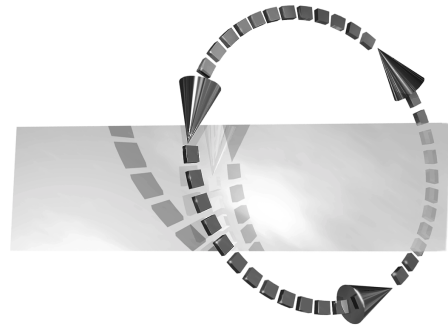




MÄK VR-Link 3.6.1-ngc Release Notes

This release note provides the following release-specific information for MÄK VR-Link Release 3.6.1-ngc:

Systems Supported	2
Disk Space Requirements	2
Backward Compatibility	2
Network Compatibility	3
RPR FOM Versions Supported	3
RTI Support	4
New Features and Changes	5
Bug Fixes	6

Systems Supported

Table 1 lists the platforms currently supported by MÄK VR-Link 3.6.1-ngc. Please contact MÄK if you are interested in purchasing VR-Link for other platforms. Application code must be built with the indicated compilers in order to link to VR-Link libraries.

Table 1: Platforms supported

Platform	Compiler
SGI IRIX6.5	MipsPro7.2.1 C++ compiler (available from SGI)
Sun Sparcstation with Solaris2.7 or later	SPARCCompiler C++ version 5.2 (available from Sun)
PC with Red Hat Linux 6.0 (Linux 2.2.16-3)	GNU C++ (g++) distributed with Red Hat distribution. The command: <code>g++ -v</code> returns: gcc version egcs-2.91.66 19990314/Linux (ecgs-1.1.2 release)
PC with Windows NT, Windows 95/98/2000	Microsoft Visual C++ 6.0.

Disk Space Requirements

On SGI IRIX, you need approximately 270 MB for VR-Link files and 50 MB for all documentation files. Other UNIX installations require approximately 50 MB for VR-Link files and an additional 50 MB for documentation files.

On PCs, you need 50 MB for VR-Link files and 50 MB for all documentation files.

Online Help Browser Requirements

The online help system for the FOM Mapper Builder is an HTML-based system. To view online help, you must have a web browser installed that supports Javascript and Java. Java and Javascript must be enabled.

Backward Compatibility

VR-Link 3.6.1-ngc is completely compatible with VR-Link 3.6-ngc. There is a high degree of backwards compatibility with VR-Link 3.5-ngc. For example, the *talk* and *listen* example programs from VR-Link 3.5-ngc compile and link unmodified against VR-Link 3.6.1-ngc.

Network Compatibility

HLA only

VR-Link 3.6.1-ngc is compliant with:

- ♦ RPR-FOM 0.5, 0.7, 0.8, and 1.0
- ♦ MÄK Real-time RTI 1.3.3-ngc and 1.3.4-ngc
- ♦ DMSO RTI NG v1, v2, v3.0, v3.1, v3.2.

DIS only

This release 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.

RPR FOM Versions Supported

VR-Link 3.6.1-ngc has built-in support for versions 0.5, 0.7, 0.8, and 1.0 of the RPR FOM. It also supports VR-Link's ability to support alternative FOMs through the FOM Mapper. By default, VR-Link 3.6.1-ngc uses RPR FOM 1.0.

If you want to use a version of the RPR FOM other than 1.0, pass the version number (0.5, 0.7, or 0.8) 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-Link08", "MyApp", new DtRprFomMapper(0.8));
```

VR-Link examples like *f18* and *hlaNetdump* have a command line option, *-f*, that you can use to choose an RPR FOM version by passing either *v1RPR05*, *v1RPR07*, or *v1RPR08* as the argument to *-f*.

RTI Support

VR-Link 3.6.1-ngc comes with the MÄK Real-time RTI 1.3.4-ngc, which is link-compatible with DMSO RTI NG. VR-Link 3.6.1-ngc has been tested with DMSO RTI NG v1, v2, v3.0, and v3.2.

VR-Link 3.6.1 may work with older versions of the MÄK Real-time RTI, but we recommend using it with release 1.3.3-ngc or release 1.3.4-ngc.

If you use the MÄK Real-time RTI, remember to make sure that VR-Link can find your FED file, and optional *rid.mtl*. Put the FED file in the directory from which you are running, or set the environment variable RTI_CONFIG to the directory that contains it. See the MÄK Real-time RTI 1.3.4-ngc Release Notes for a list of the options for specifying the location of the *rid.mtl* file.

Library Names

VR-Link 3.6.1-ngc expects library names that match those of DMSO RTI NG v3.2. If you run VR-Link 3.6.1-ngc with DMSO RTI NG v3.2, both the UNIX and PC versions work without any modification. If you want to run VR-Link 3.6.1-ngc with DMSO RTI NG v1 or v2, you must make the following changes:

- For UNIX, you must use the libraries *libRTI-NGv2.so* or *libRTI-NGv1.so*, which are supplied by MÄK in the *.lib* directory. These libraries correspond to DMSO RTI NG v1 and v2. To use VR-Link 3.6.1-ngc with RTI NG v1 or v2, copy the appropriate library to the DMSO RTI NG *lib* directory and rename it *libRTI-NG.so*. (You might want to rename the v3.x *libRTI-NG.so* library first.)
- For PCs, rename the libraries that come with RTI NG v1 or v2 to the names used by the RTI NG v3.2 libraries (*libRTI-NG.lib* and *libfedtime.lib*).

New Features and Changes

This section describes the following features and changes to VR-Link:

- ♦ Ability to work without having RTI advisories on
- ♦ Support for IFF objects
- ♦ Ability to choose the network device to use with multicast traffic
- ♦ Updates to the FOM Mapper Builder.

Ability to Work Without RTI Advisories

The RTI uses advisory messages to let federates know that other federates have subscribed to parameters and attributes that the federate has published. This information gives VR-Link the ability to only send out updates when it knows that other federates are interested in receiving them. However, advisories require an extra amount of handshaking between federates and the RTI, which may slow network response. The function call `DtExerciseConn::ignoreAdvisories()` allows a federate to indicate that it wants to send updates regardless of whether or not other federates have subscribed to its attributes or parameters.



If you disable advisory messages in the RTI's RID file, you must call `DtExerciseConn::ignoreAdvisories()`. Otherwise, VR-Link will not send updates and interactions.

Support for IFF Objects

This release of VR-Link supports IFF objects based on RPR FOM 2.0 draft 4. See the header files *iffSR.h*, *iffPub.h*, *refIff.h*, and *refIffList.h*.

Choosing the Network Device for Multicast Traffic (DIS Only)

VR-Link allows you to set the device to use for multicast traffic in DIS. This is done by creating a *DtNetSocket* and passing it into the constructor for your normal *DtExerciseConn*, for example:

```
DtNetSocket* ns = new DtNetSocket(3000, DtInetAddr("229.7.7.7"),
    DtReadWrite, DtInetAddr("184.83.3.12"));
DtExerciseConn* ec = new DtExerciseConn(ns, 1, 2, 3);
```

The example creates a *DtNetSocket* listening to multicast address 229.7.7.7 on port 3000 on the device whose address is 184.83.3.12. This *DtNetSocket* is then used by the *DtExerciseConn* for communication in exercise ID 1 with site ID 2 and application ID 3.

Updates to the FOM Mapper Builder

The FOM Mapper Builder has the following new features and fixes to known problems from the previous release.

- ♦ The FOM Mapper Builder no longer requires an RTI. Consequently the problem caused by differences between how the DMSO RTI and the MÄK Real-Time RTI capitalize attribute names is no longer an issue.
- ♦ The FOM Mapper Builder displays a pop-up menu if you right-click a FOM class. The menu allows you to show properties, unmap a class, or undo all changes to a class.
- ♦ The previous release allowed IOM classes of different class derivations to be mapped to FOM classes of the same class derivation. There is no support for this type of mapping in VR-Link. IOM class mappings to FOM classes in the same class derivation must now be from the same IOM class derivation.

An exception is the extension classes *myExtendRepository* and *myExtendInteraction*. Since these serve only as placeholders, they can be mapped to any FOM class without regard to previous or subsequent IOM mappings. The caveat is that they should be replaced with an extension that does fall under the IOM class derivation.

- ♦ Unmapping an attribute is not allowed when attribute mapping is inactive. (This fixes a known problem in the last release.)
- ♦ The labels for mapped classes (showing mappings) are now restored if a class was unmapped followed by an undo. (This fixes a known problem in the last release.)

Bug Fixes

VR-Link 3.6.1 fixes the following bugs:

- ♦ In HLA, VR-Link incorrectly encoded and decoded the number of environment records in *DtEnvironmentProcessSR* as an unsigned short. It should be an unsigned long.
- ♦ In *DtOwnershipHandler*, the functions *attributeOwnershipAcquisitionNotification()* and *attributeOwnershipDivestitureNotification()* were not getting called.
- ♦ Setting a discovery condition for *DtReflectedEmitterSystemLists* did not work.

Known Problems

In the Bookmarks panel of the PDF version of the *VR-Link 3.6 Release Notes*, the section describing the FOM Mapper Builder is incorrectly listed under the Known Problems bookmark.