Using Eclipse workspaces with Rational Method Composer

Rational Method Composer Version 7.2.0

Contents

Workspace method libraries

Rational Method Composer is built using Eclipse technology. Beginning in release 7.2, you can use Eclipse workspaces to organize Rational Method Composer method libraries and method plug-ins. Such workspace-based libraries are referred to as *workspace method libraries*.

An Eclipse workspace is a locally maintained directory in which your work files and settings (preferences, perspectives layout, and references to other file directories) are stored. For more information on Eclipse workspaces and Eclipse projects, see the *Workbench User Guide* in Eclipse documentation.

Workspace method libraries are an optional extension to Rational Method Composer that improve multi-author library set-ups. You can continue to use method libraries without the new workspace feature as well. For example, you might want to use libraries without workspaces when you work on your own or work with very small teams, but use workspace method libraries in situations in which there are many method plug-in authors that need concurrent access to the method library through a source control system.

Using workspace method libraries offers several advantages:

- You can create a scalable organization of your method plug-ins.
- Method libraries in a workspace can be distributed in many physical locations using different source control systems.
- You can create local plug-ins in a distributed method library that are not shared.
- With the import feature, you can more easily load and unload method plug-ins from different libraries.
- Multiple users can work on a method library concurrently (when working on separate plug-ins).
- Every method plug-in is an Eclipse project and can be imported or exported as such.

Differences between modes of library organization

Rational Method Composer supports two modes of working with method libraries:

- Single-project method library mode, in which the method library is maintained as a single Eclipse project. This is the default mode in which a library is loaded by pointing to a directory that contains a library.xmi file and a .project file as well as all method plug-ins, method configurations, and estimation data in sub-directories.
- Workspace method library mode, in which the method library is maintained as an Eclipse workspace consisting of method plug-in projects, method configuration projects, and an estimation project. With workspace method libraries, every method plug-in becomes its own Eclipse project (recognizable by its own .project file in the plug-in directory on the file system). A library.xmi file is not required.

  Three new kinds of Eclipse projects are supported for workspace method libraries:

  - Configuration projects, which store a number of related method configurations that can be exported and deployed as one unit
  - Method plug-in projects, which store the plugin.xmi file as well as all other files that make up a method plug-in
  - Estimation projects, which store the global set of estimation model and estimation data for all method plug-ins. There can be only one estimation project per method library.

Converting a single-project method library to a workspace method library

To use your method plug-ins and configurations from a single-project method library in the workspace method library mode, you must first convert the method library to workspace projects.

This section describes the procedure for converting a single-project method library to a workspace method library.

1. From the **File** menu, select **Open** > **Method Library**.
2. At the Open Method Library window, select **Open method library from workspace**. Click **Finish**. The workspace method library is opened.
   Note:
   When you do this for the first time the library is empty and the Library view shows an empty Configurations folder.
3. To convert and import an existing single-project method library, select **File > Import**. At the Select window, choose **Library Configuration** and click **Next**.
4. At the Specify import directory window, in the **Directory** field, provide the path name of the single-project method library to import. Click **Next**.
5. Before the library is imported, you are prompted to review the changes that will be made. Examine the information in the Review library changes window and click **Finish** after making any changes.
6. At the Backup Library window, choose whether to backup the old library by clicking **OK** or **Skip**.
7. The Create New Configuration Project window is displayed, prompting you to provide a name for the configuration project.

   In contrast to a single-project method library, when working with a workspace method library, you can combine method plug-ins and configuration from different sources (for example, combining many different libraries into one workspace) and you can manage configuration in multiple configuration projects. To distinguish these projects from one another, enter a descriptive name for this project, replacing the default string, "configuration." Click **OK**.

8. Verify that all resource files have been imported. (Compare your old library with the new and copy in any resource files that may have been missed.)

Creating new method plug-ins in workspace method libraries

To create a new method plug-in in a workspace method library, create the plug-in using the steps described in **Creating Method Plug-ins** in the Getting Started section with these additions:

1. At the Create a new method plug-in window, supply the required information and click **Next**.
2. The Create method plug-in project window is displayed, at which you specify a name and location for the project for the plug-in you are creating.
    a. Specify a project name in the **Project name** field. In general the project name will be the same as the plug-in name, although experienced Eclipse users can change the name to differentiate a method plug-in from a project name.
    b. Specify a storage location for the project. You can choose the current Rational Method Composer workspace directory location or specify another location.
        - To store the method plug-in in the current workspace directory, accept the default selection, **Save the project in the current workspace**. Verify that the project name shown is correct and click **Finish**
        - To select another location in which to save the method plug-in files, clear the checkbox for **Save the project in the current workspace** and specify the new location using the **Browse** button or by typing the path name in the **Project path** field. You can store the plug-in in a workspace or in another location such as a source control directory (for example a Rational ClearCase view directory). Click **Finish**.

Importing method plug-in projects or method configuration projects into a workspace method library

This section describes the procedure for importing method plug-in projects or method configuration projects into a workspace method library.

You can share your method plug-ins or method configuration projects among workspace method libraries using source control systems such as Rational ClearCase or CVS. In addition, you can copy method plug-ins or method configuration projects not under source control from one workspace library to another. Method plug-ins from single-project method libraries cannot be shared. For information on converting single-project method libraries into workspace method libraries, see Converting a single-project method library to a workspace method library.

To import method plug-ins or configuration projects provided by other Rational Method Composer users, follow these steps:

1. From the main product menu, select **File > Import**.
2. From the Import window, select **Select Existing Projects into Workspace** as your import source. Click **Next**.
3. Specify a root directory (a Rational ClearCase VOB directory in a view, for example) and press **TAB**.
4. Select the projects to import.
5. If you are using a source control system such as Rational ClearCase, clear the checkbox **Copy projects into workspace** because you want to keep the project in the Rational ClearCase view location and not work with a local copy.
6. Click **Finish**. Any problems with the import are displayed in the Probems window. Right-click on an error description for more options.

Note:
To circumvent a workspace refresh defect in Eclipse, after completing the import, re-open your workspace library by selecting **File > Open > Method Library**. Select **Open method library from workspace** and click **OK**.

Using multiple Rational Method Composer workspaces

Sometimes it is necessary to maintain more than one workspace method library, such as when working in multiple unrelated method design endeavors. By default, Rational Method Composer creates the workspace directory in your home directory, for example, C:\Documents and Settings\IBM\RMC\workspace.72 on Windows XP for Rational Method Composer 7.2. To create another workspace you need to start Rational Method Composer with the option and argument `-data` *pathname*, When this instance of Rational Method Composer is started, a new workspace folder is created using the pathname you specified. You can do this from a command line or, for easy and permanent reference, by creating a shortcut.

To create a shortcut that starts Rational Method Composer with a different workspace, follow these steps:

1. Create a copy of the Rational Method Composer start-up icon.
2. Select the copy and right-click. Select **Properties**.
3. In the **Target** field, add the -data option and a pathname as shown in the following example: "C:\Program Files\IBM\RMC72\rmc\rmc.exe" -data "C:\primary_workspace" The workspace folder is created when you start Rational Method Composer.