Telelogic DOORS

Using Telelogic DOORS for Rose Interface

IBM

Telelogic DOORS for Rose Interface

Using Telelogic DOORS for Rose Interface

Release 2.10

Before using this information, be sure to read the general information under the "Notices" chapter on page 33.

This edition applies to **VERSION 2.10**, **Telelogic DOORS for Rose Interface** and to all subsequent releases and modifications until otherwise indicated in new editions.

Table of contents

1 About this manual

Welcome to Telelogic® DOORS for Rose Interface™.

Telelogic DOORS for Rose Interface transfers data between Telelogic® DOORS® (DOORS) and Rational Rose® (Rose). It lets you add traceability to your software design process.

This manual describes how to use version 2.10 of Telelogic DOORS for Rose Interface. It assumes that you know how to use both DOORS and Rose.

Typographical conventions

The following typographical conventions are used in this manual:

Typeface or Symbol	Meaning
Bold	Important items, and items that you can select, including buttons and menus. For example: Click **Yes** to continue.
Italics	Book titles
`Courier`	Commands, files, and directories; computer output. For example: Edit your `.properties` file.
>	A menu choice. For example: Select **File > Open**. This means select the File menu, then select the Open command from it.

Related documentation

The following table describes where to find information in the documentation set:

For information on	See
How to use DOORS	The Telelogic DOORS documentation set
How to use Rose	The Rose documentation set
What's new in version 2.10 of Telelogic DOORS for Rose Interface	The Telelogic DOORS for Rose Interface readme file

For information on	See
How to install Telelogic DOORS for Rose Interface	*Telelogic DOORS Installation Guide*

You'll find PDF versions of the DOORS manuals on:

- The Telelogic Lifecycle Solutions DVD
- The support website at https://www.support.telelogic.com

2 Concepts

This chapter introduces and explains the following background concepts:

- About Telelogic DOORS for Rose Interface
- About surrogate modules
- Creating traceability
- Keeping surrogate modules updated
- Developing model elements in DOORS
- Rearranging surrogate module data

About Telelogic DOORS for Rose Interface

DOORS is a powerful tool for creating, structuring and managing complex information, such as the requirements and use case scenarios associated with software development projects.

Rose is an object-oriented modeling tool that supports the analysis and design of software. Telelogic DOORS for Rose Interface lets you transfer data between DOORS and Rose, allowing you to add traceability to your software design process.

With Telelogic DOORS for Rose Interface, you can:

- Manage your software development projects by establishing traceability between requirements and software design.
- Assess how requirements impact your software design.
- Identify requirements that have not been addressed by the software design.
- Find design elements that are not justified by requirements.
- Find out which requirements are associated with a specific element of the software design, and vice versa.
- Verify and demonstrate that your design meets your customer requirements.

About surrogate modules

Telelogic DOORS for Rose Interface lets you send elements in a Rose model to a DOORS project.

Telelogic DOORS for Rose Interface creates a DOORS formal **surrogate module** to store the Rose data. The surrogate module is an anchor point in DOORS that lets you link the requirements in DOORS to the software elements in Rose.

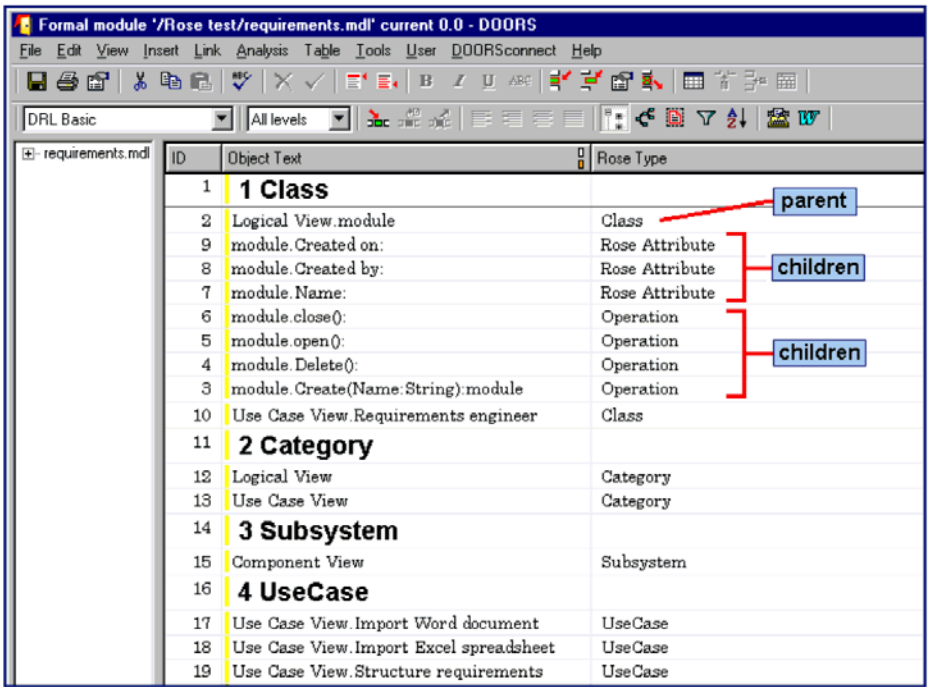
The surrogate module has the same name as the Rose model.

For each element you send, Telelogic DOORS for Rose Interface automatically creates an object in the surrogate module. The new object has the following DOORS attributes:

DOORS attribute	Value
Deleted in Rose?	This is used to keep track of whether the element has been deleted from the Rose model. When you first send the element to DOORS, it is set to **False**. If you subsequently delete the element from the Rose model, it is set to **True** the next time you update the data in DOORS, so that you can see what it was linked to (see "Keeping surrogate modules updated," on page 6).
Documentation from Rose	If you send the documentation associated with the element in the Rose model, it is stored in this attribute.
External Documents from Rose	The location of any attachments to the Rose Model File.
Object Short Text	The name of the element in the Rose model.
Object Text	The section and name of the element in the Rose model.
Read only in Rose	This attribute is **True** for read-only Rose elements, and **False** for elements that can be modified.
Rose Element Identifier	A unique identifier assigned by Rose to the element in the Rose model.
Rose Model File	The location of the Rose model.
Rose Type	The type of the element in the Rose model, for example **UseCase**, **Class** or **Category**.

DOORS attribute	Value
Stereotype from Rose	The stereotype of the element in the Rose model.

The objects in the surrogate module are organized in a hierarchy based on the type of the model element, as shown in the following screenshot:



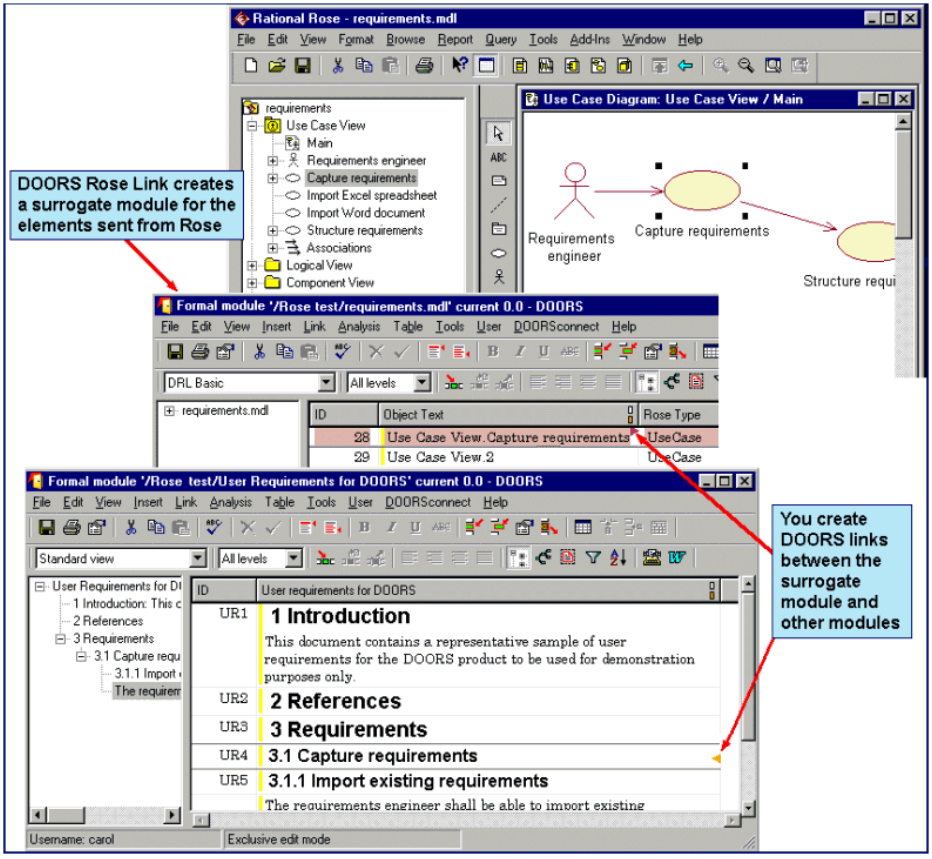
All of the classes are grouped under the heading **Class**, all of the categories are grouped under the heading **Category**, and so forth. When you send class elements, you can also send the operations and attributes associated with the class. Each operation and attribute becomes a child of the class object.

Creating traceability

You use a surrogate module in the same way as all other DOORS modules. You can link its objects to objects in other modules, you can create your own views and you can exploit the power of DOORS traceability tools.

The following screenshot shows how the surrogate module acts like a bridge between the Rose model and the DOORS project. In this example, the **Capture**

Requirements use case in Rose is linked, via the surrogate module, to requirements in the **User Requirements for DOORS** module:



With Telelogic DOORS for Rose Interface, you can apply standard DOORS traceability functions, such as impact and trace analysis, to elements in Rose models. For example, you can find out which requirements are impacted by a particular element or set of elements in your Rose model, or you can find out if there are any requirements which are not linked to elements in your Rose model.

Keeping surrogate modules updated

You use DOORS to store your requirements, and Rose to store your software model.

When you work on requirements, you use DOORS. When you work on your software model, you use Rose.

As you develop your software model in Rose, you should periodically feed the changes you make in Rose back to the surrogate module, to keep it up to date. To update, refer to "Updating the surrogate module," on page 22.

If you delete an element from the Rose model, the next time you update the surrogate module, the **Deleted in Rose?** attribute of the corresponding DOORS object is set to **True**. This lets you identify the requirements that are linked to deleted model elements so you can link them to valid model elements.

Note In earlier versions of Telelogic DOORS for Rose Interface, we used the terms **synchronizing** and **performing consistency checks** to mean updating the surrogate module.

Developing model elements in DOORS

You can use DOORS to develop model elements and then send them to Rose.

For example, you may decide to generate your use cases in DOORS while you're developing the requirements for your project. You may find it helpful to create links between the requirements and use cases as you develop them.

Note Telelogic DOORS for Rose Interface provides a template to help you develop use cases in DOORS. To access the template, restore the module archive `use_case_template.dma` in your `Telelogic\DOORS 6\training` directory.

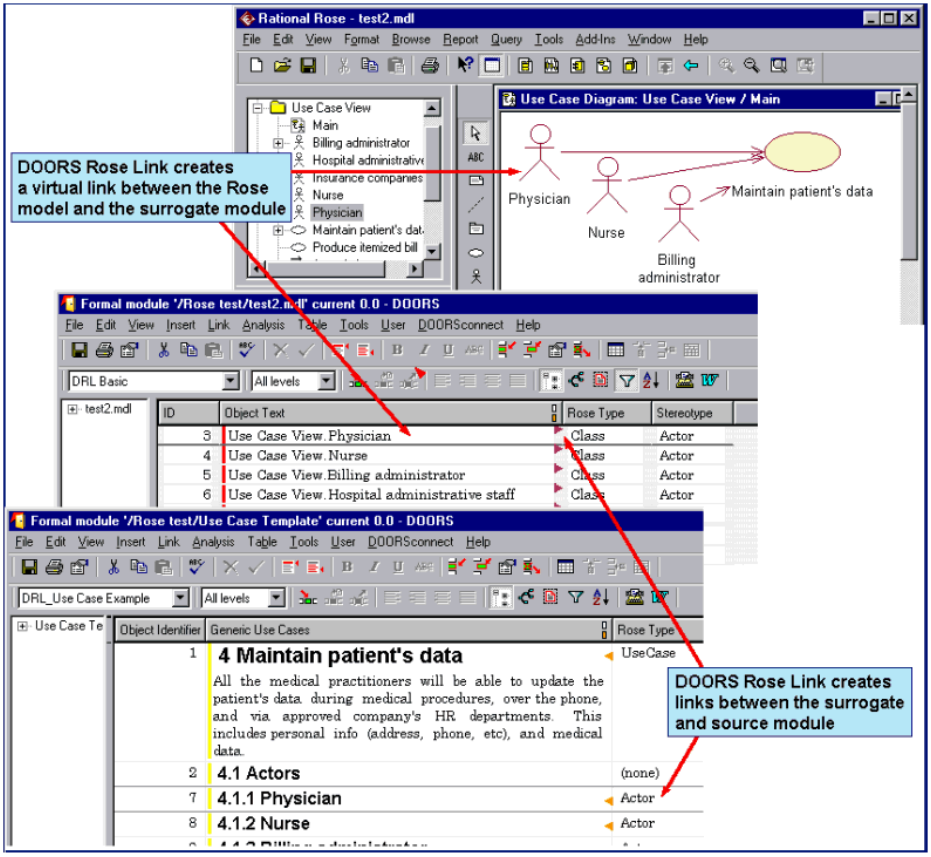
When you are ready to transfer the use cases to your Rose model, use Telelogic DOORS for Rose Interface to send the data from DOORS to Rose. Each object you send becomes an element in the target Rose model.

The new element has the following properties:

Rose property	Value
Name	The DOORS object heading. If the object does not have a heading, the object identifier is used.
Documentation	The DOORS object text.

Telelogic DOORS for Rose Interface automatically sends the data straight back to DOORS, creating a surrogate module in DOORS which contains an object for each object you sent. It creates a link between the object in the original

DOORS module and the corresponding object in the DOORS surrogate module, as shown in the following screenshot:



As you develop your software model in Rose, you should periodically feed the changes you make in Rose back to the surrogate module, to update it (see "Updating the surrogate module," on page 22).

Note Telelogic DOORS for Rose Interface does not update the original DOORS module; it only updates the surrogate module. If you want to keep the original module up to date, you must do so manually.

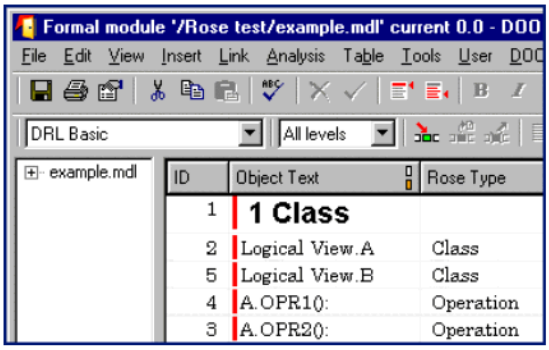
Rearranging surrogate module data

By default, the data in surrogate modules is arranged so that each class operation is located below the last sub-class that inherited the operation.

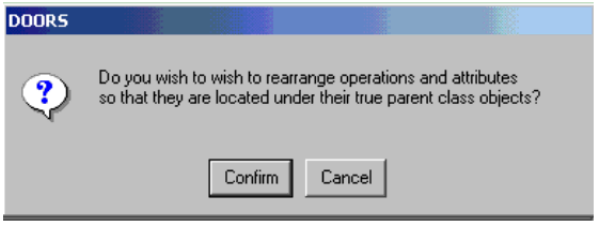
For example, there are two classes A and B, where:

- B is a sub-class of A
- A has two operations, OPR1 and OPR2
- B inherits both operations

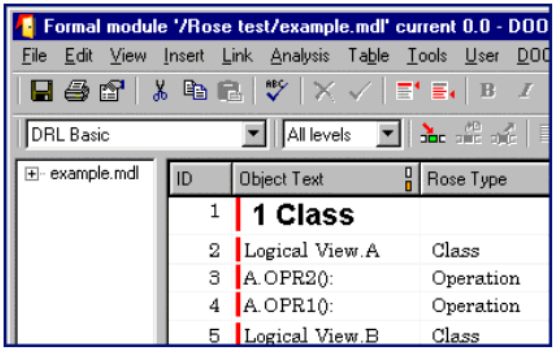
In the DOORS surrogate module, the operations are displayed below the B class heading, as follows:



When sending data from DOORS to Rose, or updating a surrogate module, you can rearrange the data so that each operation is located under its original base class. The following dialog box is displayed:



In this example, the operations are moved below the A class heading, as follows:



3 Using Telelogic DOORS for Rose Interface

This chapter describes how to use:

- Starting and stopping Telelogic DOORS for Rose Interface sessions
- Sending data from DOORS to Rose
- Navigating from DOORS to Rose
- Linking directly to DOORS Object IDs
- Sending data from Rose to DOORS
- Working with read-only Rose elements
- Updating the surrogate module
- Navigating from Rose to DOORS
- Linking to items in a Rose diagram
- Linking to items in a Rose browser tree
- Enabling and disabling the rearrange data option

Starting and stopping Telelogic DOORS for Rose Interface sessions

Before you can use Telelogic DOORS for Rose Interface to transfer data or navigate between DOORS and Rose, you must start a Telelogic DOORS for Rose Interface session. You can start a session on either Windows® or UNIX®.

Follow the steps below according you the platform you are using.

To start a Telelogic DOORS for Rose Interface session on Windows:

1. Start Rose.
2. Log in to DOORS.

You are ready to start transferring data between Rose and DOORS.

To start a Telelogic DOORS for Rose Interface session on UNIX:

1. Log in to DOORS.
2. From DOORS Explorer or from a module window, click **DOORSConnect > Rose > Start Session**.

   Telelogic DOORS for Rose Interface starts Rose, then opens up a communications channel between DOORS and Rose. This communications

channel lets you transfer data between the currently selected folder or project in DOORS and the model currently open in Rose.

Your Telelogic DOORS for Rose Interface session is automatically ended when you exit from DOORS. You can also end your session by clicking **DOORSConnect > Rose > Stop Session**, from DOORS Explorer or a module window.

Note Ending a session shuts down the communications channel between DOORS and Rose. It does not shut down DOORS or Rose.

Sending data from DOORS to Rose

You can send data from one DOORS module to more than one Rose model. A single DOORS project or folder can contain multiple surrogate modules. If the surrogate module does not already exist, you must have **create** access to its project or folder.

To send data from DOORS to Rose:

1. Using Rose, open the target model.
2. Using DOORS, open the module that contains the data you want to send to Rose. If you have previously sent data from this DOORS module to Rose, or are using the `use_case_template.dma` file to create Rose Items, go to Step 4.
3. Click **DOORSConnect > Rose > Setup Module**.

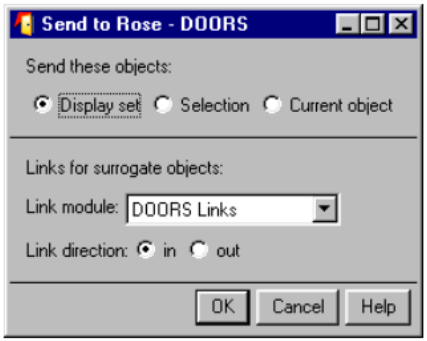
   In the source module, Telelogic DOORS for Rose Interface:

   - Creates a new attribute called **Rose Type**.
   - Creates a new view, which is identical to the previous view except that it includes the new **Rose Type** column.

     The name of the new view is **DRL_viewname**, where **viewname** is the name of the previous view.
4. Using the new view of the module, set the **Rose Type** attribute of every object you want to send to Rose. For example, for use cases, set the **Rose Type** attribute to **UseCase**.

   The **Rose Type** value tells DOORS Rose what type of element to create in the target Rose model.

5. Using the new view of the module, click **DOORSConnect > Rose > Export Items to Rose**. The following dialog box is displayed:



6. Click one of the **Send these objects** radio buttons to specify which objects you want to send to Rose.

   These radio buttons are described in the following table:

Radio button	Description
Display set	Send all the objects in the current view.
Selection	Send only the objects that are currently selected.
Current object	Send only the current object.

   Note You can only send objects that have a value for their **Rose Type** attribute.

7. Use the drop-down list to select the link module you want to use for the links between the current (source) module and the surrogate module.

   Use the **link direction** radio button to specify the direction of the links. Select **in** if you want the links to go from the surrogate module to the current module, otherwise select **out**.

8. Click **OK**.

Telelogic DOORS for Rose Interface:

- Sends the specified objects to the Rose model you opened in Step 2. It displays a DOORS message telling you how many objects were sent. Click **OK** to acknowledge the message.

  Each object becomes an element in the Rose model. The object heading is used as the name of the element (if it doesn't have a heading, the DOORS

object identifier is used instead), and its object text is used as its documentation.

- Creates a surrogate DOORS module for the Rose model, if the surrogate module does not already exist.

  The surrogate module has the same name as the Rose model. For example, if the Rose model is called `test.mdl`, the surrogate module is called `test.mdl`. If the name of the Rose model includes any characters that are not allowed in DOORS module names, these characters are omitted in the DOORS module name. The surrogate module has the following views:

This view	Contains these columns
DRL Basic	ID (Object Identifier) Object Text (Name of Rose element) Rose Type Stereotype
DRL Details	ID (Object Identifier) Object Text (Name of Rose element) Rose Type Stereotype Documentation
DRL Deleted elements	ID (Object Identifier) Object Text (Name of Rose element) Rose Type Stereotype Deleted in Rose?
No Parameters	Identifier Model (Name of Rose element) Rose Type

- Sends the data back to the surrogate module (as if you had selected all the elements you just sent, in Rose, and then sent them back to DOORS). This creates a virtual link between the surrogate module and the model.
- Creates a link between each object in the source DOORS module and the corresponding element in the surrogate module.

- Opens the **DRL Basic** view of the surrogate module, with a filter that shows only the data you just sent.

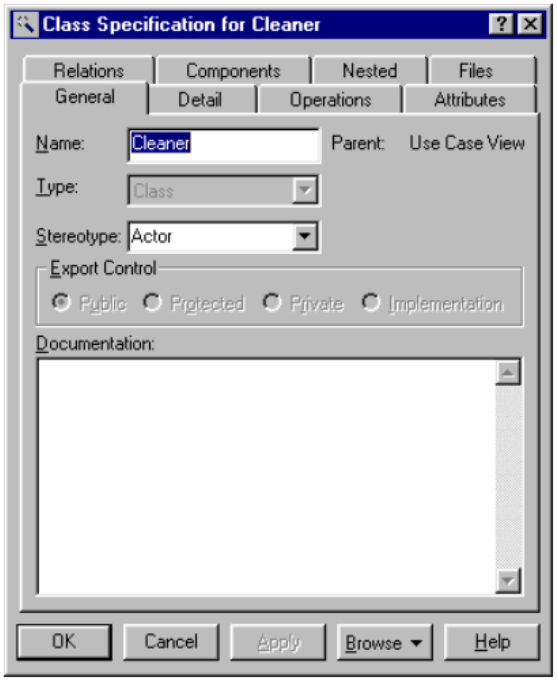
Note It is important that you do not change the name of the surrogate DOORS module, the Rose model or any of the DOORS attributes that Telelogic DOORS for Rose Interface creates, such as the **Rose Type** attribute. If you change these names, subsequent updates will not work properly.

Navigating from DOORS to Rose

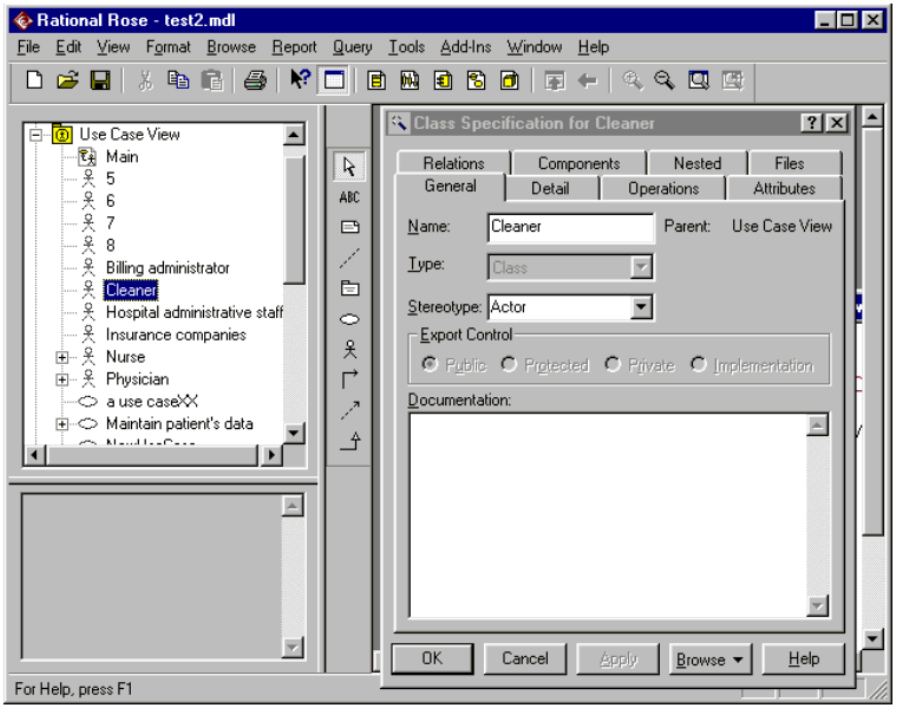
To navigate from an object in a DOORS surrogate module to the corresponding model element in Rose:

1. Using DOORS, open the surrogate module.
2. Using DOORS, select the object in the surrogate module that you want to navigate to in Rose.
3. Using DOORS, click **DOORSConnect > Rose > Find Item in Rose**.

   The following Specification window is displayed in Rose, for the corresponding element in Rose:



4. Using Rose, click **Browse > Select in Browser** to show the element highlighted in the Rose Browser, as follows:



Linking directly to DOORS Object IDs

When you export items that have a Rose Type of either **Class** or **Actor** to Rose, an **Object Identifier** attribute is created. This attribute stores the following information about the object in DOORS with which the Rose element is associated:

- The absolute path to the module containing the object
- The unique module identifier
- The DOORS Object ID

An example **Object Identifier** is `/myproject/mymodule-00000010-SR2`, where `/myproject/mymodule` is the absolute path to the module, `00000010` is the unique identifier of the module and `SR2` is the Object ID of the object in the DOORS module.

Note The unique identifier of a module is automatically generated by DOORS when the module is created.

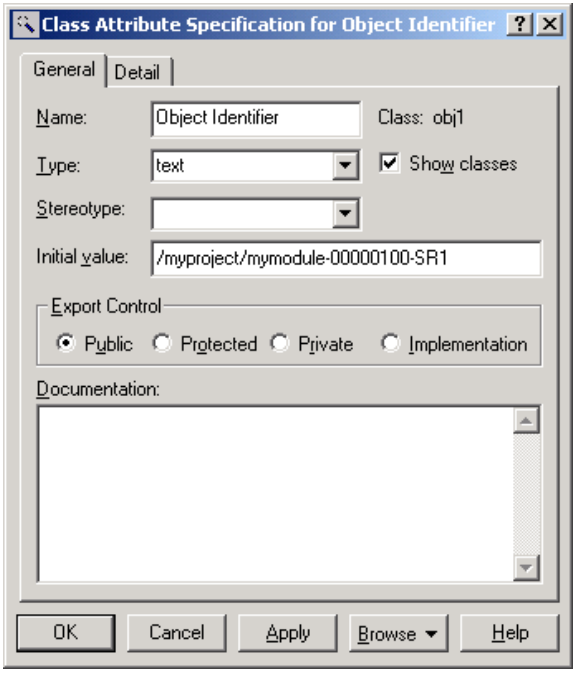
The Object Identifier is stored in the DOORS surrogate modules and is displayed in all of the default views in a column titled Target Identifier. It is also stored as an attribute below the Rose element.

You can associate the Rose element with up to 50 DOORS objects by adding the module path and the DOORS Object ID to the **Documentation** field of the **Class Attribute Specification for Object Identifier** dialog. The next time you send data from Rose to DOORS the surrogate module is updated and links are created from the surrogate module to the object in the DOORS module you specified in the Documentation field.

To associate a Rose element with a DOORS object:

1. In Rose, double-click the Object Identifier.

   The **Class Attribute Specification for Object Identifier** dialog is displayed.



2. In the **Initial Value** field, type the path to the module, including the module name and the DOORS Object ID of the object that you want to associate. Separate each entry with a comma.

   For example, add `,myproject/mymodule-SR4,myproject/mymodule-SR5` to the end of the text that is currently displayed in the Initial Value field.

3. Click OK.

4. Select **Tools > DOORS > Send to DOORS** or **Tools > DOORS > Update DOORS**.

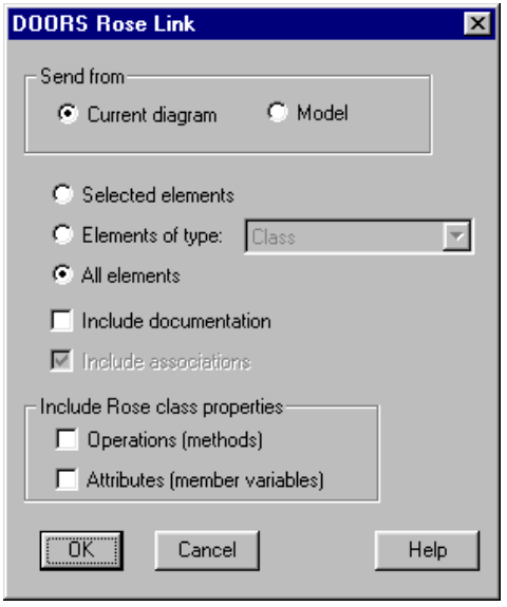
   Links are created in DOORS between the appropriate objects in DOORS.

   In the above example, two new links are created between the appropriate object in the surrogate module and Object ID **SR4** in **mymodule**, and Object ID **SR5** in **mymodule**.

Sending data from Rose to DOORS

To send data from Rose to DOORS:

1. Using Rose, open the model that contains the data you want to send.
2. Select **Tools > DOORS > Send to DOORS**.



3. Use the radio buttons to specify which elements you want to send to DOORS.

   The radio buttons are described in the following table:

Radio button	Description
Current diagram	Only send elements in the current Rose Diagram.
Model	Send all elements in the current model.

Radio button	Description
Selected elements	Only send the selected elements in the current diagram or model.
Elements of type	Send all elements of the specified type in the current diagram or model. Use the drop-down list box to select the type.
All elements	Send all the elements in the current diagram or model.

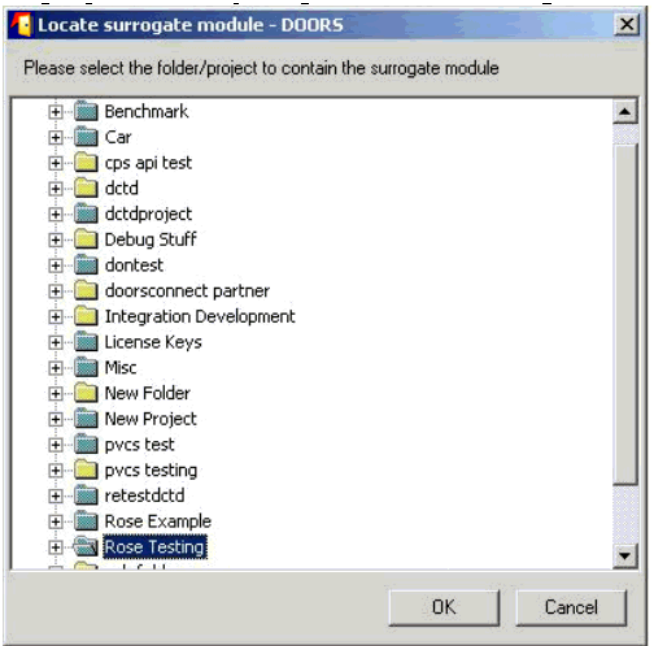
Note If you have a read-only element that you want to send to DOORS, make it writable when you send it, then reset its state to read-only. This ensures that it is automatically updated in the future. For more information, see "Working with read-only Rose elements," on page 22.

In addition, select the following Send options:

- To send the documentation property with each element, check the **Include documentation** box.
- To send elements from the current diagram with associations, check the **Include associations** box.
- To send elements of type **Class**, use the **Include Rose class properties** check boxes to specify whether you want to send the operations and attributes associated with the classes.

4. Click **OK**.

If this is the first time that the Rose model has been sent to the DOORS module, the following dialog is displayed.



5. Select the folder or project to which you want to send the Rose model. If you are sending data for the first time, you need to have create access to this project or folder.
6. Click **OK**.

   Telelogic DOORS for Rose Interface then:

   - Creates a surrogate DOORS module for the Rose model, if the module does not already exist.

     The surrogate module has the same name as the Rose model. For example, if the Rose model is called `test.mdl`, the surrogate module is called `test.mdl`. If the name of the Rose model includes any characters that are not allowed in DOORS module names, these characters are

ignored and do not appear in the DOORS module name. The surrogate module has the following views:

This view	Contains these columns
DRL Basic	ID (Object Identifier) Object Text (Name of Rose element) Rose Type Stereotype
DRL Details	ID (Object Identifier) Object Text (Name of Rose element) Rose Type Stereotype Documentation
DRL Deleted elements	ID (Object Identifier) Object Text (Name of Rose element) Rose Type Stereotype Deleted in Rose?
No Parameters	Identifier Model (Name of Rose element) Rose Type

- Sends the data to the surrogate module, creating a new object for each item sent.
- Displays a DOORS message listing how many surrogate objects were created. Click **OK** to acknowledge the message.
- Displays a message asking whether you want to rearrange the data so that each operation is located under its original base class in the DOORS surrogate module. Click **Yes** or **No** as appropriate.
- Opens the **DRL Basic** view of the surrogate module, with a filter showing only the data you just sent.

Note Do not change the name of the surrogate DOORS module, the Rose model or any of the DOORS attributes that Telelogic DOORS for Rose Interface creates, such as the

Rose Type attribute. If you change these names, subsequent updates will not work properly.

Working with read-only Rose elements

If you have a read-only element that you want to send to DOORS, and there is a possibility that the model element might change in the future, make it writable when you send it, then reset it to read only. This ensures that it is automatically updated in the future, even if it is read only at the time of the update.

If you send a read-only Rose element to DOORS, Telelogic DOORS for Rose Interface cannot mark the element in Rose as sent because the element is not writable. As a result, in subsequent updates, the corresponding surrogate object will not be automatically updated.

To update the surrogate object, manually select the object in Rose and send it to DOORS again. You can also send other elements to DOORS at the same time.

If the model element is now writable in Rose, when you send it to DOORS, Telelogic DOORS for Rose Interface automatically marks it in Rose as sent. All future updates will now automatically update the corresponding surrogate object in DOORS, together with any documentation and class properties that were sent with the writable element. Future updates now work even if the element is read-only in Rose at the time of the update.

Updating the surrogate module

To update a surrogate module to feed changes made in Rose back to DOORS:

1. Using Rose, open the appropriate model.
2. Using Rose, click **Tools > DOORS > Update DOORS**.

Telelogic DOORS for Rose Interface:

- Updates all the data previously sent to the surrogate module to reflect changes made in the Rose model, provided that the data wasn't read-only in Rose when it was sent to DOORS (see "Working with read-only Rose elements," on page 22).

  If an element has been deleted from the Rose model since the surrogate module was last updated, the **Deleted in Rose?** attribute of the corresponding surrogate object is set to **True**.

- Displays a DOORS message telling you how many modifications were made to surrogate objects, and how many surrogate objects have been deleted

from the Rose model since the last update. For example, if you've changed the name and documentation of a Rose element since the last update, the message says that two modifications were made. Click **OK** to acknowledge the message.

- Displays a message asking whether you want to rearrange the data so that each operation is located under its original base class in the DOORS surrogate module. Click **Yes** or **No** as appropriate.
- Opens the **DRL Basic** view of the surrogate module, with a filter showing only the data that was just updated.

Navigating from Rose to DOORS

To navigate from elements in Rose to the corresponding objects in the DOORS surrogate module:

1. Using Rose, open the appropriate Rose model and select the elements you want to find in DOORS.
2. Using Rose, click **Tools > DOORS > Find in DOORS**.

   Telelogic DOORS for Rose Interface opens the surrogate DOORS module, with a filter showing the corresponding DOORS objects.

Linking to items in a Rose diagram

To link to one or more items in a Rose diagram:

1. Select one or more items from your Rose diagram.
2. Select a DOORS requirement object in DOORS.
3. Using DOORS, click **DOORSConnect > Rose > Link to item(s) in Rose Diagram**.

   Your selected Rose items are imported into the local Rose surrogate module, and a link is created between the DOORS object and the imported objects.

Linking to items in a Rose browser tree

To link to one or more items in a Rose browser tree:

1. Select one or more items from your Rose component browser tree.
2. Select a DOORS requirement object in DOORS.
3. Using DOORS, click **DOORSConnect > Rose > Link to item(s) in Rose browser tree**.

Your selected Rose items are imported into the local Rose surrogate module, and a link is created between the DOORS object and the imported objects.

Enabling and disabling the rearrange data option

Processing surrogate modules

By default, the data in surrogate modules is arranged so that each class operation is located below the last sub class that inherited the operation.

When you either send data from Rose to DOORS or update a surrogate module, you see a message that asks if you want to rearrange the data so that each operation is located under its original base class in the DOORS surrogate module. This message is displayed after the data has been sent to DOORS and you have closed the DRL metrics window.

For example, say you have two classes A and B, where:

- B is a sub class of A.
- A has two operations OPR1 and OPR2.
- B inherits OPR1.

In the DOORS surrogate module, by default the operation OPR1 is displayed below the B class heading as A.OPR1, as follows:

1. Class	
LogicalView::A	Class
LogicalView::A::OPR2	Operation
LogicalView::B	Class
LogicalView::A::OPR1	Operation

If you rearrange the data, LogicalView::A::OPR1 is moved to below the A class heading, as follows:

1. Class	
LogicalView::A	Class
LogicalView::A::OPR2	Operation
LogicalView::A::OPR1	Operation
LogicalView::B	Class

Disabling the rearrange data option

By default, the data in surrogate modules is arranged so that each class operation is located below the last sub-class that inherited the operation.

When sending data from DOORS to Rose, or updating a surrogate module, you can choose to rearrange the data, so that each operation is located under its original base class.

If you want to disable the rearrange data option, edit the file `drlpostprocessing.inc` in the folder `lib\dxl\addins\doorsconnect\rose` in the DOORS home directory.

Comment out the following lines by putting two forward slash (`//`) characters at the start of each line:

```
makeColumns (m)
//setup "no parameters" view which removes
//operation parameters
rearrangeOperationsAndAttributes (m)
```

4 Troubleshooting

This chapter describes how to troubleshoot:

- If your DOORS menu options are missing in Rose
- If you get a "suite objects.dll not found" error
- If associations are not updated
- If Rose does not does not exit properly
- If you see two DOORSConnect menus in DOORS

If your DOORS menu options are missing in Rose

If you re-install Rose after you have installed Telelogic DOORS for Rose Interface, the DOORS menu options will disappear from your Rose Tools menu.

To fix this, re-install Telelogic DOORS for Rose Interface.

If you get a "suite objects.dll not found" error

When you are installing Rose on Windows platforms, you may get a message saying that you must update the PATH variable. If you ignore that message, when you click **Start Session** to start a Telelogic DOORS for Rose Interface session, you will get a "suite objects.dll not found" error.

To fix this problem, update the PATH variable. Edit your `autoexec.bat` file to add the `\rational\common` folder to your PATH. For example, add: `"c:\program files\rational\common"`.

If associations are not updated

When you update the surrogate module in DOORS, associations are not automatically updated when you use the **Update DOORS** menu option in Rose.

Use the **Send to DOORS** menu option in Rose instead. You can do one of the following:

- Explicitly select the associations in Rose, and send **selected elements** to DOORS.
- Send the **current diagram** to DOORS; this updates everything, including the associations in the current diagram.

If Rose does not does not exit properly

If you allow DOORS to start the Rose session then you must exit DOORS before Rose can be exited in the normal fashion.

If you see two DOORSConnect menus in DOORS

When you upgrade Telelogic DOORS for Rose Interface, a second **DOORSConnect** menu might sometimes appear in DOORS.

Follow the steps below to fix this problem:

To remove one, do the following:

1. Navigate to the `$DOORSHOME\lib\dxl\addins` directory.
2. Open the `addins.idx` file in a text editor.

   An example of this file is shown below:

   ```
   userU _ User
   doorsconnectD _ DOORSConnect
   doorsconnectD _ DOORSConnect
   ```

3. Delete the third line and save the file.
4. Restart DOORS.

5 Contacting support

This chapter contains the following topics:

- Contacting IBM Rational Software Support
- What to do before you contact Support
- Sending an automated problem report form
- Other information

Contacting IBM Rational Software Support

Support and information for Telelogic products is currently being transitioned from the Telelogic Support site to the IBM Rational Software Support site. During this transition phase, your product support location depends on your customer history.

Product support

- If you are a heritage customer, meaning you were a Telelogic customer prior to November 1, 2008, please visit the DOORS Support Web site on https://support.telelogic.com/doors.

  Telelogic customers will be redirected automatically to the IBM Rational Software Support site after the product information has been migrated.
- If you are a new Rational customer, meaning you did not have Telelogic-licensed products prior to November 1, 2008, please visit the IBM Rational Software Support site on http://www.ibm.com/software/rational/support/.

What to do before you contact Support

If your site has a designated on-site support person, please contact that person before you contact our Support team.

To help our Support team solve your problem, please have the following information available:

- Your name, title, company name, e-mail address, fax number and telephone number.
- Your support ID and support password.
- The version and build number of DOORS that you're running.

To get this information, run DOORS, and click **Help > About DOORS**.

- The operating system you're running DOORS on, for example Windows XP.
- What operating system your DOORS database is running on, if different.
- If you are reporting a new problem, please have a clear statement of the problem, including the exact text of any error messages produced by DOORS, your operating system, or any other tools that were running when the problem occurred.
- If you are calling about a problem you reported earlier, you need the original tracking number the Support team assigned to your problem.
- If you want, you can use the automated problem report form. For information, see "Sending an automated problem report form," on page 30.

The support center on our web site is at https://support.telelogic.com.

Sending an automated problem report form

To send an automated problem report to Support:

1. Select **Help > Generate Support Request**.

   The **Telelogic Support Information** dialog box is displayed with some of your product information automatically included.

   Review the information to make sure it's accurate.
2. From the **Impact** drop down list box, select the severity of the problem.
3. In the **Summary** box, summarize the problem.
4. In the **Problem** box, type a detailed description of the problem.
5. If available, attach a snapshot.

   Click either **DOORS Window Snapshot** or **Screen Snapshot**, whichever is applicable, and select the snapshot from your machine.
6. If possible, use the buttons in the **Attachment Information** area to add video capture, system details and files.

   Note The **Add Product Files** button is unavailable in DOORS. Product information is collected and added automatically.

   Add any relevant information to help Support resolve the problem.
7. You can either:

- Display the information you've entered in a new window so that you can copy it. For example, you may want to add this information to someone else's information.

  Click **Just Text (No Email)**.

- Open the email to edit it before sending it to Support.

  Click **Preview and Send** to submit the report.

Guidelines for writing a problem report

- Be as specific as possible when you summarize the problem in the **Summary** box and when you explain the situation and provide details in the **Problem** box. For example, "The system crashed when I tried to add an attribute" is more helpful than "It crashed."
- Indicate if there were any system changes, such as customizations or upgrades, before the problem occurred.
- If the problem is reproducible, list the specific steps to be followed in order to demonstrate the problem and also indicate the model you are using to perform the steps.
- In the problem description, include anything different or unusual that you observed before the problem happened.
- Make screen captures of anything that you feel will help and attach them to the problem report.
- Include any error messages and code samples you have related to the problem.
- If you have multiple unrelated questions or issues, please submit them separately.

Automatic responses and recording defects

When you send the online form to Support, the customer service system immediately searches the Knowledge Base based on the **Summary** and **Problem** descriptions you entered. If there is an exact match of your problem in the knowledge base, the system automatically sends an email to you with a pointer to the most likely solution.

The problem is also automatically recorded in the defect tracking system as assigned to a support representative. This representative works with you to be certain that your problem is solved. The defect tracking system also records new problems with their solutions in the Knowledge Base to provide rapid assistance for other customers.

Automatically generated problem reports

If your DOORS system crashes, it displays a message asking if you want to send a problem report about the crash to DOORS Support.

If you decide to send this generated report, the system displays the same online form that is available from **Help > Generate Support Request**. In this case, the form contains information about the crash condition, in addition to the information that is usually filled in describing your system.

Add any more information that you can to help the Support staff identify the problem before you click the **Preview and Send** button.

Changing the email address of the problem report

If you want to send the problem to someone other than DOORS Support, for example to your internal support, you can change the default email address of the problem report.

To change the email address of the problem report:

1. Open **System** in the **Control Panel**.
2. On the **Advanced** tab, click **Environment Variables**, then click **New** under **variables**.

   The **New User Variable** window is displayed
3. In **Variable Name** enter **ILX_RECIP**, the variable that controls the address used by the problem report.
4. In **Variable Value** enter the email address you want to use.
5. Click **OK** to add the value.
6. Click **OK** to save the value in the environment variables.
7. Click **OK** to save the value in the system properties.

Other information

For Rational software product news, events, and other information, visit the IBM Rational Software Web site on http://www.ibm.com/software/rational/.

6 Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send written license inquiries to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send written inquiries to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions. Therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Intellectual Property Dept. for Rational Software
IBM Corporation
1 Rogers Street
Cambridge, Massachusetts 02142
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of

performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Additional legal notices are described in the legal_information.html file that is included in your software installation.

Trademarks

IBM, the IBM logo, ibm.com, Telelogic, Telelogic DOORS, and Telelogic DOORS for Rose Interface are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at www.ibm.com/legal/copytrade.html.

Microsoft, Windows and/or other Microsoft products referenced herein are either trademarks or registered trademarks of Microsoft Corporation.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product or service names may be trademarks or service marks of others.

Index