Rational Rose RealTime Release Notes

Contents

Chapter 1

Introduction

Thank you for selecting **Rational Rose RealTime**, real-time industry's leading environment that unifies software teams by integrating with best-in-class Rational products including Rational ClearCase, Rational RequisitePro and Rational SoDA.

These release notes describe:

- "PLEASE READ FIRST" on page 2
- "Platform requirements — Windows NT" on page 5
- "Licensing Requirements" on page 9
- "Integration Notes" on page 11
- "Known Problems and Limitations" on page 15
- "Documentation Updates" on page 25
- "Technical Support" on page 27

Please read these Release Notes before you install or use Rational Rose RealTime.

Note: *In some cases, you may note discrepancies between the printed documentation and the online documentation. In these cases, the online documentation is the most up-to-date and should be followed.*

Please visit the Rational Web site for the latest Release Notes, patches and information:

http://www.rational.com/support

If you encounter any problems while installing or running Rose RealTime, please check here first to see if you encountered a known problem. If you find a problem that is not yet documented, please call Rational Technical Support so we can investigate it, provide you with a workaround, and track the problem for further action.

Contact information for this release is in "Technical Support" on page 27.

PLEASE READ FIRST

Please read the following notes before you install Rose RealTime.

You Must Install License Keys to Run this Product!

With your software shipment, you received an envelope containing your Start-up License Key Certificates for this product. You need these keys in order to run your software. The start-up keys will expire a few weeks after shipment. Be sure to request your permanent license keys as soon as they are available. The availability date for your permanent keys is indicated in the top section of your Start-up Licence Key Certificate.

Please check the *Rational Rose Realtime Install Guide* for detailed instructions on how to install your start-up license keys and how to request your permanent license keys.

Note: *If you are installing license keys on a Unix platform, please refer to the Install Guide for Instructions. **Do not** follow the instructions on the Rational Start-up License Certificate or on the envelope in which the certificate is delivered to you.*

Notice To Beta Customers

IF YOU WERE A BETA CUSTOMER FOR RATIONAL ROSE REALTIME, PLEASE UNINSTALL BOTH THE WINDOWS AND UNIX COMPONENTS OF THE BETA VERSION FROM YOUR SYSTEM BEFORE BEGINNING YOUR INSTALLATION.

Notice To Customers Running Earlier Rose RealTime releases

You cannot run earlier versions of Rose RealTime on Windows NT or Windows 2000 with the latest version of Rose RealTime.

Minimal Install not supported

A minimal install is not supported for Rational Rose RealTime as a stand alone product. A minimal install for the Rational Suites on Windows is supported, though.

Chapter 2

Platform and Toolchain Requirements

This section describes the platform and toolchain requirements for running Rose RealTime.

Platform requirements — Windows NT

The minimum supported configuration for running Rose RealTime on Windows NT is:

- Windows NT 4.0 with service pack 5 or 6a
- Minimum Pentium 150 MHz or faster CPU
- Minimum 64 MB of RAM; we recommend 128 MB RAM
- Minimum 65 MB of disk space for the Rose RealTime installation
- Minimum 800 * 600 * 256 color video resolution or better
- Browser recommendation — Internet Explorer 4 or 5

Platform requirements — Windows 2000

The minimum supported configuration for running Rose RealTime on Windows 2000 is:

- Windows 2000 Professional
- Minimum Pentium 150 MHz or faster CPU
- Minimum 64 MB of RAM; we recommend 128 MB RAM
- Minimum 65 MB of disk space for the Rose RealTime installation
- Minimum 800 * 600 * 256 color video resolution or better
- Browser recommendation — Internet Explorer 4 or 5

Platform requirements — Unix

The minimum supported configuration for running Rose RealTime on Solaris is:

- Solaris 2.5.1, Solaris 2.6, 2.7, or HPUX 10.20
  - ❑ For Solaris operation, we recommend an UltraSparc 1 or greater
  - ❑ For HPUX operation, we support installation of the HP 700 series architecture
  - ❑ Please see the Rational Rose RealTime website (http://www.rational.com/support) for a list of the required Unix patches applicable to your operating system.
- We recommend 128 MB of RAM
- Minimum 250 MB of disk space for the Rose RealTime installation

Toolchain requirements

Help Viewer (Windows platforms only)

The Help Viewer requires that Microsoft Internet Explorer (version 3.02 or later) be set up on a user's computer. It is not required that Internet Explorer be used as the system's default browser, or that the Internet Explorer icon be visible on the user's desktop.

If you choose not to have Internet Explorer as the default browser, you will need to run Hhupd.exe (in redist). This file is the distribution executable that installs the run-time components needed for an HTML Help Project, such as Hh.exe, Hhctrl.ocx, Itss.dll, and Itircl.dll. Hhupd.exe is in the Redist folder of the HTML Help Workshop folder.

Compiler

You must have a C++ compiler installed on your system to make use of the code generation and execution capabilities for Rose RealTime. Different compilers are required for host workstation and for embedded system targets. The list of supported compilers and targets is provided in the *Getting Started Guide for Rational Rose RealTime.*

Real-time Operating System

If you are planning to deploy your model on a real-time operating system, your operating system, hardware and tool lineup must be one of the supported lineups listed in the *Getting Started Guide*. If you do not have a supported lineup, you may be able to get support for your lineup from a Rational RoseLink partner, or by customizing the Rose RealTime Services Library for your target. See the *C++ Guide* or *C Guide* for instructions on customizing the Services Library and compiling for new target platforms.

Chapter 3

Licensing Requirements

Please refer to the *Install Guide* for instructions on "Installing a Startup License" and "Obtaining a Permanent License"

Note: *If you are installing licenses on a Unix platform, **do not** follow the instructions on the Rational Start-up License Certificate or on the envelope in which the certificate is delivered to you.*

If you have existing licenses for release 6.1 or 6.1.1, you do not need new licenses. Please save your license file (license.dat) before uninstalling.

License Usage

A toolset launched manually will require a license.

A second toolset session launched manually on the same platform will require a second license.

A toolset launched as a COM Automation server will NOT require a license unless it is explicitly made visible by setting the RRTEI Application object's Visible property to True. For more information on RRTEI, see the *RRTEI Reference Guide*.

A toolset launched with a compiled script passed as a command line argument will require a license since the toolset will be made visible before running the script.

A toolset launched with both the -runScriptAndQuit and a compiled script passed as a command line argument will NOT require a license since the toolset will not be made visible when running the script.

Note: *A toolset session, regardless of how it is initiated, will require a license if the toolset is made visible.*

Chapter 4

Integration Notes

This section describes integrating Rose RealTime with other tools.

Migrating from Rational Rose and ObjecTime Developer

In order to migrate models into Rose RealTime from either Rational Rose or ObjecTime Developer where models were previously stored in a configuration management system, the model must be brought into the Rational Rose or the ObjecTime Developer tool and written out to a single file. Please refer to Migration in the *Getting Started Guide*.

When importing a model from Rose into Rose RealTime, you are encouraged to resolve any model errors in Rose (**Tools > Check Model**) before trying to import the model. In particular it is important to fix unresolved references. In general, Rose is not concerned with unresolved references; however, they are very important in Rose RealTime as they can result in incomplete code generation and compilation errors.

In order to export the ObjecTime model in a format that is readable by Rose RealTime, a patch must be applied to the 5.2 or 5.2.1 toolset to format the file in a single linear form file with all the required information. The patch is available from Rational Customer Support for both the 5.2 and 5.2.1 product release only. Please contact the Rational Customer Support group for further information.

Once the model has been imported into Rose RealTime, it can then be stored in the configuration management system.

Microsoft development environment

We recommend that you install the latest service packs available from Microsoft for Visual Studio or Visual C++.

Configuration for operation with ClearCase

For more information on integration, refer to the *Team Development Guide*.

Integration with Rational Robot

Installing the 2001.03.00 release of Rational Rose RealTime will interfere with the operation of the 6.1 release of Rational Robot.

We recommend that you upgrade to the 6.2 release of Rational Robot.

ClearCase on a UNIX server and clients on both NT and UNIX

You can access a ClearCase server on UNIX with Rose RealTime clients running on both NT and UNIX workstations. For more information on integrating these tools, refer to the *Team Development Guide*.

SoDA

Both SoDA and Rose RealTime must be properly installed and licensed before starting. Note that SoDA may be installed either as part of a suite or as an individual product. See Rational SoDA for Word in *Getting Started with Rational Rose RealTime.*

Please refer to the product support page at

http://www.rational.com/support

for the latest updates on SoDA integration.

Note: *In order to generate a report using SoDA, the Rose RealTime model must have been saved at least once. If the Rose RealTime model has never been saved, it will be untitled. An untitled model will cause SoDA to generate errors.*

RequisitePro

Both RequisitePro and Rose RealTime must be properly installed and licensed before starting. Note that RequisitePro may be installed either as part of a suite or as an individual product. See RequisitePro in *Getting Started with Rational Rose RealTime.*

Please refer to the product support page at

http://www.rational.com/support

for the latest updates on RequisitePro integration.

Note: *The Rose RealTime Requisite Pro integration does not support the association of a Rose RealTime package with a RequisitePro project. Use Case and Model association is supported.*

Purify

Adding options to Purify on UNIX

Occasionally, you may need to add options during a Purify'd build on Unix. For example, Purify on HP needs to know the name of the linker or collector used by **Gnu g++**.

Options can be added by changing **PURIFY_OPTIONS** in the **CompilationMakeInsert** field of the executable component.

The default value of **PURIFY_OPTIONS** (generated in the Makefile by the code generator) is:

PURIFY_OPTIONS = -log-file=$(BUILD_TARGET).txt -windows=no

To accommodate using **g++** on HP, you can add the following, or similar:

> **PURIFY_OPTIONS = -log-file=$(BUILD_TARGET).txt -windows=no -collector=/usr/lib/gcc-ld -g++=yes**
>
> Where the path of to the collector, **gcc-ld** in most cases, should be the path that is specific to your environment.

For proper integration of Purify when running the Purify'd executable from the toolset, you should preserve the default options.

For an explanation of Purify options, see *Running a component instance with Purify* in the *Toolset Guide.*

Chapter 5

Known Problems and Limitations

The following is a list of the problems and limitations in the Rose RealTime product at the time of release. Some of these problems may have been addressed by the time you read this. For the most recent list of known problems and any fixes that may have been published, please visit the Rose RealTime web site at:

http://www.rational.com/support

File association for compiled scripts

Windows NT: Rose RealTime does not install a file association for compiled scripts (.ebx). This means that they cannot be automatically run by double-clicking on the file from Windows Explorer.

Probes

When a message hit causes a probe to halt the execution of a running model, there is no way of finding out which probe in the model causes the model to stop executing. To determine which probe execution is halted on, open trace windows for each enabled probe.

Start-up problems

If Rose RealTime has problems starting up, look at the task bar and use the Task Manager to see if there are running copies of Rose RealTime. Terminating all running copies will allow new copies of the tool to start properly.

Note that more than one copy of Rose RealTime can be run at the same time; however, if startup problems exist, find and terminate any runaway processes.

Uninstall

Sometimes files are left behind after an Uninstall. For example, if a model has been saved in one of the Rose RealTime subdirectories, the subdirectory and its parents will not be removed. You must remove these manually if you wish to return your system to a clean state.

License manager

When there is no License manager running, a pop-up window appears indicating that the toolset cannot obtain a license.

Changes to code that uses default arguments

ObjecTime Developer models which used the RTTimespec constructor with only one parameter, as in the following code:

```
timer.informIn(RTTimespec(2));
```

will result in a compile error after conversion of the model to Rational Rose RealTime. The compile error will appear something like:

```
..\rtg\Driver.cpp(67) : error C2440: 'type cast' : cannot
convert from 'const int' to 'struct RTTimespec'
No constructor could take the source type, or constructor
overload resolution was ambiguous.
```

The reason is that in ObjecTime Developer, the RTTimespec constructor included default arguments, that is, RTTimespec (long=0, long=0). The default constructor values are not supported on RTTimespec in Rose RealTime. Any code that made use of the default arguments needs to be changed to supply both constructor arguments. For example:

```
OTD = RTTimespec (2);
```

must be changed to:

```
RRT = RTTimespec (2, 0);
```

Spaces in directory names

On Windows NT, in order to allow the use of cross-compilers that do not allow spaces in the path names, you need to use the `subst` command and map a drive to the value of %ROSERT_HOME% after the installation. For example, if you wish to use the K: drive, and your Rose RealTime installation directory is:

```
ROSERT_HOME=C:\Program Files\Rational\Rose RealTime
```

you must map this directory to the drive by running the following commands from a console window:

```
subst K: "%ROSERT_HOME%"
set ROSERT_HOME=K:
```

Spaces in directory names can cause problems with the following *operating system.compiler library set.development platform*: systems:

OSE411T.ppc603-Diab-4.1a.NT4

VRTX4T.ppc603-Microtec-1.4.NT40

TORNADO101T.m68040-cygnus-2.7.2-960126.NT40

TORNADO101T.ppc-cygnus-2.7.2-960126.NT40

TORNADO101T.x86-cygnus-2.7.2-960126.NT40

TORNADO2T.m68040-cygnus-2.7.2-960126.NT40

TORNADO2T.ppc-cygnus-2.7.2-960126.NT40

TORNADO2T.m68040-cygnus-2.7.2-960126.NT40

For Tornado line-ups, you should also use the substituted drive to point to the load script directory when running Target Observability.

On the detail tab of the processor specification, the `Load Script` path should not contain any spaces. If spaces are present, when attempting to load the executable in Basic or Debugger-Tornado[2] modes, the error message, **Unable to Execute** appears. This does not occur when loading the executable in manual mode.

Spaces in directory names can also cause problems with ClearMake.

Too many files open

You can have a maximum of 1024 file pointers open simultaneously.

Executing component instances

If you are getting the message "Unable to connect to target" when trying to connect to a target (both host and embedded), change the Connect Delay on the Component Instance specification to two or more seconds.

Sequence diagrams

There still remain a few conditions under which the Sequence Diagram will incorrectly draw Messages / FOCs. If a Message or FOC appears to be incorrectly drawn, select the Message (or the Message that starts the FOC) and, using the center 'reorient' handle, slightly move the Message. This will force the Diagram to recalculate the correct display values for that message.

Case sensitivity within paths

The Unix temporary directory name is translated to all lower case. If you set the environment variable TEMP, make sure the path name is all lower case or the directory will not be found. This will cause problems when Help is started.

Error loading large models from ClearCase

If you have problems loading large models from ClearCase —that is, it takes a long time or an empty model appears—check your error messages. It may be because there are too many files open. Use the following steps to resolve this problem:

1. Go to the Control Panel, and double click on the Clearcase applet.
   Clearcase Properties dialog appears.
2. Go to the MVFS Performance tab.
3. Under Mnodes, click the Override boxes for both "Maximun number of mnodes to keep on the free list" and "Maximum number of mnodes to keep for cleartext free list", and set the value for both to "800".
4. Click **OK** to apply the changes and close the dialog.
5. Reboot the Windows NT machine.

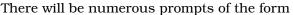## Build dependencies on case-insensitive file systems (NT only)

During a build, Rose RealTime detects and records build dependencies for comparison during subsequent builds; this is done to facilitate build-avoidance by only regenerating or recompiling targets when a build dependency has changed. These build dependencies preserve the case of the filenames involved, even when the underlying file-system (e.g. NTFS) is case-insensitive. This can cause trouble when using names which are distinct within the toolset, and distinct on case-sensitive drives, but indistinct on case-insensitive drives. In most cases, the toolset or code-generator will identify and quietly avoid or warn against potential case-insensitive name collisions.
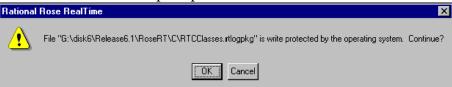
However, some case-insensitive filename collisions cannot be detected. For example, if a component is renamed to something that is case-insensitive-identical (for example, renaming from "myComponent" to "MyComponent"), a build may incorrectly reuse all previous build results, since the underlying build dependencies will be indistinct according to the file-system.

Users are advised to use case-sensitive file-systems if possible. Otherwise, users should avoid case-insensitive name collisions when they create or rename classes, components or controllable units.

RequisitePro

When the RequisitePro add-in is enabled in Rose RealTime, there is a known problem with **File > Save As**.

There will be numerous prompts of the form

as RequisitePro attempts to save the units of RTClasses and RTCClasses. The number of prompts will be less if only one of C or C++ is enabled. Clicking **OK** to all the prompts causes no harm. Other possible workarounds are disabling the RequisitePro add-in for the **Save As** (make sure the model is saved up-to-date before doing this) or removing the RT packages if they aren't required at this time.

Please refer to the product support page at

http://www.rational.com/support

for the latest updates on RequisitePro integration.

SoDA

SoDA can be installed without Rose 2000 and will still function with Rose RealTime. However, when run from Rose RealTime, the menu of templates presented are the Rose templates, not the Rose RealTime ones. Some of these templates will issue errors.

Please refer to the product support page at

http://www.rational.com/support

for the latest updates on SoDA integration.

Note that in order to install the Rose RealTime domain for SoDA, you need to select the Full/Customer install option during your SoDA installation.

Problems accessing Rose help when concurrently running Rose RealTime on Windows NT

Problem

Running both Rose RealTime and Rose may cause the latter not to be able to access its help files.

Explanation

In order to allow Rose add-ins' help to be accessible when used from within the Rose RealTime toolset, Rose RealTime temporarily replaces a registry key value installed by Rational Rose. This registry key is located in [HKEY_LOCAL_MACHINE\Software\Rational Software\Rose] and is named "HelpFileDir".

When the toolset starts up, it substitutes this key's value by its own Help File directory. The value installed by Rational Rose installation program is backed up and restored when the last instance of the running toolset exits.

Workaround

You can disable this behavior by setting the value of the key located in [HKEY_LOCAL_MACHINE\Software\Rational Software\Rose RealTime\6.1] whose name is "ReplaceRoseHelpDir" to "No".

UNIX issues

Refresh problems with Exceed

The screen sometimes does not refresh completely when running the Unix version of Rose RealTime and displaying it on a PC using Hummingbird Exceed. The problem can be fixed by changing your Exceed settings. Start Xconfig and open the Performance dialog. Use the following settings: Save Unders = No, Maximum Backing Store = When Mapped, Default Backing Store = None, Minimum Backing Store = None.

Context-sensitive help on Unix may not invoke the correct help page

Context-sensitive help on Unix may not invoke the correct help page. The workaround is to use the advanced search capabilities of the online-help to find the appropriate topic.

Cannot open some links with online help

Some links to multimedia content do not work on Unix.

The online help browser on Unix does not support hyperlinks to external files. This affects the links to the readme file and to the example models. The workaround is to open the readme file with a regular text editor and to open the examples directly with Rose RealTime.

Toolset freezing on exit

If the toolset freezes on exit, you will need to manually kill the Rose RealTime process to free up CPU and memory resources allocated to the defunct toolset session.

Under certain configurations, when starting Rose RealTime on Solaris and displaying to an exceed X-server, the "Welcome to ..." window appears momentarily, then disappears. After this, the main application GUI is locked out and freezes.

The workaround is as follows:

1. Kill the Rose RealTime process.
2. Edit the **RoseRT.ini** file in the **~/.registry.2001.03.00** directory and change **ShowStartupDialog=Yes** to **ShowStartupDialog=No**
3. Run *RoseRT -cleanup.*
4. Run *RoseRT*

HP-UX Make dependency filename restriction

On the HP-UX platform, the HP-UX Make restricts dependency filenames to 160 characters. We recommend the use of Gnu-Make to work around this problem. Alternatively, you may shorten the path names to your class files to avoid this issue.

Using the Debugger-xxgdb tool and running your component instance

When using the xxgdb tool you should run your component instance, add any breakpoints and then restart it to enable the breakpoints in gdb.

Running multiple sessions on the same workstation

If you are experiencing instability when running multiple Rose RealTime sessions on the same workstation, please try to limit your usage to a single session at any given time.

Non-GUI-based external editors

On Unix, the toolset will freeze when the user specifies "/bin/vi" as the external editor and then tries to launch the external editor from the code edit pane. If you are using a non-GUI-based external editor, this should be specified using "xterm -e /bin/vi" so that the editor has a terminal (tty) to display to.

Executing 6.1.1 and 6.2 toolsets together

The 6.1.1 and 6.2 toolsets do not work together by default, due to sharing the *MWRPC_ENDPOINT* and *MWSHM_KEY* environment variable settings. These settings are only calculated if the environment does not already have them set.

To execute a 6.1.1 and a 6.2 toolset from the same host or shell:

1. Invoke the first toolset with the **-env** parameter (and **&** to have it run in the background)

2. Find the *MWRPC_ENDPOINT* environment variable and note the value, for example, **10633**

3. Set the *MWRPC_ENDPOINT* environment variable for the second toolset to be one greater than the previous:

   setenv MWRPC_ENDPOINT 10634

   MWRPC_ENDPOINT values must be between 10000 and 65000 to be valid.

4. Set the MWSHM_KEY to the same value as the MWRPC_ENDPOINT

   setenv MWSHM_KEY 10634

   MWSHM_KEY environment variable values must be integers less than 1048576, to be valid. We recommend that you use the RPC value for convenience.

5. Invoke the second toolset with the **-env** parameter to confirm environment settings.

Use of C and C++ Add-ins

You cannot use C and C++ Add-ins at the same time in Rational Rose RealTime.
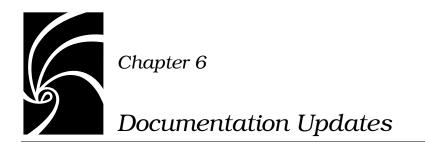
Symbolic Links with TargetRTS

When using LynxOS 3.1.0, do not install Visual Lynx 3.1.0 for Windows NT on a network (NFS) disk. It should only be installed on a local NTFS drive; otherwise, symbolic links to some **include** directories will not work properly. This will cause compilation errors if you re-compile the TargetRTS.

If you have installed Visual Lynx 3.1.0 on a network disk, and if you see compilation errors stating that **include** directories *netinet*/*in.h* or *net*/*if.h* are not found, locate the entries **net** and **netinet** in the following directory:

$LYNX_HOME/usr/lynx/3.1.0/ppc/usr/include

If these entries are text files containing the following text:
!<symlink>bsd

rename these files and create new symbolic links called '**net**' and '**netinet**' that both point to the directory *bsd*.

Chapter 6

Documentation Updates

For the most recent documentation updates please visit the Product Support section of the Rational Rose RealTime website at:

http://www.rational.com/support

Chapter 7

Technical Support

When contacting Rational Technical Support, please be prepared to supply the following information:

- Name, telephone number, and company name
- Product name and version number
- Operating system and version number (for example, Windows NT 4.0, Solaris 2.5.1/2.6/2.7, or HP-UX 10.20)
- Computer make and model
- Your case id (if you're calling about a previously reported problem)
- A summary description of the problem, related errors, and how it was made to occur

If your organization has a designated, on-site support person, please try to contact that person before contacting Rational Technical Support.

You can obtain technical assistance by sending electronic mail to the appropriate e-mail address. Electronic mail is acknowledged immediately and is usually answered within one working day of its arrival at Rational. When sending an email place "Rational Rose RealTime" in the subject line, and in the body of your message include a description of your problem.

When sending email concerning a previously-reported problem, please include in the subject field: "CaseID: v0XXXXX", where XXXXX is the caseid number of the issue. For example:

```
CaseID: v0176528 New data on rational rose realtime install issue
```

Sometimes Rational technical support engineers will ask you to fax information to help them diagnose problems. You can also report a technical problem by fax if you prefer. Please mark faxes "Attention: Technical Support" and add your fax number to the information requested above.

Telephone and fax numbers for Rational Technical Support are contained in the following table. If you have problems or questions regarding licensing, please see "License support contact information" on page 29.

Table 1   Support Telephone and Fax

Region	Telephone Number	Fax Number
Americas	800-433-5444	408-863-4300
Asia Pacific (includes support for Japan, China, India, Korea, Taiwan)	+61-2-9419-0111	+61 2 9419 0123
Europe, Middle East, and Africa (includes support for Israel)	31 (0)20 4546 200	+31 23 569 4302
Other worldwide locations	408-863-5000	

Email addresses for Rational Technical Support are listed in the following table.

Table 2   Support Email

Region	Email Address
Americas and other worldwide locations	support@rational.com
Asia Pacific (includes support for Japan, China, India, Korea, Taiwan)	support@apac.rational.com
Europe, Middle East, Africa (includes support for Israel), and Scandinavia	support@europe.rational.com

License support contact information

If you have a problem or questions regarding the licensing of your Rational Software products, please contact the Licensing Support office nearest you.

Telephone numbers for license support are listed in the following table. Ask for, or select, Licensing Support.

Table 3   License Support Telephone and Fax

Region	Telephone Number	Fax Number
Americas	800-433-5444	781-676-2510
Europe, Israel, and Africa	+31 23 554 10 62	+31 23 554 10 69
North Asia Pacific (Mainland China, Hong Kong, Taiwan)	+852 2143 6382	+852 2143 6018
Korea	+82 2 556 9420	+82 2 556 9426
South Asia Pacific Australia, New Zealand, Malaysia, Singapore, Indonesia, Thailand, The Philippines, Vietnam, Guam and India	+612 9419 0100	+612 9419 0160
Japan	+81 3 5423 3611	+81 3 5423 3622

Email addresses for license support are listed in the following table.

Table 4   License Support Email

Region	Email Address
Americas	lic_americas@rational.com
Europe, Israel, and Africa	lic_europe@rational.com
North Asia PacificMainland China, Hong Kong, Taiwan, and Korea	lic_apac@rational.com
South Asia Pacific Australia, New Zealand, Malaysia, Singapore, Indonesia, Thailand, The Philippines, Vietnam, Guam and India	lic_apac@rational.com
Japan	lic_japan@rational.com