MÄK WebLVC Server 1.3 Release Notes

This release note provides release-specific information for MÄK WebLVC Server Release 1.3.

- System Requirements 2
  - Platforms Supported 2
  - Product Compatibility 2
  - Browser Compatibility 2
  - Third Party Library Dependencies 3
  - Compiler Compatibility on Windows 3
  - License Manager 3
- Network Compatibility 4
- RTI Support 4
  - Support for HLA Evolved 4
- Brokers Provided 6
- New Features and Updates 8
- Documentation Updates 8

Document ID: WVC-1.3-2-170131

System Requirements

This section lists supported platforms and software requirements.

Platforms Supported

MÄK WebLVC Server 1.3 supports the operating systems listed in Table 1.

Table 1: Platforms supported

Platform	Compiler
Red Hat Enterprise Linux 6.0	Default compiler (64-bit)
PC with Windows Vista/Windows 7/ Windows 8/Windows 10	Microsoft Visual C++ 10.0 (64-bit) Microsoft Visual C++ 12.0 (64-bit)

You must have administrator privileges to install MÄK products on Windows Vista.

Product Compatibility

- VR-Link 5.3.1.
- MÄK RTI 4.2. Minimum version required for automatic FOM translation (see Extending WebLVC with automatic FOM translation.)
- VR-Forces 4.4, 4.4.2. Minimum version required for use with Close Air Support WebLVC Suite app.
- VR-Vantage 2.1.

Browser Compatibility

- Chrome. Recent version
- Firefox. Recent version.
- Safari. Recent version.

Third Party Library Dependencies

MÄK WebLVC Server 1.3 uses the following third party libraries:

- QT 4.7.4: VC 10.
- Qt 4.8.7: VC 12 and Red Hat 7.
- Qwt 6.0.1.
- Boost - 1.46.1.
- ICONV 1.8 - VC 10 (http://www.gnu.org/software/libiconv/.)
- ICONV 1.9.2 - VC 12 and Red Hat 7 (http://www.gnu.org/software/libiconv/).
- LIBXML2 2.6.22 (http://www.xmlsoft.org/).
- NodeJS 5.0.0 (http://nodejs.org).
- OpenSSL 1.0.2h (https://www.openssl.org).
- Poco 1.7.2 (https://pocoproject.org).
- JsonCpp 0.5.0 (https://github.com/open-source-parsers/jsoncpp).

Compiler Compatibility on Windows

MÄK provides versions of product releases that have been compiled with Microsoft Visual C++ 10.0, 11.0, 12.0, and 14.0 (most products are not available for all of these compilers). When you run MÄK products together, 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.

License Manager

To run the licensed version of MÄK WebLVC Server, you must install license management software. MÄK WebLVC Server 1.3 uses FLEXlm 11.13.0.2. If you are upgrading from a version of MÄK WebLVC Server 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 MÄK WebLVC Server installer. You can download them at:

- Windows: ftp://ftp.mak.com/out/MAKLicenseManager-win64-setup.exe
- Linux: ftp://ftp.mak.com/out/MAKLicenseManager-linux64-setup.tar.gz

A license manager support is available at:

http://www.mak.com/support/license-support.html

Network Compatibility

HLA only

The MÄK WebLVC Server 1.3 HLA brokers are compliant with:

- RPR-FOM 0.5, 0.7, 0.8, 1.0, and RPR FOM 2.0 (draft 6, 14, 17, 20)
- MÄK RTI 2.x, 3.x, 4.x (HLA Evolved support is only with MÄK RTI 4.0.4 or later)
- Pitch RTI 1.3 C++ interface.
- Pitch RTI HLA Evolved C++ interface.
- Other 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 MÄK WebLVC Server it must use the same operating system and be built with the same compiler.

DIS only

The MÄK WebLVC Server 1.3 DIS broker supports DIS 4, 5, 6, and 7, and can therefore interoperate with DIS applications of any of these versions.

RTI Support

We recommend using the latest version of the MÄK RTI. However, MÄK WebLVC Server 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 MÄK WebLVC Server.

When using the MÄK RTI, remember to make sure that the MÄK WebLVC Server 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

The MÄK WebLVC Server 1.3 HLA Evolved broker is compatible with MÄK RTI 4.0.4 or later, VR-Link 4.0.4 or later, 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. MÄK 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, 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 MÄK WebLVC Server 2.0. MÄK WebLVC Server 2.0.1 has been built against VR-Link 4.0.4 and is now compliant with the final version of the standard.

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

When using HLA Evolved:

- MÄK WebLVC Server 2.0 is only compatible with MÄK RTI 4.0, 4.0.1, 4.0.2, and 4.0.3.
- MÄK WebLVC Server 2.0.1 and later releases are only compatible with MÄK RTI 4.0.4 and later releases.

Brokers Provided

MÄK WebLVC Server 1.3 includes brokers for:

- HLA 1.3
- HLA 1516 (SISO Standard DLC C++ API for IEEE 1516)
- HLA Evolved
- DIS
- WebLVC.

Table 2 lists the translators for each broker.

Table 2: Broker translators

Translator	HLA	DIS	WebLVC
Acknowledge	X	X	
Action Request	X	X	
Action Response	X	X	
Active Sonar	X	X	
Additional Passive Activities	X	X	
Aggregate	X	X	X
Areal Object	X	X	
Collision	X	X	
Comment	X	X	
Create Entity	X	X	
Data	X	X	X
Data Query	X	X	
Designator	X	X	
Detonation	X	X	X
Emitter System	X	X	
Entity State	X	X	
Environment Process	X	X	X
Event Report	X	X	
Fire	X	X	X
Gridded Data	X	X	
IFF	X	X	
Linear Object	X	X	
Logger Control	X	X	

Table 2: Broker translators

Translator	HLA	DIS	WebLVC
Point Object	X	X	
Propulsion Noise	X	X	
Radio Receiver	X	X	
Radio Signal	X	X	X
Radio Transmitter	X	X	X
Remove Entity	X	X	
Repair Complete	X	X	
Repair Response	X	X	
Resupply Cancel	X	X	
Resupply Offer	X	X	
Resupply Received	X	X	
Service Request	X	X	
Set Data	X	X	
Start	X	X	X
Stop	X	X	X
Transfer Control	X	X	X

VR-Exchange can translate environmental processes. However, it cannot convert custom data inside these PDUs. Therefore, it is recommended that you disable environmental process translation if you are using VR-Forces and translating between DIS and HLA.

New Features and Updates

MÄK WebLVC Server 1.3 has the following new features:

- Updated to use VR-Exchange 2.5 and VR-Link 5.3.1.
- New command messages for controlling VR-Forces and VR-Vantage.
- Support for the current releases of VR-Forces and VR-Vantage.
- Added support for radio messages and VR-Forces radio text messages.
- WebLVC has modularized the VR-Forces commands into a plug-in system. This allows you to switch VR-Forces versions without changing WebLVC versions. VR-Forces versions 4.4 and 4.4.2 are supported. See the *WebLVC Users Guide* for details about changing versions.
- Support for Visual C++ 12.
- VR-Forces aggregates, spot reports, and environmental processes are now supported in WebLVC. It also supports for non-VR-F aggregate objects.
- Support for Linux.
- Support for disabling the Web server.
- All WebLVC examples have been cleaned up and updated.
- Added a data interaction translator.
- Full support for Logger commands.
- WebLVC now supports VR-Vantage 2.1 without any extra plug-ins required.
- WebLVC now supports remote control of VR-Vantage through MouseEvent and KeyPress messages.
- Support for HTTPS.
- WebLVC has new options to enable/disable the HTTP Server, the Proxy server, and settings on the location of where individual items. The complete list of new settings are:
  - --httpServer, Enables the Web Server (Default true)
  - --localHttpDirectory, Local Http Directory (Default "../html")
  - --httpProxy, Enable HTTP Proxy (Default true)
  - --localNonProxyDir, Local Non-Proxy Dir (Default "local")
  - --statusPage, Enable Status Page (Default true)

Documentation Updates

WebLVC Users Guide has been updated for this release. A new manual, *WebLVC Specification Reference Manual* introduces the WebLVC specification and MÄK-specific extensions to it.