



MÄK Data Logger 5.3 Release Notes

This document provides the following release-specific information:

Systems Supported	2
Logger API	2
Qt Toolkit Compatibility	2
Compiler Compatibility on Windows	2
License Manager	3
ODBC Driver Support	3
Network Compatibility	3
RTI Support	4
Support for HLA Evolved	4
FOM Support	5
New Features	6
Bug Fixes	7
Known Problems	9
Playing HLA 1.3 Logger Files in HLA 1516 Federations	9

Copyright © 2014 VT MÄK, 150 Cambridge Park Drive, 3rd 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®, RTIsby®, B-HAVE®, and VR-Link® are registered trademarks of VT MÄK.

Document ID: LOG-5.3-2-140319

Systems Supported

Table 1 lists the platforms currently supported by MÄK Logger 5.3. Application code must be built with the indicated compilers in order to link to Logger libraries.

Table 1: Platforms supported

Platform	Compiler
Red Hat Enterprise Linux Workstation 5.0	Default compiler (32-bit)
Red Hat Enterprise Linux Workstation 6.0	Default compiler (64-bit)
PC with Windows XP, Vista, Windows 7, Windows 8	Microsoft Visual C++ 8.0
	Microsoft Visual C++ 10.0 (32-bit and 64 bit)
	Microsoft Visual C++ 11.0 (32-bit and 64 bit)



You must be an administrator to install MÄK products on Windows Vista.

Logger API

You must have VR-Link 5.0.1 and its required compiler versions to link the Logger libraries, but you do not need it to run the executables. Please see VR-Link documentation for information about required compilers.

You must have a VR-Link developer's license to compile plug-in or applications using the Logger API.

Qt Toolkit Compatibility

The Logger GUI is built using the Qt Toolkit, a cross-platform GUI toolkit. Logger 5.3 uses Qt 4.7.4. You do not need Qt to do development that does not affect the GUI.

Compiler Compatibility on Windows

Current and prior releases of MÄK products have been compiled using different versions of the Visual C++ compiler. When you run MÄK products together, for example, the Logger and the Stealth, we strongly recommend that you run versions created with the same compiler. Mixing products compiled with different compilers can result in program instability.

License Manager

To run the licensed version of Logger, you must install license management software. Logger 5.3 uses FLEXlm 11.8 for all versions except the Windows VC++ 7.1 version, which continues to use FLEXlm 11.6. If you are upgrading from a version of the MÄK Data Logger 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 Logger installer. You can download them at:

- ♦ Windows:
<ftp://ftp.mak.com/out/MAKLicenseManager-win-setup.exe> (for 32-bit systems)
<ftp://ftp.mak.com/out/MAKLicenseManager-win64-setup.exe>
- ♦ Linux: <ftp://ftp.mak.com/out/MAKLicenseManager-linux-setup.tar.gz> (for 32-bit systems)
<ftp://ftp.mak.com/out/MAKLicenseManager-linux64-setup.tar.gz>

A license manager FAQ is available at:

<http://www.mak.com/support/faq.php#license>

ODBC Driver Support

The Logger database export feature supports the following ODBC drivers:

- ♦ MySQL ODBC 2.50 Driver
- ♦ MySQL ODBC 5.351 Driver
- ♦ MySQL ODBC 5.1 Driver
- ♦ MySQL ODBC 5.2w Driver.
- ♦ Microsoft Access.

For the VC10 64-bit Logger, a 64 bit connector is required.

The MySQL 5.1 server and connector can be downloaded at:

- ♦ server: <http://dev.mysql.com/downloads/mirror.php?id=411970>
- ♦ connector: <http://dev.mysql.com/downloads/mirror.php?id=411736>

Network Compatibility

HLA only

Logger 5.3 was built against VR-Link 5.0.1 and is compliant with:

- ♦ RPR-FOM 1.0 and a subset of 2.0 (draft 6, 14, and 17).
- ♦ MÄK RTI 4.1 or later.

DIS only

Logger 5.3 supports DIS 4, 5, 6, and 7.

RTI Support

We recommend using the latest version of the MÄK RTI. However, the Logger should work with other RTIs that conform to the HLA 1.3 specification, the IEEE 1516 SISO DLC API, or HLA Evolved and are built using the same compiler as your version of the Logger.

When using the MÄK RTI, remember to make sure that the Logger can find your FED file, and optional *rid.mtl*, file by either putting them in the directory from which you are running, or by setting the environment variable the directory that contains them.

Support for HLA Evolved

Logger 5.3 updates the HLA Evolved version to be compatible with MÄK RTI 4.0.4, VR-Link 5.0.1, and the HLA Evolved specification. A full discussion of this issue is as follows:

The HLA Evolved libraries included with the MÄK RTI 4.0 through 4.0.3 were unintentionally built against one of the last draft versions of the HLA 1516-2010 specification header files. Between the version of the header files used by the MÄK RTI and the final version of the specification there was one change. The HLA Evolved API has three functions for createFederationExecution. In the version originally used by the MÄK RTI there was an ambiguity. Due to optional parameters, two of these functions could be invoked in such a way that it was not clear which one was being used. Here are the functions as they used to be:

```
// variation 1
virtual void createFederationExecution (
    std::wstring const & federationExecutionName,
    std::wstring const & fomModule,
    std::wstring const & logicalTimeImplementationName = L"");

// variation 2
virtual void createFederationExecution (
    std::wstring const & federationExecutionName,
    std::vector<std::wstring> const & fomModules,
    std::wstring const & logicalTimeImplementationName = L"");

// variation 3
virtual void createFederationExecution (
    std::wstring const & federationExecutionName,
    std::vector<std::wstring> const & fomModules,
    std::wstring const & mimModule,
    std::wstring const & logicalTimeImplementationName = L"");
```

Variation one is unambiguous. However, consider the following code:

```
createFederationExecution( L"FedExName", vectorOfFomModules, L" ");
```

It is not clear whether you are using variation two or three, and the compiler would throw an error.

To fix this ambiguity, the final version of the HLA Evolved API renamed the third version of the function to `createFederationExecutionWithMIM`. However this revision did not make it into the header files used by the MÄK RTI. RTI 4.0.4 corrects this and uses the final version of the HLA Evolved header files. As a result, we recommend that anyone using the MÄK RTI with HLA Evolved upgrade to this version. Because the API has changed between RTI 4.0.3 and 4.0.4, any HLA Evolved federates built against previous MÄK RTI versions will need to be recompiled against the new version.

Because VR-Link's HLA libraries are built against the MÄK RTI, this issue also exists in previous versions of VR-Link that supported HLA Evolved. We recommend that anyone using VR-Link with HLA Evolved upgrade to version 4.0.3 or later, which is built against the latest RTI and uses the renamed function.

This problem affects MÄK products built against the MÄK RTI and VR-Link that support HLA Evolved, such as Logger 5.0. Logger 5.2 has been built against VR-Link 5.0.1 and is compliant with the final version of the standard.

As a general rule, any version of the Logger is compatible with any version of the RTI that supports the same platform. However, due to this header file change, previous versions of the Logger and the MÄK RTI may not be compatible when using HLA Evolved. This change does not affect compatibility between the Logger and the MÄK RTI when using HLA 1.3 or 1516-2000.

When using HLA Evolved:

- ♦ Logger 5.0 is only compatible with MÄK RTI 4.0, 4.0.1, 4.0.2, and 4.0.3.
- ♦ Logger 5.1 and later releases are only compatible with MÄK RTI 4.0.4 and later releases.

FOM Support

Logger 5.3 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. It also supports VR-Link's ability to support alternative FOMs through the FOM Mapper. By default, Logger 5.3 uses RPR FOM 1.0.

For information about FOM mapping and selecting the correct FOM Configuring the Logger for HLA in Chapter 2, "Installing and Configuring the Logger", in *MÄK Data Logger Users Guide*.

New Features

Logger 5.3 has the following new features:

- ♦ Updated to use VR-Link 5.0.1.
- ♦ Support for DIS 7.
- ♦ The maklanddemo recordings have been improved to work better and renamed makland2013. Recordings have been added for HLA 1516 and HLA Evolved.
- ♦ Changes to the Logger connection information are now saved to the configuration file. This feature can be enabled or disabled by selecting or clearing the Save Connection Configuration check box in the Preferences dialog box, Network Settings tab. Changing this preference is saved to the configuration file. You can also change this setting by editing the `saveConnectionConfiguration` parameter *lgrConfig*.xml* or by sending the `SetSaveConnectionConfiguration` command. For details, please see Section 4.1.2, “Editing a Connection” in *MÄK Data Logger Users Guide*.
- ♦ You can now select multiple contiguous and non contiguous entity IDs in the Filter by Entity ID dialog box. You can also use wild cards to select entity IDs. For details, please see Section 9.5, “Filtering DIS Messages by Entity ID” and Section 9.6, “Filtering HLA Messages by Object Name” in *MÄK Data Logger Users Guide*.
- ♦ A leaf class column was added to the Traffic Analyzer Objects display to easily see the specific part of the object class name.
- ♦ The Logger can now continuously and simultaneously play back a recording that is being recorded by another Logger. In recent releases, the Logger was enhanced to allow it to load files that were incomplete and it provided an option to create a repaired file or just load the incomplete file. This feature allowed the Logger to load a file that was being recorded by another Logger. However, the playback was restricted to the current snapshot of the file at the time it is loaded. This release of the Logger updates this feature to continuously update the end time of the playback to reflect the end time of the ongoing recording.

When the Logger loads a file that is being recorded, it detects that the file is incomplete. It prompts you to repair the file. If you choose to play the file without repair, the Logger loads the file with the current end time. When playback is active, the Logger updates the end time of the file to reflect the end time of the ongoing recording. When you open a file without repair, some features are not available. (For example, the index table that supports efficient jumps is not available and jumps take longer.) It is still possible to create a repaired file from the recording file that will have the EOF meta data. This repaired file will be complete and fully functional. However, it will be a snapshot of the recording at the time of repair. The recording file will not be modified by the playback or repair.

For details, please see Section 7.3.8, “Playing a File While it is Being Recorded” in *MÄK Data Logger Users Guide*.

- File descriptions. You can save a description of a Logger file that can be viewed by anyone who loads the file. For details, please see Section 6.5, “[Viewing and Editing Logger File Information](#)” in *MÄK Data Logger Users Guide*.
- The Logger Remote Control Interface API has been moved into a separate library. This allows the integration of the remote control features into another application without having to include the core Logger infrastructure. There are two new libraries along with an existing utility library. The lgrUtil library contains basic Logger utility classes. The lgrCmds library contains the command factory classes. The lgrRemoteControlInterface library contains the remote control interface classes.
- The Traffic Analyzer can display entity types using string names for the different components (kind, domain, and so on). The mapping of the entity type components to string names can now use the SISO enumeration XML document. For details, please see Section 10.7.4, “[Mapping Entity Type Enumerations to Strings](#)”, in *MÄK Data Logger Users Guide*.
- In the Traffic Analyzer, you can disable the brackets around elements of an array of any specified datatype (such as char). So, an array of characters is displayed as a simple string. To disable brackets, add an entry similar to the following in *.bin/datatypeDefault.mtl*:

```
(arrayBracketMap "element_type" 0)
```

For example:

```
(arrayBracketMap "ArticulatedParameterStruct" 0)
```

For details, please see Section 10.7.5, “[Removing Brackets from the Display of Array Types](#)” in *MÄK Data Logger Users Guide*.

Bug Fixes

This release fixes the following problems:

- Changes to the list of entity types in the entity type filter were not saved to the commonEntityTypes variable in the Logger configuration file. 47899
- The 1516 and 1516 Evolved read file examples crashed when joining a federation created by a 1.3 federate (using a 1.3 FOM). 48757
- In the traffic analyzer, update coloring of recent attribute changes was working intermittently. It has been fixed to color attributes when they change. 48763
- The playback of HLA interactions did not have the correct timestamp values (from the recorded packets). 48782
- In HLA, when attempting to open a Logger file that was recorded using a FOM that does not match the current connection configuration, the Logger will now prompt the user to change connection settings, which the user can ignore or accept. Accepting will disconnect the Logger, change the connection settings to match those found in the Logger file, and then reconnect and open the file. 48843, 51285

- ♦ The *entity-type-dict.csv* file (used in traffic analyzer) was missing from 64-bit installations. 49643
- ♦ The entity ID filter did not remove all filtered entities immediately after the filter was accepted. Previously, entities would only be removed when a packet for the entity was processed. In HLA, this resulted in some static entities that had no updates never being deleted. 50932



Other filters that affect entities and objects (for example, PDU type and class) have not been updated to correct this issue.

- ♦ The Connection dialog box could become hidden behind the Logger window. You can now bring it back to the front by right-clicking the connection icon in the bottom right hand corner of the Logger window. 51222
- ♦ When selecting Bind to Annotation, if the current play time was outside the bounds of the annotation, it did not move to the start of the bound area. Now it automatically moves to the start of the bound area. 51248
- ♦ In some cases in the Traffic Analyzer, blank or badly formed values (for example, entityType or entityID) were corrected by adding variations to capitalization in the *datatypeDefault.mtl* file. Now, if additional blank or badly formatted data is noticed, the Logger inspects the FOM to determine if the basic representation used in the FOM is missing from the *datatypeDefault.mtl* file. If so, it adds an entry with a mapping to the appropriate datatype representation. 51274
- ♦ The recordFile example did not correctly shut down the Logger instance after finishing, causing a memory leak and an HLA error. 51281
- ♦ 51285
- ♦ When a tape with annotations was loaded, the last annotation was always selected. This is no longer the case. No annotations are selected until you specifically choose one. 51298
- ♦ The addition of the makland2013 tapes for HLA 1516 and HLA Evolved has fixed the exportFileToDatabase example for these protocols. 51300
- ♦ The Logger now closes an open Connection dialog box on exit. 51345

Known Problems

This section lists known problems with Logger functionality:

- ♦ The HLA 1516 Logger crashes on exit if you use the MÄK RTI 2.4.2 through 3.1.1. This is due to a problem in the RTI.
- ♦ Logger files that include the VR-Forces embarkation feature in DIS exercises, such as *embarkdemoDIS.lgr*, do not reset embarkation status after time jumps. To correct the embarkation state of entities, stop playback and start playing the file from the beginning.
- ♦ On Linux, online help may not work if the Logger lib directory is not in the LD-LIBRARY_PATH or the RTI *lib* directory is before it.

Playing HLA 1.3 Logger Files in HLA 1516 Federations

In general, the Logger uses the same file format for HLA 1.3 and HLA 1516 Logger files. You can record a Logger file from an HLA 1.3 federation, and play it back into an HLA 1516 federation (and vice versa). However, you must use the same FOM representation during record and playback. For example, if you use an HLA-1.3-style FED file during recording, you must use an HLA-1.3 style FED file during playback, even if you are using the IEEE 1516 version of the Logger and an IEEE 1516 RTI. (VR-Link and the MÄK RTI allow you to use HLA-1.3-style FED files with IEEE-1516-based federates, and vice versa).

The main reason that consistency in FOM format is necessary is that IEEE 1516 uses different names for the "Root" classes of the Object class hierarchies. A 1.3-style FED file requires a Root class a 1516-style XML files requires a Root class called *HLAObjectRoot*. The reason this can be a problem is that if the Logger is playing an HLA-1.3-based Logger file into a federation that is using a 1516-based XML file it might come across an instance of a class called, for example, "ObjectRoot.Vehicle". If it tries to register an object of this class, the RTI will complain that no such class exists. There might be a class called *HLAObjectRoot.Vehicle* in the current FOM, but the RTI or Logger does not know that this is actually the same class.

