



MÄK RTI 2.0-ngc Release Notes

This document provides the following release-specific information for MÄK RTI 2.0-ngc:

Overview	2
Supported Systems	2
Backwards Compatibility	2
Network Compatibility	2
Compatibility with the DMSO RTI	3
New Features and Updates	3
Documentation Updates	3
Fixed Bugs	4
Known Problems	4

MÄK RTI 2.0-ngc has been verified by DMSO as fully compliant with the HLA Interface Specification, version 1.3.

Overview

MÄK RTI 2.0-ngc has been verified by DMSO as fully compliant with the HLA Interface Specification, version 1.3. All services are implemented.

Supported Systems

Table 1 lists the platforms supported by the MÄK RTI 2.0-ngc. Please contact MÄK if you need it for another platform. Application code must be built with the indicated compilers in order to link to the MÄK RTI libraries.

Table 1: Platforms supported

Platform	Compiler
SGI IRIX6.5	MipsPro7.30 C++ compiler (available from SGI)
Sun Sparcstation with Solaris2.7 or later	SPARCompiler C++ version 5.2 (available from Sun)
PC with Red Hat Linux 7.2	GNU C++ (g++) distributed with Red Hat distribution. gcc version 3.0.2
PC with Windows NT 4.0, Windows 95/98/2000/XP	Microsoft Visual C++ 6.0.

MÄK RTI 2.0-ngc is built with VR-Link 3.7.3-ngc.

Backwards Compatibility

The MÄK RTI 2.0-ngc libraries are not run-time compatible with previous versions. (In other words, you cannot have a federation execution in which some federates use MÄK RTI 2.0-ngc and some use a previous version of the MÄK RTI.) However, the MÄK RTI 2.0-ngc is link compatible with previous versions, so a federate does not need to be recompiled to use this version. We recommend that all federates use the same version of the RTI when operating within the same federation (or using the same rtiexec).

Network Compatibility

The MÄK RTI is not network compatible with the DMSO RTI. (In general, no two RTI implementations are network compatible, due to differences in the low-level network mechanisms that they use (which include, but are not limited to packet layout.)) Applications that want to interoperate in the same federation execution must all use the same RTI implementation. A federation can easily switch from one RTI implementation to another between runs, but during each run, all participants must agree on which RTI to use, much as they must also agree on which FOM to use.

Compatibility with the DMSO RTI

With the exception of one additional function (`RtReadFileDescriptor()`, described in MÄK RTI documentation), the MÄK RTI 2.0-ngc has the exact same API as the DMSO RTI 1.3 NG (except v5). Unless you are using the `RtReadFileDescriptor` function, an application built against the DMSO RTI NG does not need to be recompiled in order to be used with the MÄK RTI 2.0-ngc, or vice versa.

MÄK RTI 2.0-ngc uses the same FED file format as the DMSO RTI, but a different RID file format. (RID file formats depend on the RTI-implementation.)



MÄK products, including the MÄK RTI 2.0-ngc, are not run-time compatible with the DMSO RTI NG v5 due to changes in the how the DMSO RTI handles namespaces.

New Features and Updates

MÄK RTI 2.0-ngc has the following new features:

- Full support for Time Management, federation save and restore, Data Distribution Management (DDM), and Management object Model (MOM) services.
- The RID file (*rid.mtl*) has been updated and has new parameters to support the full scope of RTI services.

Full Support for RTI Specification

As noted earlier in this document, the MÄK RTI 2.0-ngc is fully compliant with the HLA Interface Specification 1.3. It supports all RTI services.

RID File Updates

The *rid.mtl* file now includes all parameters supported by the MÄK RTI, along with comments explaining their use. It has also been reorganized so that related parameters are grouped together. See *MÄK RTI Reference Manual* for details.

Documentation Updates

The online (PDF) version of *MÄK RTI Reference Manual* and the RTIspy HTML class reference have been updated for this release.

Fixed Bugs

This release fixes the following problems in previous releases:

- ♦ A problem with the tick(*min*, *max*) that caused it to only process a single message has been fixed. It now behaves as intended, attempting to begin processing messages for at least *min* seconds and continuing to process messages (if present) until *max* seconds have elapsed. While processing of the last message is guaranteed to have begun before the *max* time has elapsed, the total processing time may exceed *max* time.

Known Problems

This release has the following known problems:

- ♦ The RTI can sometimes fail to throw an "ObjectAlreadyRegistered" exception if a federate tries to register an object with a name that is already being used by another object. The federate can register an object using a name that belongs to another object that is unknown to the local RTI component (LRC). The LRC only maintains the names of objects that are registered or discovered by the federate. While we plan to fix this problem, a good preventive measure is to always subscribe and tick the RTI before registering objects.
- ♦ If you are using the Linux version of the MÄK RTI Spy with versions of MÄK products earlier than the most recent releases, the MÄK products may crash. If this happens, contact MÄK technical support and we will provide you with a fix for your application.