



VR-Forces 3.11.0.1 Release Notes

This release note contains the following release-specific information for VR-Forces Release 3.11.0.1:

Systems Supported	3
Libraries and Binaries Built with Visual Studio 2005	3
Building on Linux	4
Disk Space Requirements	4
Compiler Compatibility on Windows	4
MÄK Product Compatibility	4
Qt Release Compatibility	4
FLEXlm Support	5
Third-Party Library Requirements	5
Patch Required for AMD Dual-processor Windows PCs	5
Font Support on Linux	5
Network Compatibility	6
Backwards Compatibility	6
RPR FOM Versions Supported	7
New Features and Updates	8
Installing VR-Forces 3.11.0.1 and Related Applications	8
The Performance Options Plug-in	9
QualNet Integration Plug-in	10
Enabling the External Communication Model	10
Updated Documentation	11
Fixed Bugs	12

Copyright © 2008 MÄK Technologies, 68 Moulton St., Cambridge, MA 02138
All rights reserved. MÄK Technologies®, VR-Forces®, RTIspey®, and VR-Link® are registered trademarks of MÄK Technologies, Inc.
Document ID: VRF-3.11.0.1-3-081103

[Known Problems](#) 14

Systems Supported

VR-Forces 3.11.0.1 is available for the platforms listed in [Table 1](#). For the availability of other platforms, contact your MÄK salesperson. 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
Red Hat Enterprise Linux Workstation 4, 5 *	Default compiler
Fedora Core 7	
Windows XP	Microsoft Visual C++ 7.1 Microsoft Visual C++ 8.0 SP1
Windows Vista **	Microsoft Visual C++ 8.0 SP1
*The VR-Forces 3D Front End for Linux is only available on WS 4.	
** The VR-Forces 3D Front End is not supported on Windows Vista.	

Libraries and Binaries Built with Visual Studio 2005

All MÄK products built with Microsoft Visual Studio require the C Runtime Library to function. The C runtime libraries have always been available from Microsoft for download, they are also installed on a user's machine when a Microsoft compiler is installed. The C runtime libraries are not part of the normal Windows installation. For customers who plan to use MÄK products on machines that do not have a compiler installed, MÄK has historically distributed a copy of the C Runtime Libraries with MÄK products. These libraries were put in the *bin* directory used by the MÄK products. MÄK products would then use the libraries in the *bin* directory and customers would not have a problem if copies of the libraries were not already installed.

Unfortunately, with the release of the new C Runtime Libraries required by Microsoft Visual Studio 2005 (MSVC++8.0), the libraries can no longer just be copied into the *bin* directory of an application. The libraries need to be installed correctly into Windows system folders. (The process is actually a little more complicated, a manifest file needs to be created to tell Windows where to find the libraries.)

To accommodate this change, MÄK is distributing the Windows installer for the C runtime libraries with all MÄK products released for MSVC++8.0. The installer is named *vc redistrib_x86.exe* and is in the base directory of any installed MÄK product.

Running the installer requires Administrator privileges for the machine the installer is run on. MÄK has chosen to not integrate the MÄK installer and the Microsoft installer so as not to require users to have Administrator privileges to install MÄK products. Therefore, if you do not have a compiler installed, or get error messages like "Software has not been installed correctly, please re-install", you must apply the patch.

For more information see this Microsoft URL:

<http://msdn2.microsoft.com/en-us/library/ms235299.aspx>

Building on Linux

The example Makefiles require gmake 3.81 or later. For operating systems with an older version of gmake (currently only Red Hat WS 4), a binary has been included for your convenience in the `./mkbin` directory. This version includes a memory fix that is needed to use our makefiles. The source code for the modified gmake is freely available from <http://ftp.mak.com/out/gmake3.80-patch1.tar.gz>.

Before you compile the examples, go to the top level of your installation and create a symbolic link called 'VR-Link' to your VR-Link installation, and another called 'RTI' to your RTI installation. You must use the versions listed under [MÄK Product Compatibility](#).

Disk Space Requirements

A full installation of VR-Forces requires approximately 600 MB of disk space. Terrain databases use approximately 110 MB and documentation (primarily the class reference) uses 200 MB.

Compiler Compatibility on Windows

MÄK provides versions of product releases that have been compiled with Microsoft Visual C++ 7.1 and 8.0. When you run MÄK products together on the same computer, for example, VR-Forces and the 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 3.11.1. If you are building for HLA and want to link with the MÄK RTI, use MÄK RTI 3.x.

VR-Forces 3.11.0.1 is binary compatible with VR-Forces 3.11. This means that it works with VR-Forces 3D Front End for VR-Forces 3.11, B-HAVE Module 1.2 for VR-Forces 3.11, and Human Characters for MÄK Products 1.0.

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 Trolltech. The VR-Forces GUI was built with Qt release 4.3.3. A VR-Forces developer's license includes a Qt development license. MÄK provides you with a Qt license key and a link for downloading Qt.

FLEXlm Support

VR-Forces 3.11.0.1 uses FLEXlm 11.4. You must copy the files in *./flexlm11.4* to your license server directory. You do not need to change license files (unless your maintenance agreement has expired.)

Third-Party Library Requirements

DtTerrainProfileWidget is based, in part, on the work of the Qwt project (<http://qwt.sf.net>). To compile examples or user projects, you must download the QWT distribution at <ftp://ftp.mak.com/out/qwt-0.4.1.zip>. This distribution is known to work with VR-Forces. Set the MAK_QWTDIR environment variable to point to the location of the QWT library.

Patch Required for AMD Dual-processor Windows PCs

VR-Link-based products use a high resolution counter for time calculations on Windows PCs. Customers who are running Windows on PCs with multiple AMD Athlon 64-bit processors may notice clock jitter, which may cause time in MAK products to run backwards. This occurs when the Windows scheduler changes the CPU the MAK process is using. If the high resolution counters on each processor are not synchronized, the application may witness a decrease in the high resolution counter value stored in the processor causing an incorrect time calculation. To fix this problem, apply the AMD Dual-Core Optimizer patch provided by AMD. You can get the patch at:

http://www.amd.com/us-en/Processors/TechnicalResources/0,,30_182_871_9706,00.html



If you get an error when you try to access this URL, reload the page.

Font Support on Linux

The *fontconfig* provided with Red Hat Enterprise Linux Workstation 4 does not properly name the fonts. Newer operating systems are not affected by this issue. In order to load the fonts, you'll need to provide the following alias in *fontconfig*. The alias can either be added to the system configuration in */etc/fonts/fonts.conf* or it can be added to *~/.fonts.conf* on a per-user basis. For more information about *fontconfig*, please see <http://fontconfig.org/fontconfig-user.html>.

```
<alias>
  <family>MAK MilStd 2525b</family>
  <prefer><family>MilStd 2525b</family></prefer>
</alias>
```

Network Compatibility

HLA only

VR-Forces 3.11.0.1 was built against VR-Link 3.11.1 and is compliant with:

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

VR-Forces 3.11.0.1 is compatible with applications that use earlier versions of VR-Link if they support versions of the RPR FOM listed.



VR-Forces is no longer compatible with the DMSO RTI-NG v6, because that RTI was not built using any of the compilers supported by VR-Forces 3.11.0.1. VR-Forces 3.11.0.1 should be compatible with commercial successors to the DMSO RTI if they conform to the HLA 1.3 or 1516 (SISO DLC API) specifications and are built with a compiler supported by VR-Forces.

DIS only

VR-Forces 3.11.0.1 supports DIS 2.0.4, IEEE 1278.1, 2.1.4, and IEEE 1278.1a, and can therefore interoperate with DIS applications of any of these versions.

Backwards Compatibility

The VR-Forces 3.11.0.1 APIs are not compatible with VR-Forces 3.10.x or earlier. Scenarios and object parameter databases created with VR-Forces 3.10.x are forward compatible. VR-Forces 3.11.0.1 is fully compatible with applications and plug-ins built with VR-Forces 3.11. For more information, please see [“Installing VR-Forces 3.11.0.1 and Related Applications,”](#) on page 8.

RPR FOM Versions Supported

VR-Forces 3.11.0.1 has built-in support for versions 1.0 and 2.0 (draft 6 and 17) of the RPR FOM. It also supports VR-Link's ability to support alternative FOMs through the FOM Mapper. By default, VR-Forces 3.11.0.1 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 3.11.0.1 relies on the LgrControl and ViewControl custom MÄK extensions.

New Features and Updates

VR-Forces 3.11.0.1 is primarily a bug-fix release. However, the installer includes two optional plug-ins that improve integration with QualNet and allow you to adjust performance and quality in the VR-Forces 3D Front End.

VR-Forces 3.11.0.1 installs two new scenarios for use with QualNet. They are called `vrforces-dtedTest` and `carBomb`, and are installed in the `./data/scenarios/QualNet` directory. The `carBomb` scenario requires the B-HAVE Module.

Installing VR-Forces 3.11.0.1 and Related Applications

VR-Forces 3.11.0.1 installs into its own directory, just like any other maintenance or feature release. However, it changes the registry so that other installers that look for VR-Forces 3.11 automatically find VR-Forces 3.11.0.1 instead.

After you install VR-Forces 3.11.0.1, you can install related applications and plug-ins, such as VR-Forces 3D Front End 1.0 and B-HAVE 1.2 for VR-Forces 3.11 to the VR-Forces 3.11.0.1 directory.

When you install some MÄK plug-ins on top of VRF 3.11.0.1, the installer places the start menu shortcuts in the VRF 3.11 program group by default. Be sure to change the program group during the installation process.

The VR-Forces 3.11.0.1 installation wizard gives you a choice of installing Typical, Minimal, or Custom versions of VR-Forces. If you choose Typical, the installer automatically installs the QualNet Ease-of-Use plug-in and the Performance Options plug-in. If you choose Custom, you can specify which plug-ins, if any, to install. They are described in the following sections.

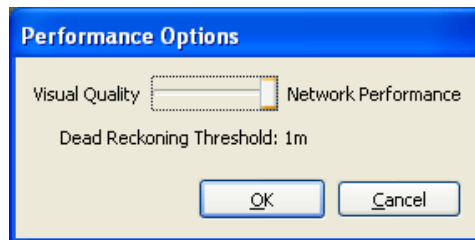
The Performance Options Plug-in

VR-Forces currently allows you to tune performance by adjusting dead-reckoning thresholds in the Networking Options dialog box. The primary benefit of this feature is to improve the appearance of entities in 3D visualization programs such as the MÄK Stealth and VR-Forces 3D Front End. Reducing thresholds improves entity appearance. However it does so at the expense of network performance, because a lower dead-reckoning threshold results in more packets being sent for each entity. Unfortunately, the purpose of the Networking Options feature is not obvious. The Performance Options plug-in simplifies the tuning process by replacing the Networking Options dialog box with the Performance Options dialog box.

When you change the performance options setting, it gets saved. The next time you run VR-Forces, the saved setting is used.

To configure performance options:

1. Choose **Options** → **Performance Options**. The Performance Options dialog box opens.



2. Adjust the slider to your desired preference for visual quality versus network performance.
3. Click OK.

QualNet Integration Plug-in

VR-Forces 3.11 introduced support for external communications servers and, specifically, the interface specified in the file `./doc/QualNet_HLA_interface_RPR-FOM 1.0_extensions_ICD3.pdf`, which was developed by Scalable Network Technologies. Use of an external communications model required manual editing of entity OPE files and `commModelParams.mtl`. The new plug-in makes configuration possible through the VR-Forces GUI. It does the following:

- Enables the external communications model in `commModelParams.mtl`
- Enables publishing of transmitters for all entities that have radios.

This release note also includes updated documentation for using VR-Forces with QualNet.

Enabling the External Communication Model

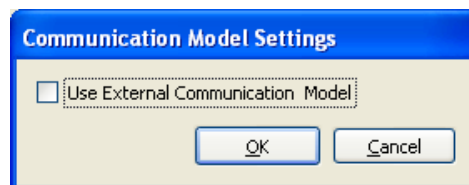
When you enable the external communication model, VR-Forces also enables publishing of transmitters for all entities that have radios. When you change the communication model setting, it gets saved. The next time you run VR-Forces, the saved setting is used.



When you change the communication model setting, existing transmitters are destroyed and new transmitters are created. Therefore, changing this option while running a scenario with QualNet will cause QualNet to no longer recognize the transmitters being published.

To enable the external communication model:

1. Choose **Options** → **Communication Model Settings**. The Communication Model Settings dialog box opens.



2. Select the Use External Communication Model check box.
3. Click OK.

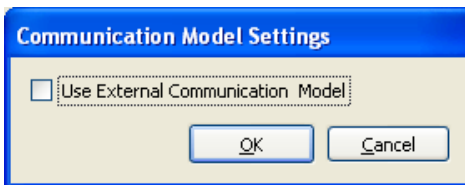
Updated Documentation

The following section is an updated version of section 11.4.1 in *VR-Forces Configuration Guide*. It provides more details of the process of running the example *VR-Forces Configuration Guide* scenario with the example QualNet scenario.

Running the Scenario

To run the scenario, use the following steps:

1. Copy the `./data/scenarios/QualNet/vrforces-CommDemo` folder to the `QualNet/scenarios` folder.
2. Start your RTI. If you are using the MÄK RTI, run the `rtiexec`.
3. Run QualNet.
4. In the QualNet window, select the Scenario Designer tab.
5. Expand the file system tree for the `scenarios` folder.
6. Expand the tree for the `vrforces-CommDemo` scenario.
7. Right-click `vrforces-CommDemo.config`. A popup menu is displayed.
8. Choose **Run Scenario**. QualNet loads the scenario and creates a federation.
9. Start VR-Forces. Be sure that you use the HLA 1.3 RPR FOM 1.0 version and that you connect to the same `rtiexec` as you did for QualNet.
10. Choose **Options** → **Communication Model Settings**. The Communication Model Settings dialog box opens.



11. Select the Use External Communication Model check box.
12. Click OK.
13. Open the `QualNet/commsDemo` scenario. When the scenario is fully loaded, the Run button on the QualNet toolbar should become enabled. If it does not, there may be a communication problem between the two applications.
14. If you want to see communications lines in VR-Forces, choose **Options** → **Transmitters**. In the Transmitter Options dialog box, select the Transmitting check box and any of the other check boxes in the Transmitting group box.
15. Click the Play button on the QualNet toolbar.
16. Run the scenario in VR-Forces.

Fixed Bugs

This release fixes the following problems that were present in prior releases:

- Dismounted infantry could not enter the highway tunnel in the VR-Village terrain.
- The showMissileImpact example crashed when you closed a scenario.
- If you changed the origin LOS elevation for an intervisibility line, the end point also changed.
- Pasting a forward slash into the name of a new entity in the Entity Editor (for example, F/A-18) caused creation of the new OPE file to fail.
- The --sessionId command line argument did not work.
- It was not possible to change an entity name or label to a different language.
- Pasting an entity failed if an entity did not have a plan and its rules of engagement were not set to fire-at-will.
- Ellipses were displayed as circles when zoomed out.
- If you tried to edit a condition that only referenced Self, VR-Forces crashed.
- VR-Forces crashed if it received a report message with an unregistered report type.
- Range rings did not draw patterns.
- Translated utility bar tabs also showed English labels.
- Wedge drawing for circles did not work.
- VR-Forces could not load *overlayPalette.mtl* files from multiple simulation model sets.
- The ground joystick controller did not work correctly.
- If you used the heading indicator to change an entity's heading, VR-Forces did not mark the scenario as having changed.
- The C7L loader crashed when loading water features.
- If you used the API to remove a tab from the utility bar, VR-Forces crashed on exit.
- DtIfStatus was not created using the factory.
- The OPD Editor did not display anything on the Systems tab on Linux.
- Dragging a missile that was flying along a route crashed the back-end.
- Changing an entity's name caused its plan to be lost.
- If a simple scenario using the external communication model with use-ready-to-send set to false was rewound, the back-end crashed.
- DFAD linear features were not imported correctly.
- The shape vector data importer did not work correctly.
- The --outputLogFile command line argument did not work for the front-end.
- The Scenario Merge Load Project dialog box displayed the last loaded scenario file.
- The back-end crashed if a fixed-wing entity could not find the waypoint for a Move To task.
- Documentation of the process for building front-end examples was improved.

- ♦ Loading scenarios with multiple OPD files in the presence of remotely simulated entities caused a crash.
- ♦ The implementations of `setLocalAcceleration()` and `localAcceleration` were missing.
- ♦ The *pvdEchelonView.h* header file had an error.
- ♦ Aggregates created from the Create menu were not ground clamped.
- ♦ DI-Guy entities did not respond to Set Weapon State.
- ♦ The range filter example did not call the correct parent method for applying symbol modifications.
- ♦ Right-clicking the minimap caused an error.
- ♦ Aggregate subordinates were not always displayed correctly in the echelon view.
- ♦ The detonation interaction munition ID was not getting set under certain circumstances, such as a destroyer firing a cruise missile.
- ♦ The altitude displayed for a point on a multi-level structure was not the same as the altitude of the polygon selected for that point.
- ♦ Setting `use-ready-to-send` to False in the *commModelParams.mtl* when integrating with QualNet did not work correctly.
- ♦ The comments for the *DtFixedWingPilotControllerDescriptor* `myMxTurnRateCut-offAngle` parameter were incorrect. The correct unit is radians, not degrees.
- ♦ The breaching scenario and takeoffandlanding scenario files specified a GDB file instead of an MTD file. This created problems when loading these scenarios in the 3D Front End.
- ♦ Dead reckoning threshold changes set at runtime though the API are now reflected on the network correctly.
- ♦ The helicopters in the Makland scenario did not consume resources.
- ♦ The OPD Editor did not properly update ingress points.
- ♦ The Spanish translation file caused a crash.

Known Problems

VR-Forces Release 3.11.0.1 has the following known problems:

- ♦ The intervisibility display does not always reflect what simulated entities perceive in the back-end. For example, the intervisibility does not ignore water surface polygons while entity models do.
- ♦ Occasionally, VR-Forces is unable to load a scenario. When this happens, delete the contents of the `./data/temp` directory and try loading again.
- ♦ The Subordinate Manager allows duplicate entries to exist.
- ♦ Subsurface entities sink to unreasonable depths when destroyed.
- ♦ DIs may unexpectedly change posture when climbing steep terrain.
- ♦ Determining when a scenario is completely loaded (all entities have been created and the aggregate hierarchy is completely constructed) is up to the user. Deciding when a scenario is completely loaded and ready to run can be difficult with large, distributed scenarios.
- ♦ Avoidance of linear vector features often fails at segment intersections.
- ♦ Auto-generated aggregate formation routes may contain spurious points if the original route contains sharp turns and short line segments.
- ♦ Intersection lines in intervisibility fans sometime change colors as you change the zoom level.
- ♦ Torpedoes can fly above water in certain circumstances.
- ♦ Fixed-wing entities crash when tasked to move into formation.
- ♦ Surface entities cannot move on the *California_High.gdb* terrain database.
- ♦ A disembark statement followed by a set formation statement in a plan does not work.
- ♦ You cannot use the lifeform acquire controller with a missile launcher.
- ♦ Ground entities' speed does not affect turning radius.
- ♦ Entity intervisibility with point features does not work correctly.
- ♦ Fixed-wing entities generate too many PDUs when moving to a waypoint in a straight line.
- ♦ Rotary-wing terrain avoidance sometimes fails.
- ♦ Putting the *vrfSim* console in select mode freezes the back-end.
- ♦ Rotary-wing entities do not maintain correct altitude in geocentric terrain when moving very large distances.
- ♦ You cannot create a new scenario with a Linux GUI and a Windows back-end.
- ♦ Aggregate subordinates cannot execute their own individual plans.
- ♦ In-frustum sensor geometry does not work for angles greater than 90 degrees.
- ♦ Geometry files referenced by a MetaFlight file must be in OpenFlight format, and not in Vega Prime's binary format (*.vsb*). MÄK's terrain libraries do not support *.vsb* files.

- ♦ Terrain databases processed with the balance option may display incorrectly.
- ♦ Surface entities are not configured with weapons.
- ♦ You cannot load vector data on geocentric terrain.
- ♦ When you record a scenario with MÄK Logger 4.0, at the beginning of the recording session, VR-Forces may freeze for a while due to the large number of messages required for subscription and publication.
- ♦ Do not play the *maklanddemoHLA1516_vrlink.xml.lgr* file (included with MÄK Logger 4.0 release) while running VR-Forces using draft 17 of the RPR FOM. Running this file in this particular configuration produces memory leaks. Run VR-Forces using RPR FOM 1.0 when working with this Logger recording.
- ♦ Rotary-wing entities are not able to reach moving waypoints or land on moving ships. A rotary-wing entity will approach a given waypoint or landing point, but will never completely reach the goal.
- ♦ Reorganizing an aggregate while it is executing a task may produce unexpected results. The aggregate may be unable to complete its task.
- ♦ When control objects are placed on moving objects (such as a route on a moving aircraft carrier), they produce a flood of network packets.
- ♦ Open File dialog boxes may be slow to display file lists. This is due to underlying problems with the Qt toolkit.
- ♦ When you execute a task that requires use of an Open File dialog box, you may receive an error message similar to:

Exception Processing Message

This is due to problems in the Qt Toolkit. To work around this problem, insert a floppy disk in the A: drive.
- ♦ Ground vehicles can climb up unrealistically steep slopes (greater than 70 degrees in some cases).
- ♦ Features cannot be selected on geodetic terrain databases in the TDB Tool.
- ♦ Embarked entities cannot shoot or be shot at by other entities.
- ♦ The *DtOrganizationManager* cannot handle changing the force of an entity. It is unable to update the echelon ID of the entity correctly.
- ♦ Fixed-wing entities do not respond to Turn to Heading tasks.
- ♦ Dead lifeform entities may unexpectedly block line of sight. Their bounding volumes are always upright, regardless of whether or not they are dead.
- ♦ Rotary-wing entities do not avoid other entities. They will fly through one another.
- ♦ Transmitters on newly created entities sometimes do not stay in position with the entity, and stay in a static position on the screen as the map is moved. To fix this problem, toggle the **Options** → **View Transmitters** menu command.
- ♦ The target acquisition and selection model can sometimes get 'stuck' trying to acquire a target. If a target has been selected but is unable to be acquired for some reason, it will continue in a futile attempt to acquire that target, even if there are other more viable targets available.

- ♦ If you set the pan increment to 100% for small terrains, the map does not pan by 100%. This is due to the algorithm used to calculate the distance that the map is offset.
- ♦ Transmitter lines are not drawn if the sender is off screen.
- ♦ If you use the MÄK Logger to record VR-Forces scenarios in HLA1516 (particularly with RPR FOM version 2.0 draft 17), and you are using the MÄK RTI, VR-Forces and the Logger may become unresponsive. To avoid this problem, disable `RTI_enableLrcWebService` in the *rid.mtl* configuration file.
- ♦ The MÄK Logger does not record batch file runs if a previous recording exists. By default, the Logger does not allow you to overwrite existing files using the remote control API. So, once you record a batch run, you cannot record a subsequent run of the same batch. You can work around this problem by editing the *lgrConfig.xml* file to permit overwriting of Logger files. Make the following edits in the configuration file for each protocol for which you want to record batch runs:

Add the following lines to the <defaults> section:

```
<var name="processRemoteControl" type="bool" value="true"/>
<var name="recordRemoteControl" type="bool" value="true"/>
<var name="allowDeleteRemoteControl" type="bool" value="true"/>
```

Add the following lines to the <commands name="startup"> section:

```
<command domain="Simulation" protocol="DIS"
  command="SetRemoteControlSettings">
  <param name="process" var="processRemoteControl"></param>
  <param name="record" var="recordRemoteControl"></param>
  <param name="allowDelete" var="allowDeleteRemoteControl"></param>
</command>
```

Use the appropriate value for the protocol attribute.

- ♦ If you create an aggregate composed of rotary-wing entities, in which some or all of them are on the ground, when you start the scenario, they will be destroyed. To prevent this from happening, set the aggregate's altitude to a reasonable height above terrain (such that all rotary-wing entities are airborne) before starting the scenario.
- ♦ Embarked entities controlled by joystick cannot disembark. For example, if you use a joystick to control a fixed-wing taking off from a carrier, the aircraft will take off, but it will remain in an embarked state.
- ♦ Certain GeoTIFF images may fail to load on linux. If you encounter such an image, please contact MÄK support.
- ♦ Rotary wing entities which are on the ground remain ground clamped even when given a set location request with a high altitude.