



MÄK Data Logger 4.4 Release Notes

This document provides the following release-specific information 4.4:

Systems Supported	2
Logger API	2
Qt Toolkit Compatibility	2
Compiler Compatibility on Windows	2
License Manager.....	3
ODBC Driver Support.....	3
Using Libraries and Binaries Built with Visual Studio 2005 and Later	3
Patch Required for AMD Dual-processor Windows PCs	4
Network Compatibility	4
RTI Support	5
FOM Support.....	5
New Features	5
Support for DDM Recording and Playback.....	6
Configuration Options for Faster Playback.....	6
Additional Connection Options	7
Documentation Updates	7
Bug Fixes	8
Known Problems	9
Playing HLA 1.3 Logger Files in HLA 1516 Federations.....	9

Copyright © 2010 VT MÄK, 68 Moulton St., Cambridge, MA 02138 All rights reserved.
MÄK Technologies®, VR-Forces®, RTIsby®, B-HAVE®, and VR-Link® are registered trademarks of VT MÄK. Document ID: LOG-4.4-2-100920

Systems Supported

Table 1 lists the platforms currently supported by MÄK Logger 4.4. 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 4.0, and 5.0	Default compiler
PC with Windows XP	Microsoft Visual C++ 7.1, 8.0, and 9.0
PC with Windows Vista or Windows 7	Microsoft Visual C++ 8.0 and 9.0



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

Logger API

You must have VR-Link 3.13.2 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 4.4 uses Qt 4.5. 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 4.4 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/flexlm11-8-win-setup.exe>
- Linux: <ftp://ftp.mak.com/out/flexlm11-8-linux.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 3.51
- Microsoft Access.

Using Libraries and Binaries Built with Visual Studio 2005 and Later

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) and later, 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 and later. The 32-bit installer is named *vc redistrib_x86.exe*; the 64-bit installer (if supported) is named *vc redistrib_x64.exe*. They are in the base directory of any installed MÄK product that requires them.

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 who 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>



You must ensure that the preprocessor defines `_SECURE_SCL=0`, and `_HAS_ITERATOR_DEBUGGING=0` are set for MSVC++8.0 and MSVC++9.0 builds. If these are not set, random crashes and assertions may be encountered during runtime.

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 MÄK products to run backwards. This occurs when the Windows scheduler changes the CPU the MÄK 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 customers, 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.

Network Compatibility

HLA only

Logger 4.4 was built against VR-Link 3.13.2 and is compliant with:

- ♦ RPR-FOM 1.0 and a subset of 2.0 (draft 6, 14, and 17)
- ♦ MÄK RTI 3.x.x (please see “[Known Problems](#),” on page 1-9.)

DIS only

Logger 4.4 supports DIS 4, 5, and 6.

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 or the IEEE 1516 SISO DLC API 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.

FOM Support

Logger 4.4 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 4.4 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 User's Guide*.

New Features

Logger 4.4 has the following changes and new features:

- ♦ The *./examples/mkwin32VC7* directory has been removed. Examples now have sample project files in the example's directory.
- ♦ Support for DDM record and playback.
- ♦ Additional connection options.
- ♦ New parameters for faster playback.
- ♦ Ability to specify a remote control ID. For details, please see the `SetRemoteControlId` command in Section 11.4.5, "[Simulation Domain Commands](#)", in *MÄK Data Logger User's Guide*.
- ♦ Ability to specify a maximum time for network reads. For details, please see Section 5.4, "[Configuring Network Read Time](#)", in *MÄK Data Logger User's Guide*.
- ♦ Ability to specify heartbeats for HLA playback and timeouts for HLA recording. For details, please see Section 5.3.4, "[Configuring HLA Heartbeating](#)" and Section 5.3.5, "[Configuring HLA Timeouts](#)", in *MÄK Data Logger User's Guide*.
- ♦ Ability to load damaged files. If a recording terminates abnormally, the Logger can repair and load the file. For details, please see Section 6.5.1, "[Opening Incomplete Files](#)", in *MÄK Data Logger User's Guide*.
- ♦ Window configuration is saved. The Logger saves the size of the Logger window and the configuration of the various data views and restores it in the next session.

- ♦ The *IISec_VPMM_Demo_Final-DIS.lgr* and *VR-Village-CarBombDemo-v3-DIS.lgr* files have been added. *makland-kabulDemoDIS.lgr* is no longer shipped. The *IISec_VPMM_Demo_Final-DIS.lgr* files takes place over a large area of California, USA. If you are viewing it in VR-Forces or VR-Vantage, we recommend using *VR-TheWorld.earth*.

Support for DDM Recording and Playback

The Logger can be configured to record and play back using HLA Data Distribution Management (DDM). There are several configuration parameters that control how the Logger handles DDM. There are two general uses cases for configuring DDM with the Logger. The first case uses the HLA DDM conveyed region information to record and play back the region information. The second case supports creating a plugin to inspect the simulation data in order to create and update regions. The DDM configuration parameters control which of these approaches is supported and they allow some variation in how they are supported. For information about how to configure the Logger DDM behavior, please see Section 5.5, “[Using Data Distribution Management](#)”, in *MÄK Data Logger User’s Guide*.

Configuration Options for Faster Playback

There are some processes that can be modified in order for the Logger to play back tapes faster at the expense of some data or feature loss. In certain cases, the data loss is acceptable and the improved performance is required because of the density of data or packets in the tape. For example in DIS and even some HLA federations, there are entity heartbeats and timeouts. So, dropping some entity state information or missing the removal of an entity can be overcome in time by subsequent state updates or entity timeouts.

Three commands have been added to control fast playback options:

- ♦ SetEntityTracking
- ♦ SetFastJumpsWithoutDiff
- ♦ SetPlaybackReadTimeLimit

For details about configuring faster playback, please see Section 5.3, “[Configuring Faster Playback](#)”, in *MÄK Data Logger User’s Guide*.

Additional Connection Options

The Connection dialog box has been revised. It has the following new options:

- ♦ DIS
 - Network device option. You can select the network device from those available on the machine.
 - Support for IPv6. IPv4 and IPv6 addresses are supported where appropriate. However, all addresses must use the same protocol type.
- ♦ HLA
 - FOM revision option for RPR FOM 2.0017. MÄK has provided several revisions to the RPR FOM 2.0017 FOM Mapper to correct inconsistencies with RPR FOM 2. draft 17. Applications using RPR FOM 2.0017 must use the same revision.
 - Ignore advisories option. Configures the logger to ignore RTI advisory messages and always send data. Default: false.
 - Empty FOM Mapper. In addition to the RPR FOM or custom FOM Mapping choices, there is a new FOM Mapping option to use a Generic FOM or Empty FOM Mapper. The Generic FOM option should be used when the FOM is not a derivative of the RPR FOM and no appropriate FOM Mapper library is available.

For details, please see Chapter 4, [Configuring the Logger Connection](#), in *MÄK Data Logger User's Guide*.

Documentation Updates

MÄK Data Logger User's Guide and the online help have been updated to support the new features.

Bug Fixes

The following bugs (with defect tracking numbers) have been fixed in this release:

- ♦ The Logger's response of sending status messages when it receives a state message has been restored. This behavior had existed in 3.12 but was missing from 4.0-4.3.
- ♦ Linux plugin libraries now all begin with "lib." 12218
- ♦ Recording is not allowed while playing a file to avoid putting the Logger in a bad state that leads to crashes. 12933
- ♦ When loading a Logger file that requires conversion, the entity tracking table is now populated. It is no longer required to first save the file to the new format. 39494, 39761
- ♦ Having a disabled network device caused the Logger to crash. 39572
- ♦ When recording more than 500 entities, closing the Logger file or closing the Logger no longer caused a crash. 40696, 40699
- ♦ Processing of Logger control messages caused the Logger to become unresponsive when switching between playback and record with the same filename. 41120
- ♦ Language translation for filenames did not work. 41571
- ♦ HLA interactions were missing after recording was stopped and started again. 41625
- ♦ In the Logger API, the `DtLgrRecordInterface::isPaused` did not return the correct state when Logger was paused or not paused. 41736
- ♦ Adding entity ID or object name filters when exporting to database caused a crash. 41803
- ♦ Using MySQL Server 5.1, exporting fire PDUs/Interactions did not populate the Fire table. The "Range" column in the table has been renamed to "FireControlSolutionRange." Range is a reserved word in SQL 5.1 and using it as a column name caused the export to fail. 41809
- ♦ If the Logger command line included text beginning with a hyphen (-) that was not a recognized command, it assumed it was a filename and if no such file existed, it gave a misleading error. The Logger will now reject any argument that starts with a hyphen that is not a valid argument. If you have a Logger file whose name starts with a hyphen, you can specify it using the "`-r OpenFile filename`" argument. 42326
- ♦ Badly formatted run commands on the command line result in diagnostic output to the Logger diagnostic file and Logger console if enabled. 42362
- ♦ The `SetRemoteControlSettings recordRemoteControl` parameter, which sets the Logger Control PDU filter, did not work. 42704

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.
- ♦ If you are running the Logger in HLA 1516 with the MÄK RTI, and the RTI parameter `RTI_enableLrcWebService` is enabled, the Logger might pause when starting or stopping recording. This pause can be 30 seconds or longer and the Logger might not be responsive during this time.

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.

