



VR-Link 5.8

Release Notes

This release note provides the following release-specific information for VR-Link Release 5.8:

Systems Supported.....	2
Disk Space Requirements.....	2
Compiler Compatibility on Windows.....	2
License Manager.....	3
Network Compatibility.....	3
Backward Compatibility.....	4
Using the Compatibility Directory.....	4
RTI Support.....	5
New Features and Changes.....	5
Fixed Bugs.....	6

Copyright © 2023 MAK Technologies, 10 Fawcett Street, Suite 204, Cambridge, MA 02138. All rights reserved. VR-Exchange™, VR-TheWorld™, VR-Vantage™, DI-Guy™, and DI-Guy Scenario™ are trademarks of MAK Technologies. MAK Technologies®, VR-Forces®, RTIsPy®, B-HAVE®, and VR-Link® are registered trademarks of MAK Technologies.
Document ID: VRL-5.8-3-230913

Systems Supported

[Table 1](#) lists the platforms currently supported by MAK VR-Link 5.8. Please contact MAK 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
Red Hat Enterprise Linux Workstation 7.0 (64-bit libraries)	default compiler
Red Hat Enterprise Linux Workstation 8.0 (64-bit libraries)	default compiler
PC with Windows 10	Microsoft Visual C++ 12.0 (64-bit)
	Microsoft Visual C++ 14.0 (64-bit)

The VC 14 build is binary compatible with all MAK VC 15 products.

VR-Link for C# supports all Windows compilers.

VR-Link for Java supports all compilers and operating systems.

VR-Link for Java requires Java 8.

Disk Space Requirements

Linux installations require approximately 2 GB of disk space. On Windows, you need approximately 1.5 GB of disk space.

Compiler Compatibility on Windows

MAK provides versions of product releases that have been compiled with different versions of Microsoft Visual C++. When you run MAK products together on the same computer, for example, the Logger and a VR-Vantage application, we strongly recommend that you run versions compiled with the same compiler. Mixing products compiled with different versions of the compiler on the same computer can result in program instability. The exception to this is that products built with VC 14 and VC 15 are binary compatible.

License Manager

To run the licensed version of VR-Link, you must install license management software. VR-Link 5.8 uses FLEXlm 11.13.0.2. If you are upgrading from a version of the VR-Link that used an older version of FLEXlm, you must upgrade your license management files. You do not need a new license. Licenses are forward compatible.

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

- ♦ Windows:
<https://vtmakceptstgagt01.blob.core.windows.net/productinstallers/licenseManager/LicenseManager5.0/licenseManager5.0-win64-20230306.exe>
- ♦ Linux:
<https://vtmakceptstgagt01.blob.core.windows.net/productinstallers/licenseManager/LicenseManager5.0/licenseManager5.0-linux64-lsb-20230315.tar.gz>

License manager support is available through the support portal:

<https://www.mak.com/mak-one/support/portal>

Network Compatibility

VR-Link 5.8 is compliant with RPR FOM 1.0 and RPR FOM 2.0 and includes FOM mappers for many draft versions of the RPR FOM. It is compliant with RTIs that support the HLA 1.3 specification, the SISO DLC HLA API 1516 version of the IEEE 1516 specification (SISO-STD-004.1-2004), or HLA Evolved.

To use an RTI with VR-Link, it must use the same operating system and be built with the same compiler.

VR-Link 5.8 supports DIS 4, 5, 6, and 7, and can therefore interoperate with DIS applications of any of these versions.

Backward Compatibility

VR-Link 5.8 has the following backwards compatibility issues:

- VR-Link versions 5.0-5.1.3 encoded environmental process DIS Version 7 PDUs incorrectly. This has been corrected. Unfortunately this means that VR-Link 5.3.1 will not inter-operate with those versions of VR-Link for DIS 7 Environmental Process PDUs. We recommend upgrading all VR-Link 5.0-5.1.3 federates if you are using that specific configuration.
- VR-Link 5.2 encoded data interactions incorrectly for RPR FOM 2.0. This is fixed in VR-Link 5.3.1. However, this incompatibility causes applications that were built with VR-Link 5.2 to crash when trying to decode VR-Link 5.3.1 interactions.

For backwards compatibility, the incorrect encoding is now provided as RPR 2.0 Revision 1. The correct encoding is RPR 2.0 Revision 2. Use Revision 1 to maintain compatibility with previous versions of VR-Link. Use Revision 2 to be compliant with RPR FOM 2.0.

- VR-Link 5.0 renamed header files to conform to MAK file naming conventions. If you have not already migrated to these new file names, the next section describes the changes.

Using the Compatibility Directory

VR-Link has a *compatibility* directory in its *include* directory. This directory has the same directory structure as the main *include* directory with header files that have the same names as previous versions of VR-Link. The header files in this directory do not duplicate the declarations in the renamed header files, they include the renamed header files themselves. To use the *./include/compatibility* directory, add it to your project's include directory list. You will not have to update the include statements in your project files.

RTI Support

VR-Link 5.8 has been tested with the MAK RTI 4.6 for HLA 1.3, HLA 1516, and HLA Evolved.

If you use the MAK 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 *MAK RTI Reference Manual* for a list of the options for specifying the location of the *rid.mtl* file.

Applications built with VR-Link 5.8 for HLA can use any RTI that conforms to the relevant dynamic link compatible HLA standard (1.3, SISO DLC 1516, HLA Evolved) and was built using the same operating system and compiler.

New Features and Changes

VR-Link 5.8 has these new features:

- ♦ During the creation of high-entity-count scenarios with VR-Forces, VR-Link now throttles requests for updates on discovered objects to let the scenario initialize more quickly.
- ♦ VR-Link now supports Earth Gravitational Model (EGM) Geoids. The *DtGeodeticCoord* class can now account for differences in mean sea level between the selected EGM and the WGS84 Reference ellipsoid in order to convert positional data before sending it over the network in accordance with the DIS and RPR specifications.
- ♦ To support new patches in DI-Guy and other features, VR-Link has new default FED files with updates to the default HLA connection settings. The new files:
 - *MAK-RPR2-1-10.fed* for HLA 1.3
 - *MAK-RPR2-1-10.xml* for HLA1516
 - *MAK-DIGuy-7_evolved.xml* FOM module for HLA Evolved
- ♦ You can now switch the VR-Link Code Generator to dark mode. Choose **Configuration > Toggle Dark Mode** to turn dark mode on or off.
- ♦ The MAK License Manager has been updated to FlexLM 11.17.1 to take advantage of security enhancements.

Fixed Bugs

This release fixes the following issues that were problems in previous releases:

- ♦ VRL-201. Added an accessor to get `DtStateRepCreator` from `DtReflectedAggregate` in HLA.
- ♦ VRL-287. The `setTrackJamTarget` function of class *Beam* now behaves as intended, sending the entire array to the VR-Link C++ API.
- ♦ VRL-288. In Java, a first tick call now works if `dispose` is called before.
- ♦ VRL-307. Fixed an issue where C# Euler transforms sometimes returned NaN.
- ♦ VRL-344. Updated the `CharacterPatchesStruct` datatype for the DIGuy-Patches attribute. `CharacterPatchesStruct` is now a struct of 3 `PatchIdentifiers` followed by an HLA ASCII string.
- ♦ VRL-345. Fixed a crash caused by deleting `DtReflectedIffList` before calling `DtExerciseConnection::drainInput`.
- ♦ VRL-346. Fixed intermittent crash when initializing a C# DIS connection.
- ♦ VRL-353. Inactive beams are now properly removed from the emitter system repository.
- ♦ VRL-355. Added the functions `setMaxNumberThreads` and `maxNumberThreads` to the *DtParallelExecutionContainer* class to set the max number of threads for the parallel execution method to create.
- ♦ VRL-363. Fixed issues with embedded system identifiers in VR-Link Java.
- ♦ VRL-365. Fixed a crash in `DtObjPtr` null check.
- ♦ VRL-379. Tracked targets and beams have been fixed in emitter beams in the Java API.
- ♦ VRL-382. VR-Link Java no longer sends unneeded IFF PDU updates when Layer 3 is active.
- ♦ VRL-388. Fixed the IFF strategy setting of ModeC to actually use ModeC instead of ModeS.
- ♦ VRL-393. Updated *disEnums.h* to include new emitter beam function enumerations from the SISO-REF-30 document.
- ♦ VRL-397. Fixed an issue with frozen entities spamming updates while frozen with certain dead reckoning algorithms.