## **Starting Help**

StarUML(tm) provides various convenient functions to use this tool for varied users.Rose Add-In increases interoperability between modeling tools by being able to edit model files written by Rational Rose in StarUML(tm). If you hesitate to use the convenient environment of StarUML due to the model files by Rational Rose, please use the Rose Add-In of StarUML(tm).

Index of Rose Add-In Help

- 1. Rose Add-In Overview
- 2. Using Rose Add-In
- 3. <u>FAQ</u>

## **Rose Add-In Overview**

Top Previous Next

This chapter contains a general overview of Rose Add-In: functions and configurations.

- Rose Add-In Functions
- Rose Add-In Configurations

## **Rose Add-In Functions**

Rose Add-In provides the following function.

**Importing Rose files:** This function allows loading of Rational Rose model

• files into StarUML(tm). Since this function directly reads and analyzes Rose files, the system does not need to have Rational Rose installed.

## **Rose Add-In Configurations**

Once Rose Add-In is installed, it is enabled for use in StarUML(tm) by default. Installed Add-Ins can be enabled or disabled through Add-In Manager in StarUML(tm). If an Add-In is disabled, no main menu and popup menu items related to it are displayed, and no StarUML(tm) events are relayed to it.

**Procedure for Enabling Rose Add-In :** 

- 1. Select the **[Tools]->[Add-In Manager...]** menu in StarUML(tm).
- 2. At the Add-In Manager dialog box, check the "Rose Add-In" checkbox in the Add-In list.

Name	Version	
🗹 💣 C++ Add-In	1.0.1.68	
🗹 💣 C# Add-In	1.0.1.70	
🗹 🏝 Java Add-In	1.0.1.94	
Pattern Add-In	1.0.1.63	
🗹 👁 Rose Add-In	1.0.1.64	
🗹 🍑 Default Extension Pack	1.0.1.314	
ন		<b>)</b> • [

3. Click the **[OK]** button to close the dialog box.

**Procedure for Disabling Rose Add-In :** 

- 1. Select the **[Tools]->[Add-In Manager...]** menu in StarUML(tm).
- 2. At the Add-In Manager dialog box, uncheck the "Rose Add-In" checkbox in the Add-In list.
- 3. Click the **[OK]** button to close the dialog box.

# Using Rose Add-In

Top Previous Next

This chapter describes the main function of Rose Add-In: importing Rose files.

• Importing Rose Files

# **Importing Rose Files**

Top Previous Next

The main function of Rose Add-In is to convert Rational Rose model files into StarUML(tm) models.

**Procedure for Importing Rose Files :** 

- 1. Select the **[File]->[Import]->[Rational Rose...]** menu.
- 2. At the Import Rose Model File dialog box, click the **[Browse...]** button to activate the Select Rose Model File dialog box.

le path:	Browse
)ption	
<ul> <li>Convert Instance elemination</li> </ul>	ents to <u>R</u> ole elements (recommended)
<ul> <li>Generate diagram and</li> </ul>	view elements
Add the original file pat	th to the project attachments list
Add the original file pat Add the original file pat	th to the project attachments list
✓ Add the original file pat ogress	th to the project attachments list
✓ Add the original file pat ogress	th to the project attachments list

3. Select a Rational Rose model file to import and click the **[Open]** button.

Open		?×
Look jn:	: 🗹 Desktop 💽 🔶 📰 🕶	
History History Desktop My Computer My Network P	My Computer	
	File <u>n</u> ame:	ben
	Files of type:     Rational Rose model files(*.mdl;*.ptl)     Car	ncel

- 4. Examine the option group box in the Import Rose Model File dialog box and check the necessary options.
- 5. Click the **[Open]** button to import the Rose model file.

## **Options**

Various options are available for importing Rational Rose model files.

**Convert Instance elements into Role elements** : By default, Rational Rose mixes role elements and instance elements. The roles and instances contained in Rose model files must be converted into either instances or roles for

• StarUML(tm). This option item determines whether to convert Rose instance elements such as Stimulus and Object into role elements such as Message and Classifier.

**Generate diagram and view elements** : Determines whether to generate

• diagram and view elements for it. If unchecked, diagram and view elements are not generated.

**Add the original file path to the project attachments list** : If checked, the path of the original Rose file will be added to the project attachment file list.

• After importing a Rose file, select the project and double-click the Rose file path in the attachment file list to start Rational Rose and open the original file in it (if Rational Rose is installed in the user's system).

#### **File Formats Supported**

Rose Add-In can read the following file formats.

- .ptl file : A .ptl file is used for saving one or more model components in
- Rational Rose. Model components saved as a .ptl file can be copied or attached to other models.
- .mdl file : A .mdl file is a Rational Rose model file. It usually contains the entire model.

## FAQ

The following are frequently asked questions and answers for Rose Add-In.

- 1. Which Rose file versions can Rose Add-In read?
- 2. [Rational Rose...] cannot be found under the [Import] menu.
- 3. Model structure and views are different when reading a Rose file.

### 1. Which Rose file versions can Rose Add-In read?

Rose Add-In was designed for Petal File versions V45 and V46. These correspond to Rational Rose product versions 7.6.x and 7.7.x.

2. [Rational Rose...] cannot be found under the [Import] menu..

There are two possible reasons why the menu does not show up. One is that Rose Add-In is not installed in StarUML(tm) or related files are damaged. To correct this, run the installation program to reinstall StarUML(tm) or select to reinstall Rose Add-In only. If you are familiar with the structure of StarUML(tm) Add-Ins, you may examine the registry or menu files to check for damage in the Add-In and correct the problem. Another possible reason for the problem is that Rose Add-In is disabled by Add-In Manager. If so, go to Add-In Manager and enable Rose Add-In. For the procedure to enable Rose Add-In, see the section on <u>Rose Add-In Configurations</u>.

#### 3. Model structure and views are different when reading a Rose file.

Rational Rose differs from the UML specification standard in many ways. On the other hand, StarUML(tm) complies with the UML 1.4 specification standard. As a result, a Rose file may not look identical when opened in StarUML(tm). Although Rose Add-In is designed to read the model information defined in Rose files with minimum loss and alteration, view locations and some information may change. Please refer to the StarUML(tm) log window for any changes or loss of information and make appropriate modifications.