

Lab Manual: Using Rational Rose

1. Use Case Diagram

Creating actors

1. Right-click on the Use Case View package in the browser to make the shortcut menu visible.
2. Select the New:Actor menu option. A new actor called New Class is placed in the browser.
3. With the actor called New Class selected, enter the desired name of the actor.

Documenting Actors

1. If the documentation window is not visible, open the documentation window by selecting the Documentation menu choice from the View menu.
2. Click to select the Actor in the browser.
3. Position the cursor in the documentation window and enter the documentation.

Creating use cases

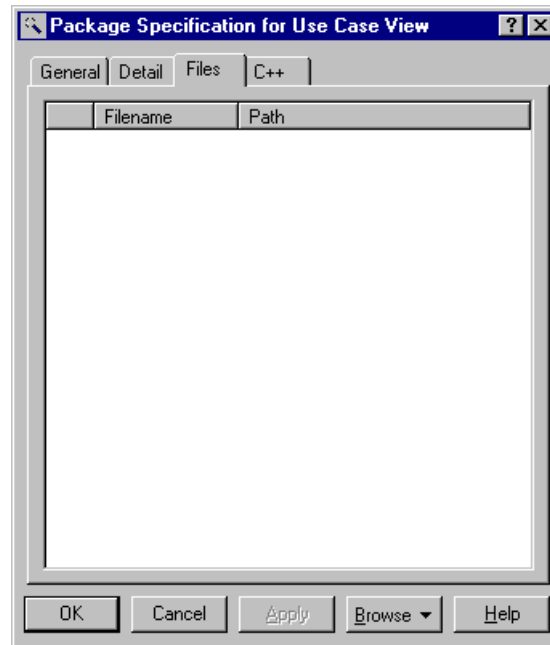
1. Right-click on the Use Case View in the browser to make the shortcut menu visible.
2. Select the New:Use Case menu option. A new unnamed use case is placed in the browser.
3. With the use case selected, enter the desired name of the use case.

Creating a use case brief description

1. Click to select the use case in the browser.
2. Position the cursor in the documentation window and enter the brief description for the use case. If the documentation window is not visible, select the View: Documentation menu choice to make the window visible.

Linking flow of events documents to use cases

1. Right-click on the use case in the browser to make the shortcut menu visible.
2. Select the specification menu option.
3. Select the Files tab.
4. Right-click to make the shortcut menu visible.
5. Select the Insert File menu option.
6. Browse to the appropriate directory and select the desired file.
7. Click the Open button.
8. Click the OK button to close the specification.



Creating the Main Use Case diagram

1. Double-click on the Main diagram in the use case view in the browser to open the diagram.
2. Click to select an actor in the browser and drag the actor onto the diagram.
3. Repeat step 2 for each additional actor needed in the diagram.
4. Click to select a use case in the browser and drag the use case onto the diagram.
5. Repeat step 4 for each additional use case needed in the diagram.

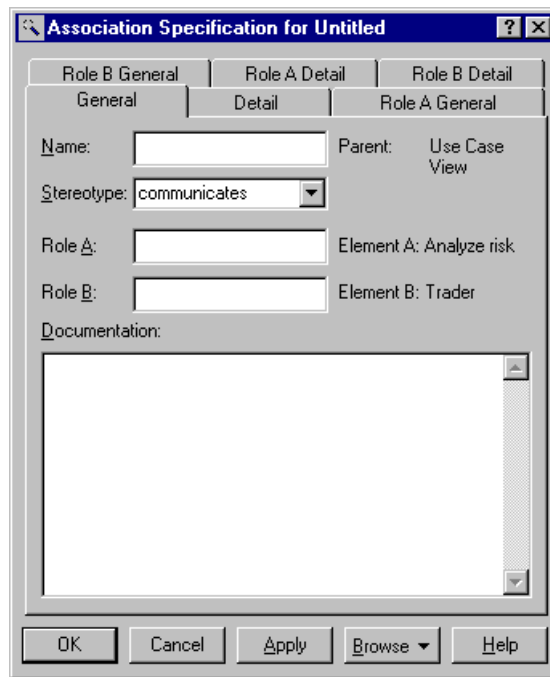
Note: Actors and use cases may also be created directly on a use case diagram by using the toolbar.

Creating Communicates Associations

1. Click to select the association icon or the Unidirectional Association icon from the toolbar. **Note:** *if the association icon is not present on the toolbar, it may be added by right clicking on the toolbar, selecting the Customize menu choice from the shortcut menu, and adding the icon to the toolbar.*
2. Click on an actor initiating a communication and drag the association line to the desired use case.

To add the Communicates stereotype (optional):

1. Double-click on the association line to make the Specification visible.
2. If this is the first time the stereotype is being used, enter Communicates in the Stereotype field. If a Communicates stereotype has already been created, click the arrow in the Stereotype field to make the drop-down menu visible and Select Communicates.
3. Click the OK button to close the specification.
4. Right-click on the association line to make the shortcut menu visible.
5. Select the Show Stereotype menu choice.
6. Repeat the preceding steps for each additional association relationship.

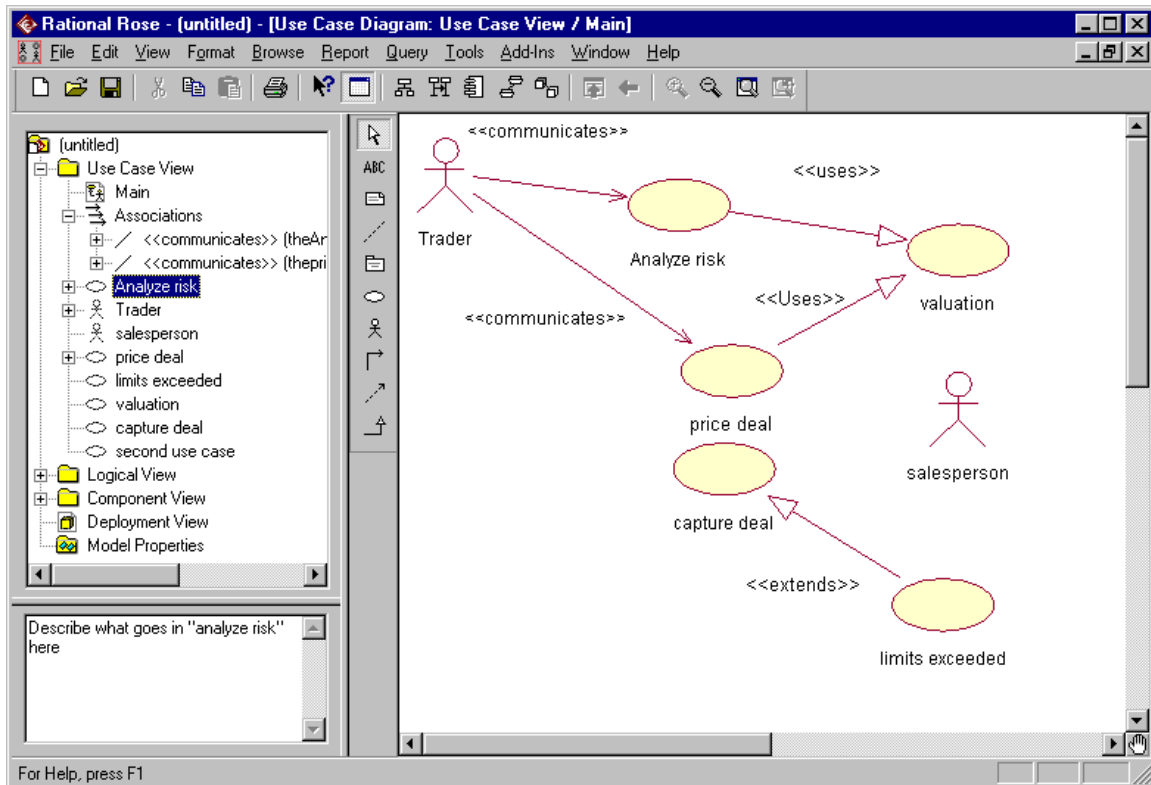


Creating Uses relationships

1. Click to select the Generalization icon from the toolbar.
2. Click on the using use case and drag the Generalization icon to the used use case.
3. Double-click on the generalization arrow to make the Specification visible.
4. If this is the first time a uses relationship is being created, enter Uses in the Stereotype field. If a Uses stereotype has been created, click the arrow in the Stereotype field to make the drop-down menu visible and select Uses.
5. Click the OK button to close the Specification.
6. Right-click on the generalization arrow to make the shortcut menu visible.
7. Select the Show Stereotype menu choice.

Creating Extends Relationships

1. Click to select the Generalization icon from the toolbar.
2. Click on the use case containing the extended functionality and drag the Generalization icon to the base use case.
3. Double-click on the generalization arrow to make the Specification visible.
4. If this is the first time an extends relationship is being created, enter Extends in the Stereotype field. If an Extends stereotype has already been created, click the arrow in the Stereotype field to make the drop-down menu visible and select Extends.
5. Click the OK button to close the Specification.
6. Right-click on the generalization arrow to make the shortcut menu visible.
7. Select the Show Stereotype menu choice.



Creating Additional Use Case Diagrams

1. Right-click on the Use Case View in the browser to make the shortcut menu visible.
2. Select the New:Use Case Diagram menu option.
3. While the use case diagram is selected, enter the name of the use case diagram.
4. Open the diagram and add actors, use cases, and interactions to the diagram as needed.

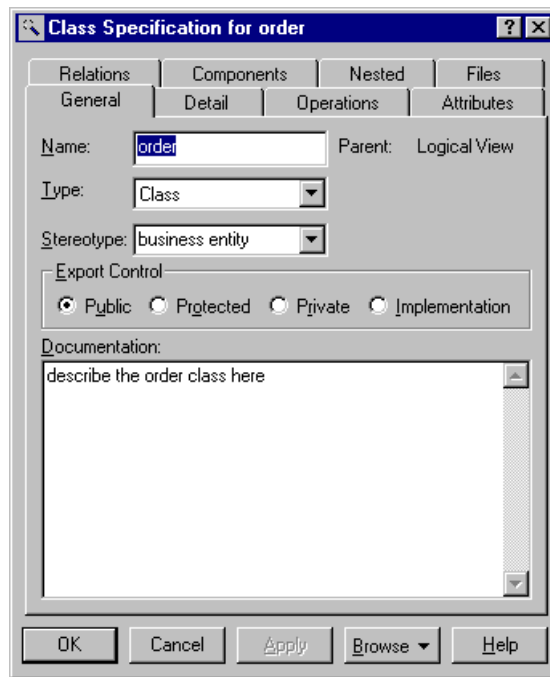
2. Class Diagram

Creating Classes in the Rose browser

1. Right-click to select the Logical View in the browser.
2. Select the New:Class menu choice. A class called New Class is placed in the browser.
3. While the new class is still selected, enter the name of the class.

Creating Stereotypes for Classes in Rational Rose

1. Right-click to select the class in the browser and make the shortcut menu visible.
2. Select the Specification menu choice.
3. Select the General tab.
4. Enter the name of the stereotype.
5. Click the OK button to close the Specification.



Documenting Classes

1. Click to select the class in the browser.
2. Position the cursor in the documentation window and enter the documentation for the class.
- 3.

Creating packages in the Rose browser

1. Right-click to select the Logical View in the browser.
2. Select the New:Package menu choice.
3. While the package is still selected, enter the name of the package.

Relocating Classes in the Rose browser

1. Click to select the class in the browser.
2. Drag the class to the desired package.
3. Repeat the steps for each class that is to be relocated.

The main Class Diagram

Rose automatically creates the Main class diagram in the Logical View of the model.

To add packages to the Main class diagram:

1. Double-click on the Main diagram in the browser to open it.
2. Click to select the package in the browser.
3. Drag the package onto the diagram.
4. Repeat the preceding steps for each package that is to be added to the diagram.

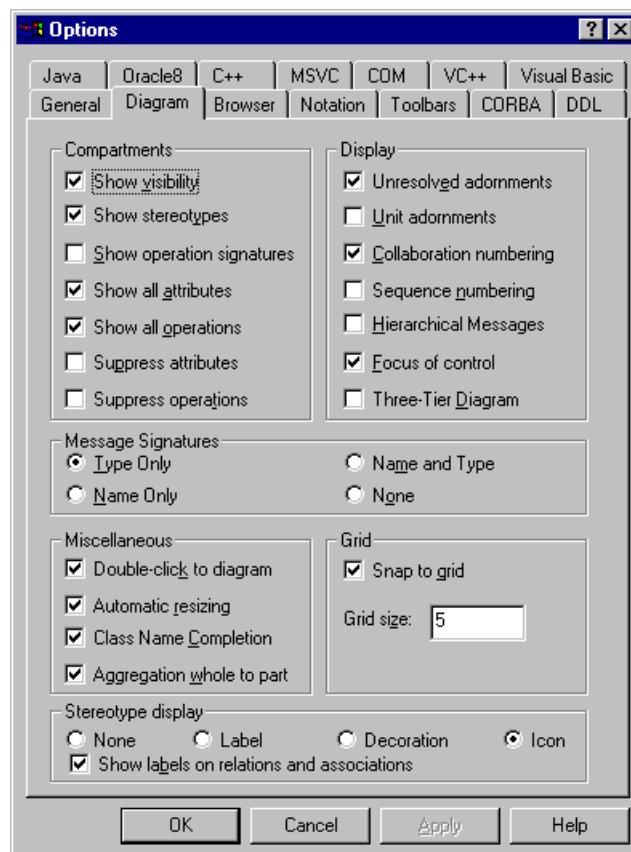
Creating a Package Main Class Diagram

1. Double-click on the package on a class diagram.

2. Rose will open the package and create (or display) the Main class diagram for the package.
3. Click to select a class in the browser and drag the class onto the diagram.
4. Repeat step 3 for each additional class that is to be placed on the diagram.

Creating a View of Participating Classes

1. Right-click on the use case in the browser to make the shortcut menu visible.
2. Select the New:Class Diagram menu choice.
3. While the diagram is still selected, enter the name of the class diagram.
4. Double-click on the diagram in the browser to open the diagram.
5. Click to select a class in the Logical view of the browser and drag the class onto the diagram.
6. Repeat step 5 for each additional class that is to be placed onto the diagram.



To Set the Visibility Display

1. Select the Tools:Options menu choice.
2. Select the Diagram tab.
3. Select the Show Visibility checkbox to set the default to show the visibility of all classes.

To set the visibility for a selected class:

1. Right-click to select the class on a diagram and make the shortcut menu visible.

2. Click to select or deselect the Show Visibility menu choice.

To Show the Stereotype

1. Right-click on the class in a class diagram to make the shortcut menu visible.
2. Select the Show Stereotype menu choice.

Creating an Association Relationship

1. Click to select the association icon from the toolbar.
2. Click on one of the associated classes in a class diagram.
3. Drag the association line to the other associated class

Creating an Aggregation Relationship

1. Select the aggregation icon from the toolbar.
2. Click on the class playing the role of the “part” in a class diagram and drag the aggregation line to the class playing the role of the “whole”.

Naming Relationships

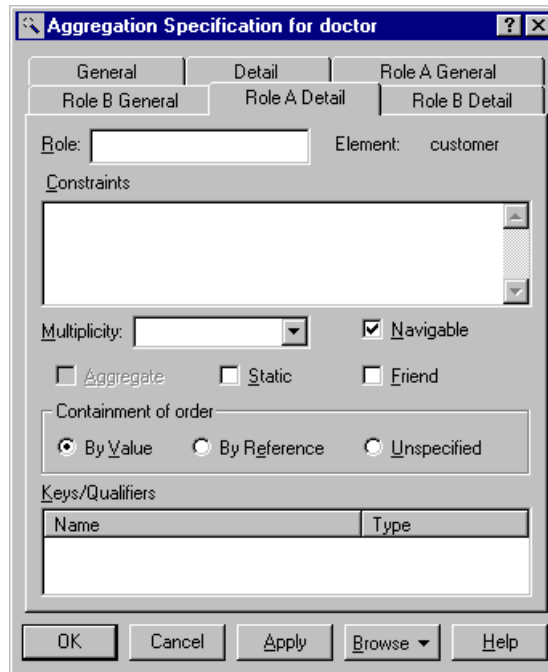
1. Click to select the relationship line on a class diagram.
2. Enter the name of the relationship.

Creating Role Names

1. Right-click on the relationship line near the class that it modifies to make the shortcut menu visible.
2. Select the Role Name menu choice.
3. Enter the name of the role.

Creating Multiplicity

1. Double-click on the relationship line to make the Specification visible.
2. Select the Detail tab for the role being modified (Role A Detail or Role B Detail).
3. Enter the desired multiplicity.



Creating a Reflexive Relationship

1. Select the association (or aggregation) icon from the toolbar.
2. Click on the class and drag the association (or aggregation) line outside the class.
3. Release the mouse button
4. Click and drag the association (or aggregation) line back to the class.
5. Enter the role names and multiplicity for each end of the reflexive association (or aggregation).

Creating Package Relationships

1. Select the dependency relationship icon from the toolbar.
2. Click on the dependent package and drag the arrow to the package it depends upon.

Documenting Operations

1. Click the + next to the class in the browser to expand the class.
2. Click to select the operation.
3. Position the cursor in the documentation window and enter the documentation for the operation.

Creating Attributes

1. Right-click to select the class in the browser and make the pop-up menu visible.
2. Select the New:Attribute menu choice. This will create an attribute called Name in the browser.
3. With the new attribute selected, enter the desired name.

Documenting Attributes

1. Click the + next to the class in the browser to expand the class
2. Click to select the attribute.
3. Position the cursor in the documentation window and enter the documentation for the attribute.

Creating a Class diagram to show the Attributes and Operations for a Package

1. Right-click to select the package in the browser and make the shortcut menu visible.
2. Select the New:Class Diagram menu choice. A class diagram called NewDiagram will be added to the browser.
3. With the new diagram selected, enter the name of the diagram.

Adding Classes to a Diagram Using the Query Menu

1. Double-click on the diagram in the browser to open the diagram.
2. Select the Query:Add Classes menu choice.
3. Select the desired package.
4. Click to select the desired classes and click the >>> button to add the classes to the diagram or click the All >> button to add all the classes to the diagram.

Filtering Relationships

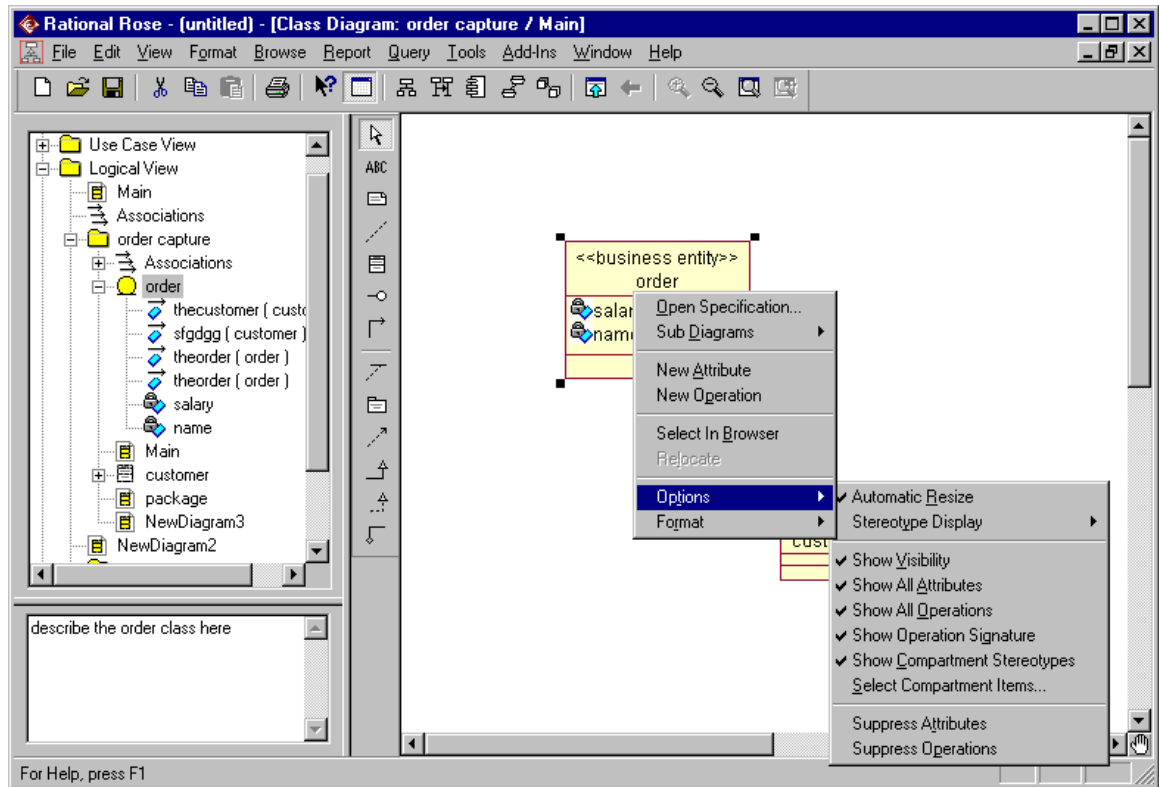
1. Double-click on the diagram in the browser to open the diagram.
2. Select the Query:Filter Relationships menu choice.
3. Click the None button in the Type field to hide all relationships shown on the open diagram.
4. Click the OK button to close the Relations window.

Displaying some Attributes or Operations

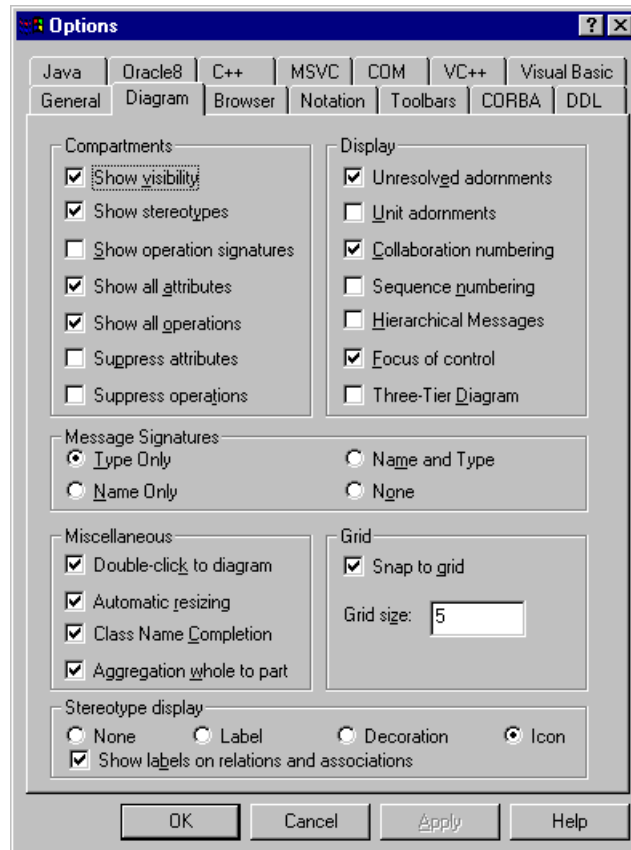
1. Right-click to select the class on an opened class diagram and make the shortcut menu visible.
2. Select the Edit compartment menu choice.
3. Click to select the attributes and operations to be displayed.
4. Click the >>>> button.
5. Click the OK button to close the Edit Compartment window.

Showing all Attributes and Operations

1. Right-click on the class in the class in a diagram to make the shortcut menu visible.
2. Select the Show All Attributes menu choice to display all the attributes for the class.
3. Repeat step 1 and select the Show All Operations menu choice to display all the operations for the class.



Note: to always display the attributes and operations for a class, you can set the Show All Attributes and Show All Operations selections using the Tools:Options menu.



Creating Association Classes

1. Click to select the class icon from the toolbar.
2. Click on the diagram to place the class.
3. With the class selected, enter the name of the class.
4. Add the attributes and operations to the class.
5. Click to select the Link attribute icon from the toolbar.
6. Click on the association class and drag the link attribute line to the association it modifies.
7. Create any additional relationships for the association class.

Creating Inheritance

1. Open the class diagram that will display the inheritance hierarchy.
2. Click to select the class icon from the toolbar and click on the open class diagram to draw the class.
3. With the class still selected, enter the name of the class.
Note: The class could also be created in the browser and added to the opened class diagram.
4. Click to select the Generalization icon on the toolbar.
5. Click on a subclass and drag the generalization line to the superclass.
6. Repeat step 5 for each additional subclass.

Creating an Inheritance Tree

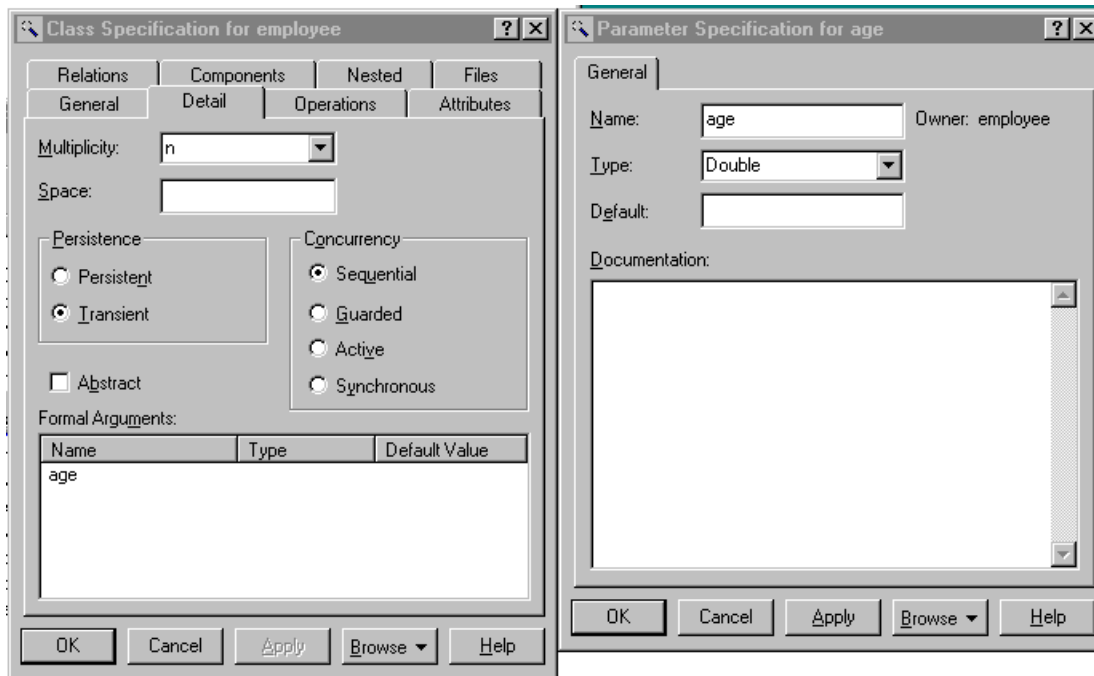
1. Open the class diagram that will display the inheritance hierarchy.
2. Click to select the class icon from the toolbar and click on the opened class diagram to draw the class.
3. With the class still selected, enter the name of the class.
Note: The class could also be created in the browser and added to the opened class diagram.
4. Click to select the Generalization icon on the toolbar.
5. Click on one subclass and drag the generalization line to the superclass.
6. For each subclass that is part of the inheritance tree, select the Generalization icon from the toolbar, click on the subclass, and drag the generalization line to the inheritance triangle.
Note: an inheritance tree may be created from two separate generalization arrows by selecting one arrow and dragging it onto the other arrow.

Relocating Attributes and Operations

1. Click the + sign next to one subclass in the browser to expand the class.
2. Select the attribute or operation to be relocated.
3. Drag the attribute or operation to the superclass.
4. Delete the attribute or operation from all other subclasses.
5. Repeat steps 2 through 4 for each additional attribute or operation to be relocated.

Creating Parameterized Classes

1. Click to select the Parameterized class icon from the toolbar. You may have to add this icon to the toolbar using the Customize option.
2. Click on a class diagram to place the parameterized class.
3. While the parameterized class is still selected, enter its name.
4. Right-click to select the class in the browser or on a class diagram.
5. Select the Specification menu choice.
6. Select the Detail tab.
7. Right-click on the Formal Arguments field to make the pop-up menu visible.
8. Select the Insert menu choice, which will insert a formal argument with a name of *argname* and a type of *argtype*.
9. With the *argname* selected, enter the name of the formal argument.
10. Click to select the *argtype* and enter the type of the formal argument.
11. Repeat steps 7 through 10 for each argument.
12. Click the OK button to close the Specification.



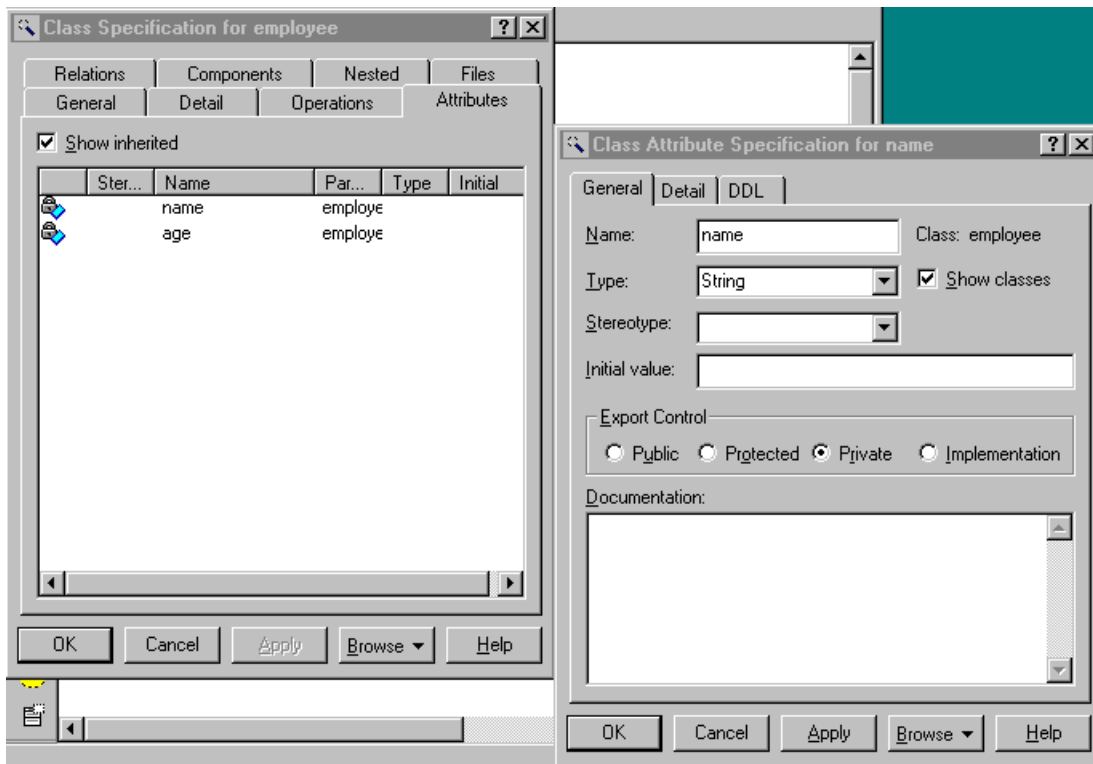
Creating Instantiated Parameterized Classes

1. Click to select the class icon from the toolbar.
2. Click on the class diagram to place the class.
3. With the class still selected, select the Edit:Change Into:Instantiated Class menu choice.
4. Right-click on the instantiated class to make the pop-up menu visible.
5. Select the Specification menu choice.
6. Select the Detail tab.
7. Right-click on the Actual Arguments field to make the pop-up menu visible.
8. Select the Insert menu choice, which will add an argument with name = argname and type = argtype.
9. With the argname still selected, enter the name of the argument.
10. Click to select the argtype and blank it out.
11. Repeat steps 7 through 10 for each actual argument.
12. Click the OK button to close the Specification.
13. Click to select the Dependency icon from the toolbar.
14. Draw a dependency relationship between the instantiated class and the parameterized class.
15. Add relationships to the instantiated class as needed.

Setting Attribute Data Types and Initial Values

1. Right-click to select in the browser or on a class diagram and make the shortcut menu visible.
2. Select the Specification menu choice.
3. Select the Attributes tab.

4. Attributes will have a data type set to *type* and an initial value set to *initval*. Click to select the *type* or *initval* and place the field in edit mode.
5. Enter the desired data type or initial value.



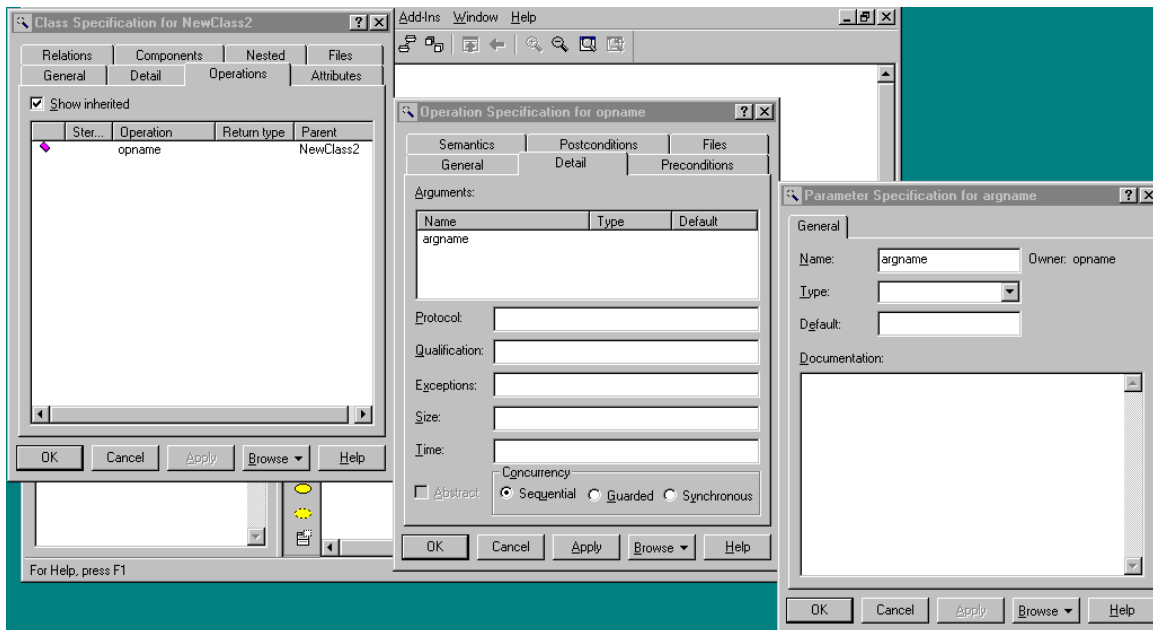
Setting Operation Signatures

1. Right-click to select a class in the browser or on a class diagram and make the shortcut menu visible.
2. Select the Specification menu choice.
3. Select the Operations tab.
4. Double-click on an operation to make the Operation Specification visible.
5. Enter the return class on the General tab.
6. Select the detail tab.
7. Right-click on the Arguments field to make the shortcut menu visible.
8. Select the Insert menu choice. This will add an argument. Enter the name, data type, and default value for the argument.
9. Click the Ok button to close the Operation Specification.
10. Click the Ok button to close the Class Specification.

Note: Attribute and Operation detail may also be set on a class diagram by selecting the displayed item and using the following format:

attribute: type = initial value

operation (argname: argtype = default): return class



Creating Virtual Operations

1. Click the + next to the class in the browser to expand the class.
2. Right-click on the operation to make the shortcut menu visible.
3. Select the Specification menu choice.
4. Select the cg tab.
5. Click to select the value for the property Operation Kind. This will place the value in edit mode.
6. Select virtual to make the operation virtual. Select abstract to make the operation pure virtual.
7. Click the OK button to close the Specification.

3. Interaction Diagrams

Creating a Sequence Diagram

1. Right-click to select the use case in the browser and make the shortcut menu visible.
2. Select the New:Sequence Diagram menu choice. An unnamed sequence diagram is added to the browser.
3. With the new sequence diagram selected, enter the name of the sequence diagram.

Creating Objects and Messages in Sequence Diagrams

1. Double-click on the sequence diagram in the browser to open the diagram.
2. Click to select the actor in the browser.
3. Drag the actor onto the sequence diagram.
4. Click to select the object icon from the toolbar.
5. Click on the sequence diagram window to place the object.
6. While the object is still selected, enter the name of the object.

7. Repeat the preceding steps for each object and actor in the scenario.
8. Click to select the object message icon from the toolbar.
9. Click on the actor or object receiving the message.
10. While the message line is still selected, enter the name of the message.
11. Repeat steps 7 through 9 for each message in the scenario.

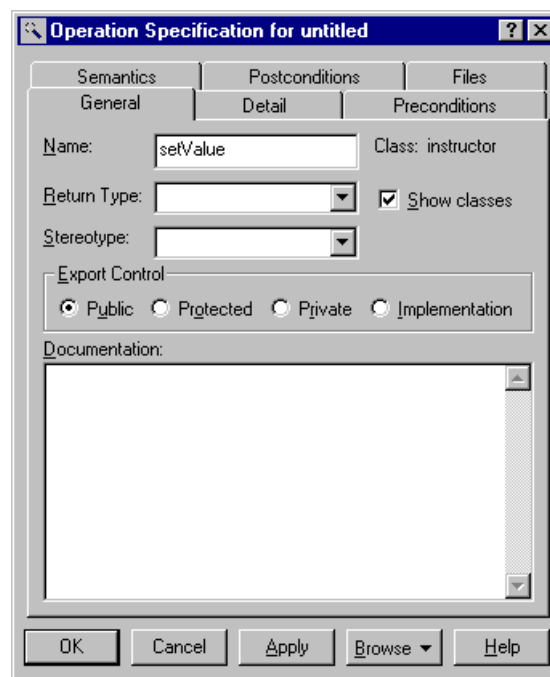
Assigning Objects in a Sequence Diagram to Classes

1. Click to select the class in the browser.
2. Drag the class onto the object in the sequence diagram. Rose will add the class name preceded by a colon to the object name. If the object is unnamed, the name is set to:className.

Mapping Messages to New Operations

1. Assign the objects to classes if that has not been done previously.
2. Right-click on the message arrow to make the shortcut menu visible.
3. Select the <new operation> menu choice. This will open the Operation Specification.
4. Enter the name of the operation in the Operation Specification.
5. Click the OK button to close the Operation Specification.
6. Right-click on the message arrow to make the shortcut menu visible.
7. Select the operation from the list of operations for the class.

Note: If the desired operation already exists for the class, you do not have to re-create the operation; simply select the operation from the list of operations for the class.



Creating Collaboration Diagrams from Sequence Diagrams

1. Double-click on the sequence diagram in the browser to open the diagram.

2. Choose the Browse>Create Collaboration Diagram menu choice or press the F5 key.
3. Rearrange the objects and messages on the diagram as needed.

4. State Transition Diagram

Creating State Transition Diagrams

1. Right-click to select the class in the browser and make the shortcut menu visible.
2. Select the State Diagram menu choice. This will automatically create and open the diagram.
3. To open the diagram at another time, click the + to expand the class in the browser, and double-click on its state transition diagram in the browser.

Creating States

1. Click to select the state icon from the toolbar
2. Click to place the state on the state transition diagram.
3. With the state still selected, enter the name of the state.

Creating State Transitions

1. Click to select the state transition icon from the toolbar.
2. Click to select the originating state on the state transition diagram.
3. Drag the state transition to the successor state.
4. If the state transition is a named transition, enter the name while the state transition arrow is still selected.

Creating Start States

1. Click to select the start icon from the toolbar.
2. Click on the state transition diagram to draw the start icon.
3. Click to select the state transition icon from the toolbar.
4. Click on the start icon and drag the arrow to the desired state.

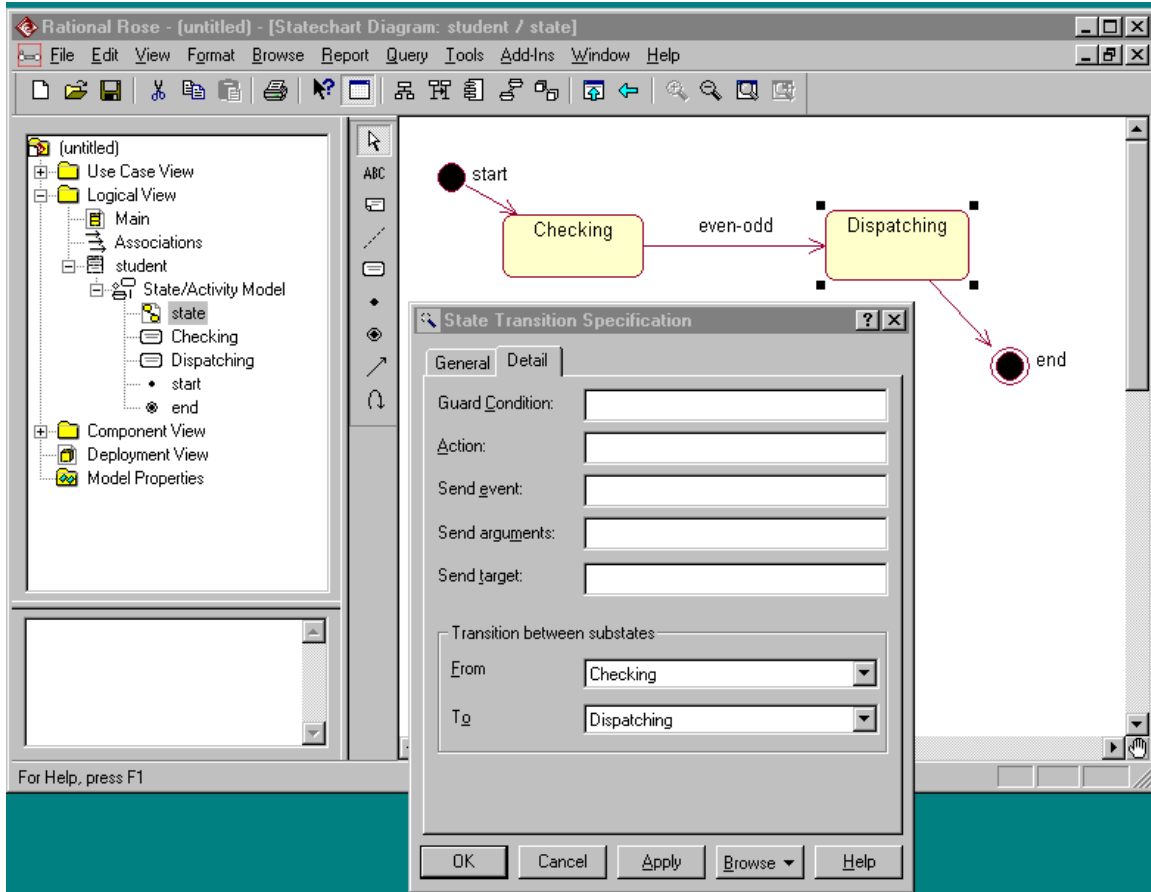
Creating Stop States

1. Click to select the stop icon from the toolbar.
2. Click on the state transition diagram to draw the stop icon.
3. Click to select the state transition icon from the toolbar.
4. Click on the stop icon and drag the arrow to the desired state.

Adding State Transition Details

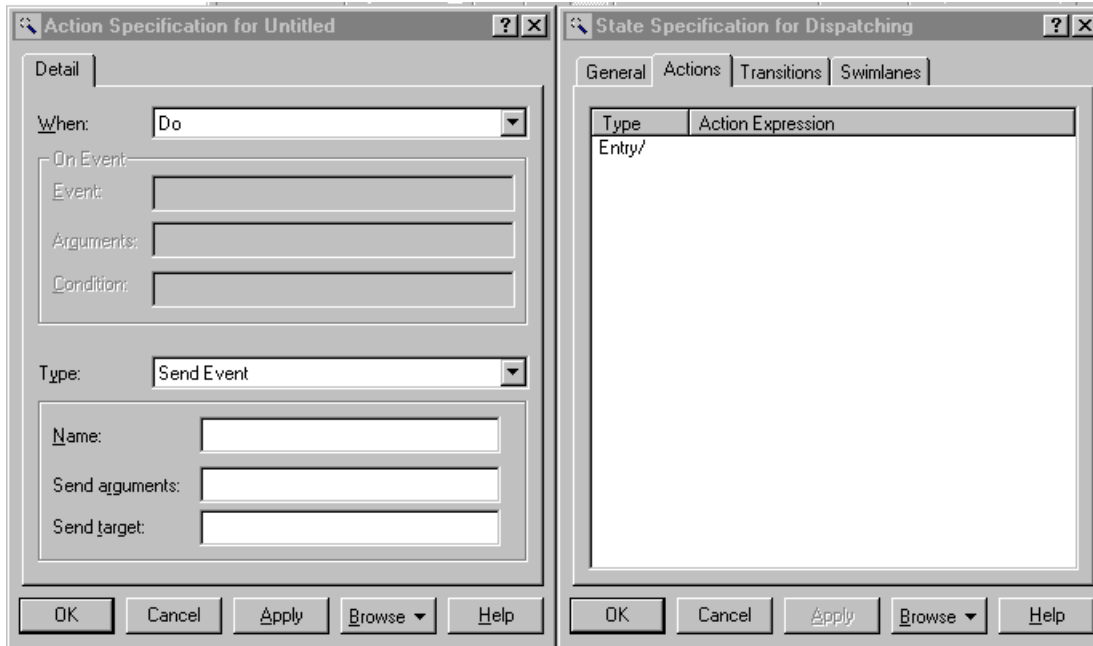
1. Right-click on the state transition arrow to make the shortcut
2. Select the specification menu choice.
3. Select the Detail tab.
4. Enter the action, guard, and/or the event to be sent.

5. Click the OK button to close the specification.



Creating Entry Actions, Exit Actions, and Activities

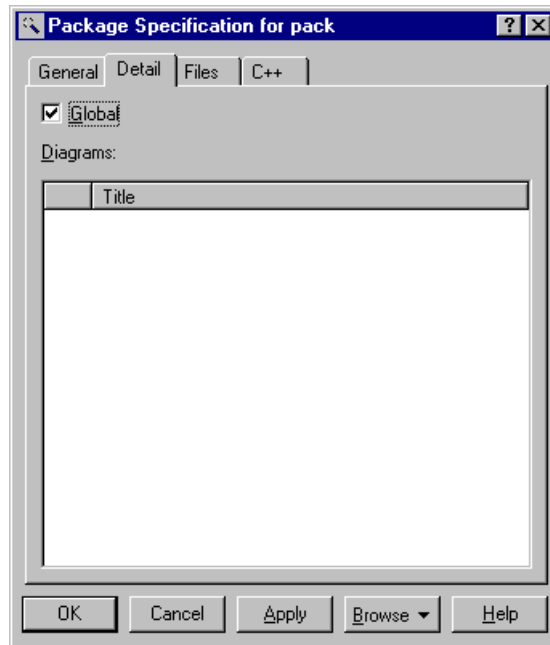
1. Right-click on the state to make the shortcut menu visible.
2. Select the specification menu choice.
3. Select the Detail tab.
4. Right-click on the Action field to make the shortcut menu visible.
5. Select the Insert menu choice. This will create an action called entry.
6. Double-click on entry to make the Action Specification visible.
7. Select the type of action: simple or send event.
8. Enter the action or send event information.
9. Select when the action should occur: on entry, on exit, entry until exit, or upon event.
10. Click the OK button to close the Action Specification.
11. Click the OK button to close the State Specification.



5. Architectural Diagrams

Making a Package Global

1. Right-click to select the package on a class diagram.
2. Select the Specification menu choice.
3. Select the Detail tab.
4. Click to select the global checkbox.
5. Click the OK button to close the specification.



Creating Component View Packages

1. Right-click to select the Component View package on the browser and make the shortcut menu visible.
2. Select the New:Package menu choice. This will add an item called NewPackage to the browser.
3. With the NewPackage still selected, enter the name of the package.

The Main Component Diagram

1. Double-click on the Main Diagram under the Component View package on the browser to open the diagram.
2. Click to select a package and drag the package onto the diagram.
3. Repeat step 2 for each additional package.
4. Dependency relationships are added by selecting the dependency icon from the toolbar, clicking on the package representing the client, and dragging the arrow to the package representing the supplier.

Mapping Logical Packages to component Packages

1. Right-click to select the component package on the browser and make the shortcut menu visible.
2. Select the Specification menu choice.
3. Select the Packages tab.
4. Right-click to select the logical package and make the shortcut menu visible.
5. Select the Assign menu choice.
6. Click the OK button to close the Specification.

Creating Components

2. Open a component diagram.
3. Click to select the Package Specification icon on the toolbar.

4. Click on the diagram to place the component. This will also add the component to the Browser.
5. While the component is still selected, enter the name of the component.

Mapping Classes to Components

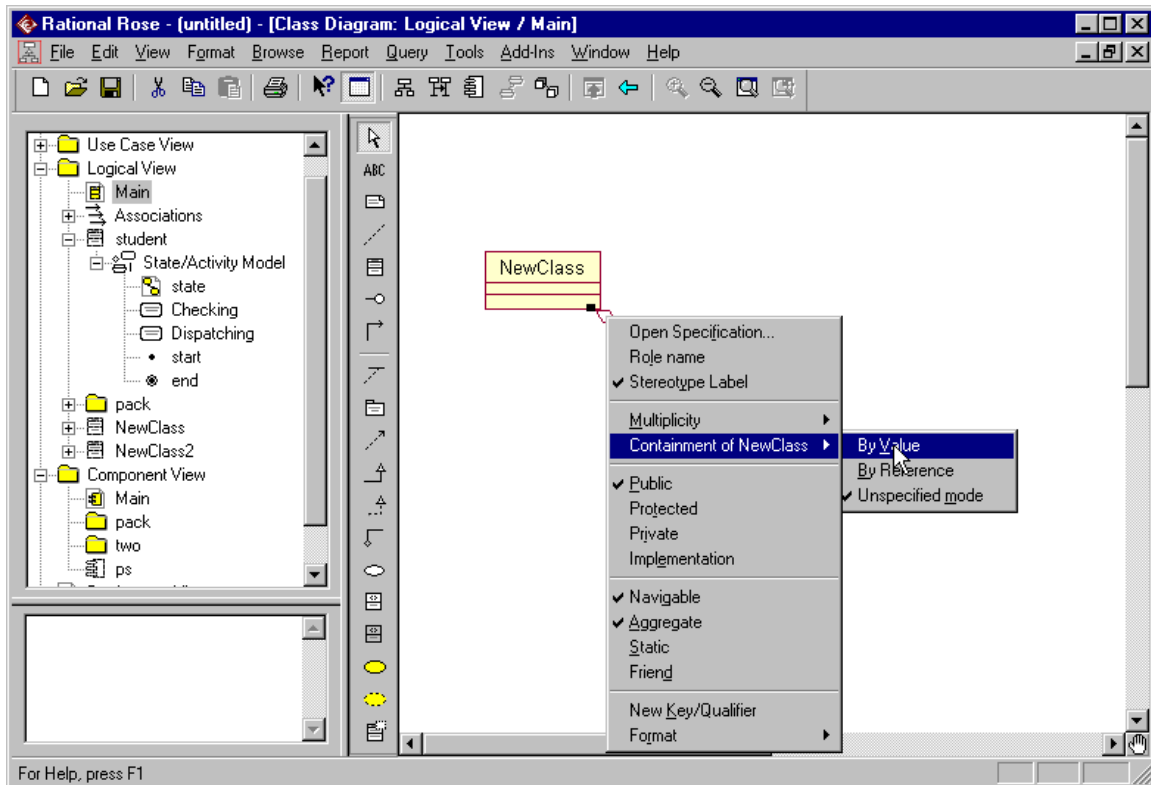
1. Right-click to select the component on the browser and make the shortcut menu visible.
2. Select the Specification menu choice.
3. Select the Classes tab.
4. Right-click to select the class and make the shortcut menu visible.
5. Select the assign menu choice.
6. Click the OK button to close the Specification.

Creating the Deployment Diagram

1. Rose automatically creates the deployment diagram. To open the diagram, double-click on the Deployment Diagram on the browser.
2. To create a node, click to select the Processor icon, and click on the diagram to place the node.
3. With the node still selected, enter the name of the node.
4. To create a connection between nodes, click to select the connection icon from the toolbar, click on one node on the deployment diagram, and drag the connection to the other node.

Setting Aggregation Containment

1. Right-click on the aggregation diamond to make the shortcut menu visible.
2. Select the desired containment: By value or By Reference.



Creating Dependency Relationships

1. Click to select the dependency relationship icon on the toolbar.
2. Click on the class playing the role of the client.
3. Drag the dependency relationship line to the class playing the role of the supplier.

6. Round-trip Engineering with Rose

Overview

Code generation Steps

1. Create Needed Property Sets
2. Create Body Components in the Component Diagram
3. Attach Property Sets to Classes
4. Select the Components and Generate the Code
5. Evaluate the Code Generation Errors

Reverse Engineering Steps

1. Create a project
2. Add Project Caption

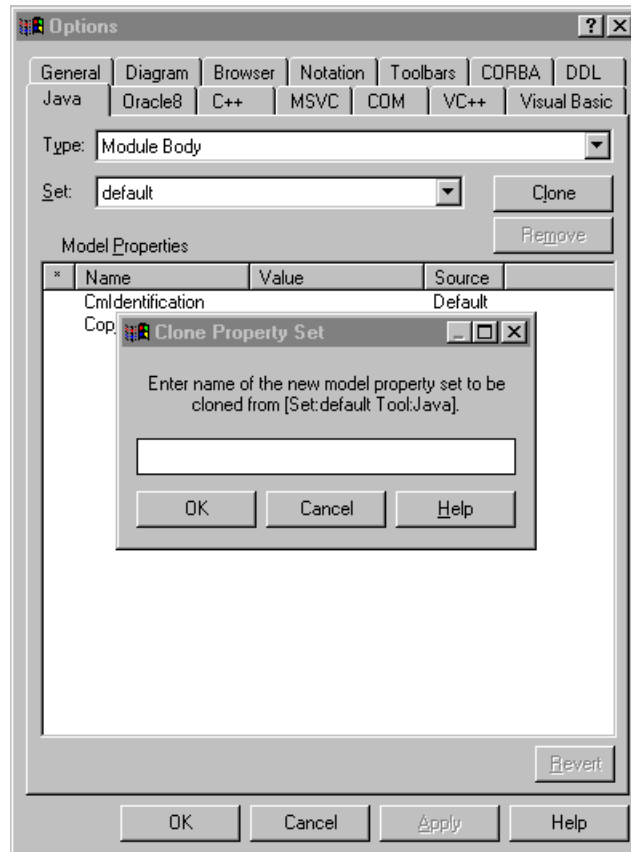
3. Add Referenced Libraries and Base Projects
4. Set the File Type and Analyze the Files
5. Evaluate the Errors
6. Select Export Options and export to Rose
7. Update Rose Model.

Code Generation Steps

Step 1: Create Needed Property Sets

There are code generation properties associated with the project, the class, the roles, the attributes, and operations. Project properties concern file name, default container, and placement of the generated code. Class properties concern constructors etc. Operation properties deal with the operation kind (abstract, static etc.) and allow the operation to be made constant. The property set for roles deals with the generation of get/set methods, visibility of methods, and the container class to be used.

1. Select the Tools:Option menu choice
2. Select the cg tab
3. Click on the arrow to make the drop-down menu visible
4. Select the desired type of property set.
5. Click the Clone button to make the Clone Property Set window visible.
6. Enter the name of the new property set.
7. Click the OK button to close the Clone Property Set window.
8. Click to select the property to change.
9. Click the Edit Value button to place the property in edit mode.
10. Enter the new value or select the new value from the drop-down menu if one is provided.
11. Repeat steps 8 through 10 for each property to be changed.
12. Click the Apply button to apply changes.
13. Repeat the preceding steps for each new property set.
14. Click the OK button to close the Options window.



Step2: Create Body Components in the Component Diagram

1. Double-click on a component diagram to open the diagram.
2. Click to select the Package Body icon on the toolbar.
3. Click to place the Package Body icon on the diagram.
4. While the body is still selected, enter the name of the body-this is typically the same name as the name of the specification.
5. Click to select the Dependency icon on the toolbar.
6. Click on the package body in the diagram, and drag the dependency to the package specification.

Step 3: Attach Property Sets to Classes

Each class is examined to determine its code needs. If a property set exists, it is attached to the class.

1. Right-click to select the element in the browser or on a diagram and make the shortcut menu visible.
2. Select the Specification menu choice.
3. Select the cg tab.
4. Click the arrow on the Set field to make the drop-down menu visible.
5. Select the desired property set.
6. Click the OK button to close Specification.

Overriding a property

1. Right-click to select the element in the Browser or on a diagram and make the shortcut menu visible.
2. Select the Specification menu choice.
3. Select the cg tab.
4. Click the arrow on the Set field to make the drop-down menu visible.
5. Select the desired property set.
6. Click to select the value to be changed.
7. Click the Edit Value button to place the property in edit mode.
8. Enter the new value or select the new value from the drop-down menu if one is provided.
9. Repeat steps 6 through 8 for each property to be changed.
10. Click the OK button to close the Specification.

Step 4: Select the Components and Generate the Code

Code may be generated for an entire package, a class, or a set of classes:

- If a class is not mapped to a component, the name of the class is used to generate the name of the file containing the code. The code is placed in a directory structure that corresponds to the package names in the logical view.
 - If a class is mapped to a component, the name of the component is used to generate the name of the file containing the code. The code is placed in a directory structure that corresponds to the package names in the component view.
1. Click to select the package, class, or set of classes.
 2. Select the Tools:Java:Code Generation menu choice.
 3. Rose will display the status in the Code Generation Status window.

Step 5: Evaluate the Code Generation Errors

Rose writes all warnings and errors to the Log window. If part of the design for the class is not complete, Rose will write a warning message to the log and use a default value.

The common warnings and errors output from the code generator are:

- Error: missing attribute data type; void is assumed
- Warning: unspecified multiplicity/cardinality indicators; one is assumed
- Warning: missing operation return type; void is assumed

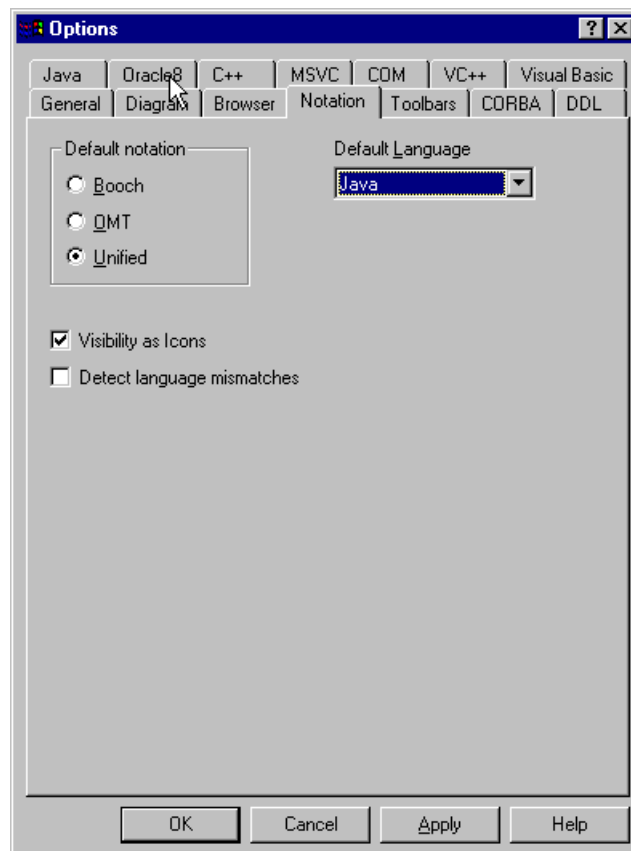
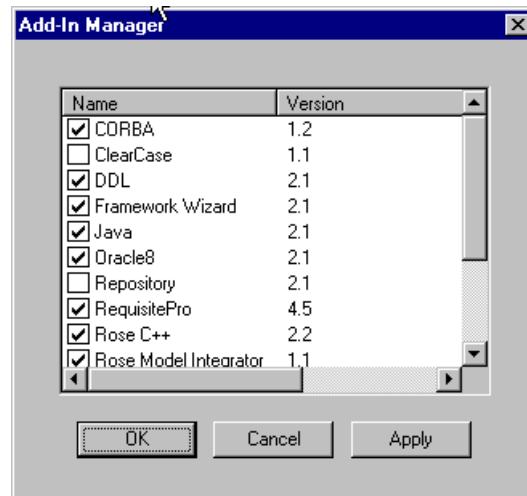
Forward-Engineering in Rose Java

Setting the Model's Default Language to Java

If most of the classes in your model will be Java classes, you might want to set the default language of the model Java. If you do this, each class you create

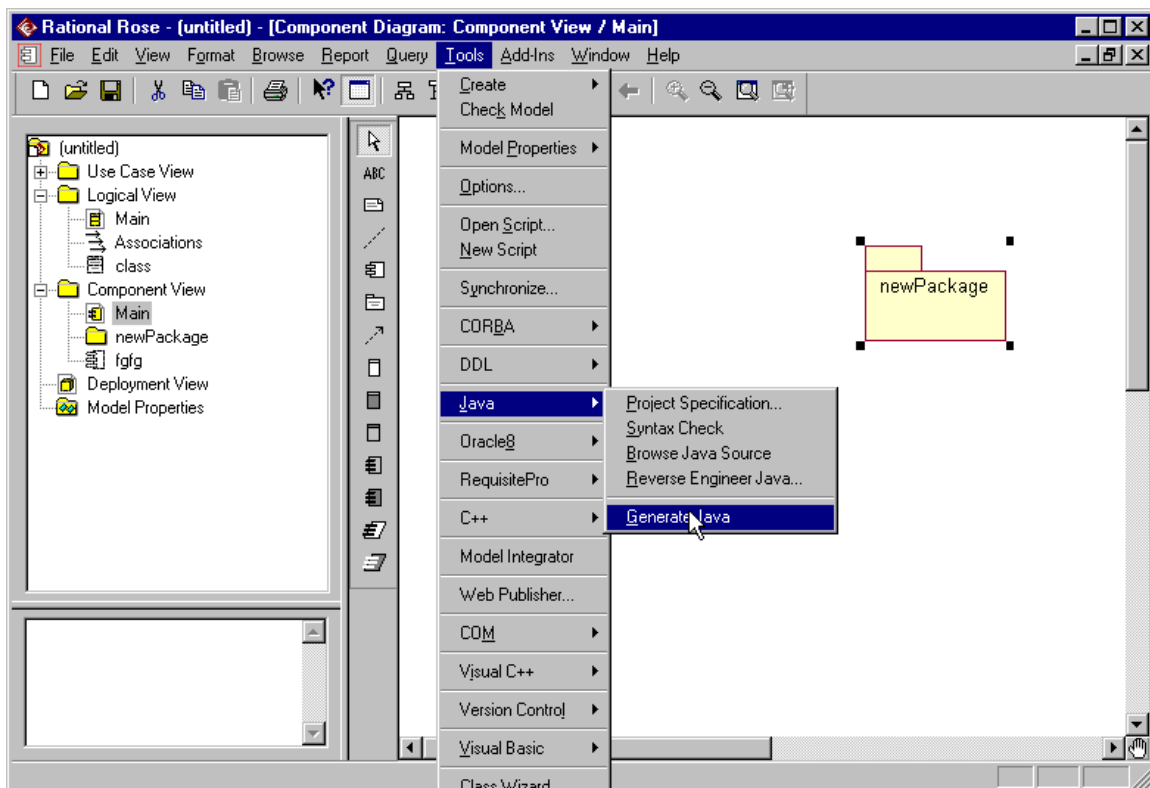
will a Java class by default. To set the default language to Java:

1. Verify that the Java Add-In is active by selecting **Add-In Manager** from the **add-Ins** menu. Activate it, if it is not currently active.
2. On the **Tools** menu, point to **Model Properties** and select **Edit**.
3. Click the **Notation** Tab in the **Model Properties** dialog.
4. Select **Java** from the **Default Language** list.
5. Click **OK** to set Java as the default language for your model.



Generating Java Source from a Class Diagram

1. Open your model and display the Class Diagram that contains the classes and packages for which you want to generate Java source.
2. Select one or more classes or packages in the diagram.
3. Optionally, use the syntax checker before generating your Java source.
4. On the **Tools** menu, point to **Java**, and then click **Generate Java**.
5. If a package or component hasn't been mapped to an entry in your classpath settings (for example, this is the first time you've generated code for a component or package), the Component Mapping dialog appears enabling you to set up the appropriate associations.
6. Check the Rose Log window to view the results of the Java generation, including any errors that occurred.
7. Correct any errors and repeat step 4 until no errors are returned.
8. View and edit the generated code.



Generating Java Source from a Component Diagram

1. Open your model and display the Component Diagram that contains the components and packages for which you want to generate Java source.
2. Select one or more components or packages in the diagram.
3. Optionally, use the syntax checker before generating your Java source.

4. On the **Tools** menu, point to **Java**, and then click **Generate Java**.
5. If a package or component hasn't been mapped to an entry in your classpath settings (for example, this is the first time you've generated code for a component or package), the Component Mapping dialog appears enabling you to set up the appropriate associations.
6. Check the Rose Log window to view the results of the Java generation, including any errors that occurred.
7. Correct any errors and repeat step 4 until no errors are returned.
8. View and edit the generated code.

Checking for Java Syntax Errors

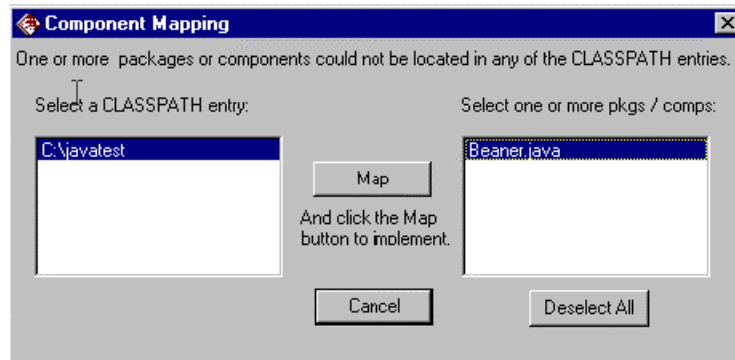
1. Open your model and display the Component Diagram that contains the components for which you want to generate Java source.
2. Select one or more packages and components in the diagram that contains the components for which you want to generate Java source.
3. On the **Tools** menu, point to **Java**, and then click **Syntax Check** to perform a check for rudimentary syntax errors in the component specifications.
4. Check the Rose Log window to view the results of the syntax check. If syntax errors are found at this level, you will probably not be able to compile the source that would be generated.
5. Correct the components as needed.
6. Repeat steps 3 through 5 until no more syntax errors are found. When no syntax errors are found, you can proceed to generate Java source from the packages and components you checked.

Note: The common warnings and errors output from the code generator are:

- Error: missing attribute data type; void is assumed
- Warning: unspecified multiplicity/cardinality indicators; one is assumed
- Warning: missing operation return type; void is assumed

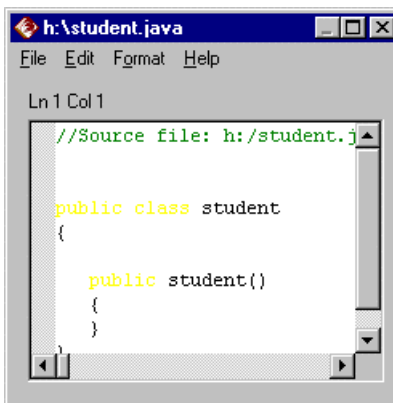
Mapping Components for Code Generation

1. From the list of Class Path entries, select the path (directory structure) where you want to put the .java file when it is generated.
2. From the list of packages and components, select one or more items to map to the Class Path entry you selected. The list of packages and components shows only the topmost level. Once you locate this level relative to your chosen Class Path, all its sub packages and components in your model will be located within this hierarchy.
3. Click **Map** to map your selected package or component to the selected directory structure and complete code generation.



Browsing Java Source

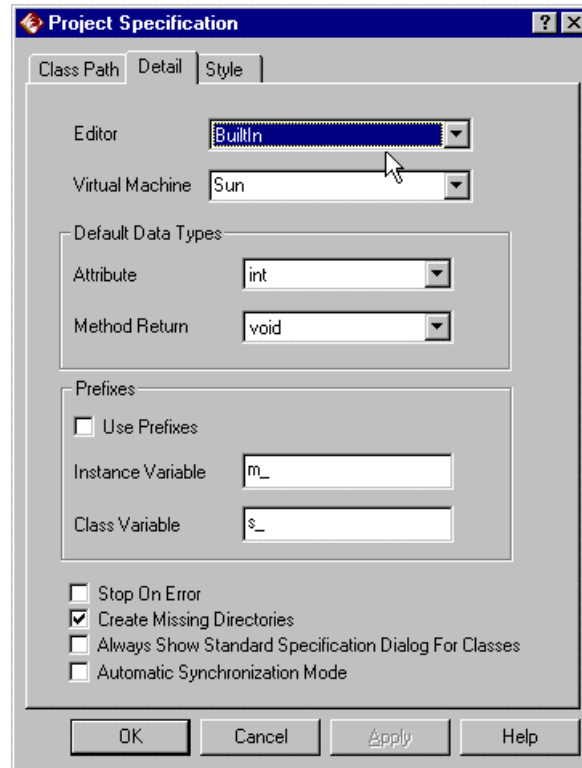
1. Select the class or component in whose source you want to browse.
2. Click **Browse Java Source** from the **Tools – Java** menu. The source file is displayed using the application you established as your default (The default source editor).



Setting the Default Source Editor

1. On the **Tools** menu, point to **Java**, and then click **Project Specification**.
2. Click the **Detail** tab.
3. In the Editor list, click **Builtn** or **WindowShell**.
4. Click **OK** to save the Editor property value

Note: If you select **WindowShell**, be sure your file association (a standard Windows function) for .java files is set to the editor of your choice. It is recommended to select **Builtn**, which the Rose built-in editor.



Java Field Specification for a Bean Property

Forward Engineering in Rose CORBA

Consists of generating CORBA source from one or more classes, package, or components in a Rose model. Forward engineering in Rose CORBA is component-centered. This means that the CORBA source generation is based on the component specification rather than on the class specification. This does not mean, however, that you must work only in component diagram in order to generate CORBA source. Instead, you can create a class and then assign it to a valid CORBA component in the Rose Browser, effectively creating the required component from your class.

Rose CORBA maps the classes in your model to corresponding CORBA constructs based upon the stereotypes you assign to them. Any attributes and relationships you define for Rose CORBA classes map to CORBA IDL attributes. In addition, any operations you define for Rose CORBA interfaces map to CORBA IDL operations.

Generating CORBA Source from a Class Diagram

1. Open your model and display the Class Diagram that contains the classes and packages for which you want to generate CORBA source.
2. Select one or more classes or packages in the diagram.

3. On the **Tools** menu, point to **CORBA**, and then click **Generate Code**.
4. If the Component Mapping dialog appears, use it to map any unresolved package or component locations.
5. Check the Rose Log window to view the results of the code generation, including any errors that occurred.
6. Correct any errors and repeat step 2 through 5 until no errors are returned.

Generating CORBA Source from a Component Diagram

1. Open your model and display the Component Diagram that contains the components or packages for which you want to generate CORBA source.
2. Select one or more components or packages in the diagram.
3. Optionally, use the syntax checker before generating your CORBA source.
4. On the **Tools** menu, point to **CORBA**, and then click **Generate Code**.
5. If the Component Mapping dialog appears, use it to map any unresolved package or component locations.
6. Check the Rose Log window to view the results of the code generation, including any errors that occurred.
7. Correct any errors and repeat steps 4 and 5 until no errors are returned.
8. View and edit generated code.

Mapping Components during CORBA Source Generation

1. From the list of IncludePath entries, select the path (directory structure) in which to put the IDL file when it is generated.
2. From the list of packages and components, select one or more items to map to the IncludePath entry you selected. The list of packages and components shows only the topmost level. Once you locate this level relative to your chosen IncludePath, all its sub packages and components in your model will be located within this hierarchy.
3. Click **Map** to map your selected package or component to the selected directory structure and complete code generation

Setting the Model's Default Language to CORBA

If most of the classes in your model will be CORBA classes, you might want to set the default language of the model CORBA. If you do this, each class you create will a CORBA class by default. To set the default language to CORBA:

6. Verify that the CORBA Add-In is active by selecting **Add-In Manager** from the **add-Ins** menu. Activate it, if it is not currently active.
7. On the **Tools** menu, point to **Model Properties** and select **Edit**.
8. Click the **Notation** Tab in the **Model Properties** dialog.

9. Select **CORBA** from the **Default Language** list.
10. Click **OK** to set CORBA as the default language for your model.