoidance did not work properly for rotary-wing entities that were set to the same location. 53824
- ♦ When you added an indirect fire weapon to an entity using the Entity Editor, the "fire-support-capable" flag did not get set. 53828
- ♦ The supported slots setting for embarkation ingress points did not work. 53829
- ♦ The Entity Editor did not save favorites. 53830
- ♦ Entities followed in reverse. 54157
- ♦ DIs got stuck in the read demo scenario. 52167
- ♦ Terrain profile entity tracking caused a crash on Linux. 52227
- ♦ The bhaveRemoteControl example did not build on Linux. 52230
- ♦ Changing the time multiplier affected the maximum speed of surface entities. 52109
- ♦ Setting MAK_NUM_LANES to 1 (instead of 2) in *featureconfig.txt*, resulted in odd driving behavior. 52102
- ♦ If you placed a satellite on the ground, you could not change its altitude. 53688
- ♦ It was not possible to create a satellite at altitudes greater than 200000m. 53712
- ♦ Entities did not report that they were under fire from Munition Target events. 53733
- ♦ The SailToHeading task caused a crash if some parameters were not set. 54066
- ♦ Enabling the publish-transmitter setting in the radio definition in the OPE file did not cause transmitter PDUs/objects to be published. (This setting must now be true in the OPE file and commModelParams.mtl for transmitters to be published.) 54480
- ♦ Aggregates with sub aggregates did not convoy. 53096
- ♦ The back-end crashed if you tried to create multiple entities using B-HAVE and you did not select an area. 53098
- ♦ Disembarkation behaviors were inconsistent. 52436

