



VR-Link 3.10.1 Release Notes

This release note provides the following release-specific information for VR-Link Release 3.10.1:

Systems Supported.....	2
Disk Space Requirements	2
Building VR-Link on UNIX	2
Compiler Compatibility on Windows	3
FLEXlm Support.....	3
Backward Compatibility	3
Network Compatibility.....	3
RPR FOM Versions Supported.....	3
RTI Support	4
New Features and Changes	4
Bug Fixes	4
Known Problems	5

Copyright © 2006 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: VRL-3.10.1-3-060817

Systems Supported

Table 1 lists the platforms currently supported by MÄK VR-Link 3.10.1. 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 for HLA and DIS

Platform	Compiler
SGI IRIX 6.5	MipsPro7.3 C++ compiler (available from SGI)
Sun Sparcstation with Solaris 2.8 or later	SPARCompiler C++ version 5.2 (available from Sun)
Red Hat Enterprise Linux Workstation 3.0	GNU C++ (GCC) 3.2
Red Hat Fedora Core 3	GNU C++ (GCC) 3.4.4
Red Hat Enterprise Linux Workstation 4.0	GNU C++ (GCC) 3.4.4
PC with Windows 2000/XP	Microsoft Visual C++ 7.1
	Microsoft Visual C++ 8.0

Table 2: Platforms supported for TENA

Platform	Compiler
Red Hat Fedora Core 3	GNU C++ (GCC) 3.4.4
Red Hat Enterprise Linux Workstation 4.0	GNU C++ (GCC) 3.4.4
PC with Windows 2000/XP	Microsoft Visual C++ 7.1

Disk Space Requirements

On SGI IRIX, you need approximately 270 MB for VR-Link files and 60 MB for all documentation files. Other UNIX installations require approximately 50 MB for VR-Link files and an additional 60 MB for documentation files.

On PCs, you need 50 MB for VR-Link files and 60 MB for all documentation files.

Building VR-Link on UNIX

VR-Link does not require you to use the MÄK build system, but if you choose to, you must use gmake 3.81 or later. You can download it from the following URL:

<http://savannah.gnu.org/projects/make/>

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, for example, the Logger 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.

FLEXlm Support

VR-Link 3.10 uses FLEXlm 10.8 for Windows and Linux versions. Solaris and IRIX versions continue to use FLEXlm 9.2.

Backward Compatibility

VR-Link 3.10.1 is not fully compatible with 3.9.6 and earlier releases due to changes to *vlutil* and to the link line for UNIX. For more details, please see “Compiling and Linking,” and “Changes to the vlutil Library,” in *VR-Link 3.10 Release Notes*.

Network Compatibility

HLA only

VR-Link 3.10.1 is compliant with:

- ♦ RPR-FOM 0.5, 0.7, 0.8, 1.0, and a subset of 2.0 (draft 6, 14, and 17)
- ♦ MÄK RTI 2.x, 3.x
- ♦ DMSO RTI NG 6.0
- ♦ Pitch RTI 1.3 C++ interface.

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 3.10.1 has built-in support for versions 0.5, 0.7, 0.8, 1.0, and 2.0, drafts 6, 14, and 17, of the RPR FOM. By default, VR-Link 3.10.1 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, 0.8, 2.0006, 2.0014, or 2.0017) 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-Link20017", "MyApp", new DtRprFomMapper(2.0017));
```

VR-Link examples like *f18* and *hlaNetdump* have a command line option, `--rprFomVersion`, that you can use to choose a RPR FOM version, using one of the version numbers listed in the previous paragraph.

RTI Support

VR-Link 3.10.1 has been tested with DMSO RTI NG v6, Pitch RTI 1.3, and MÄK RTI 3.0.

VR-Link 3.10.1 may work with older versions of the MÄK RTI, but we recommend using it with release 3.0.



VR-Link 3.10.1 for Sun Solaris is not compatible with versions of the MÄK RTI prior to version 2.4.

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

New Features and Changes

VR-Link 3.10.1 is a maintenance release. It has no new features.

Bug Fixes

VR-Link 3.10.1 fixes the following bugs:

- ♦ A bug has been fixed in the Common Object LROM Mapper that caused it to throw an exception when uninitialized platform position, velocity, acceleration, or orientation was received from a publishing application.

Known Problems

This release of VR-Link has the following known problems:

- ♦ VR-Link correctly encodes and decodes `EnvironmentalProcess` objects for RPR FOM 2, Draft 14. However, the RPR FOM specifies a way of encoding which prevents VR-Link from correctly decoding modified or custom `EnvironmentalRecords`. Unfortunately, many MÄK products built with VR-Link use customized `EnvironmentalRecords`. If you use these products or if you use customized records in your applications, please do not use the RPR FOM 2 Draft 14 FOM Mapper. This is not a problem in other FOM versions, and it has been resolved in future versions of the FOM.
- ♦ The RTI 1516 API specifies that all strings passed to and from the RTI are handled as wide strings. VR-Link (and the MÄK RTI) store strings internally as narrow strings. This means that true multibyte wide strings used by RTI Federates may not be handled correctly when received by VR-Link. VR-Link handles name reservation wide strings correctly, however, as this is a 1516-only function. VR-Link provides conversion functions from Wide To Narrow/Narrow to Wide string conversion in *vlStringUtil.h*.
- ♦ On non-windows platforms, the classes *DtSharedMemoryPoolManager* and *DtSharedMemoryPoolClient* (*vlShmPool.h*) sometimes have difficulty creating a large memory pool (greater than 1 MB) if the pool name is longer than two or three characters. The classes refuse to create the memory pool even if the requested pool is smaller than the system resource SHMMAX. Setting the pool name to fewer than three characters will work around this problem.
- ♦ Section 6.10 in *VR-Link Developer's Guide* describes interoperability issues that arise when you try to run an HLA 1.3 federate with an IEEE 1516 federate. In particular it describes the requirement to use an identical FED or FDD file for all federates. In future releases, we intend to rectify this situation, and allow run-time interoperability between federates that have loaded a 1.3-style FED file and federates that have loaded a 1516-style XML file that otherwise includes the same set of classes, attributes and parameters.

