



How to Transform, Build and Execute Simple Oven Supplement to Quick Start Guide

Version 1.2
June 18, 2007

MDA Mentor

Pathfinder Solutions LLC
33 Commercial Drive, Suite 2
Foxboro, MA 02035 USA
www.PathfinderMDA.com
508-543-7222

Table Of Contents

- 1. Transform and Build..... 3**
 - Task 1: Reuse elements from Example Project.....3
 - Task 2: Create a PathMATE Project.....8
 - Task 3: Transform Your Model to Code.....9
- 2. Run SimpleOven with Spotlight..... 10**
 - Task 1: Execution Control 10

1. Transform and Build

Now you are ready to transform your model into an executable system. This section takes you through the following steps:

- Instantiate a Reference Project
- Create a Java Project to build the generated code
- Create a PathMATE Project
- Transform Your Model
- Build an Executable System

Task 1: Reuse elements from Example Project

SimpleOven exists as a sample model in the PathMATE examples library. Instantiate SimpleOven as a reference project in order to:

- Copy the implementation for the ExternalDeviceControl realized domain.
- Copy the properties that integrate realized domain implementation for ExternalDeviceControl and SoftwareMechanisms.

PROCEDURE: Instantiate a Reference Project

To instantiate the reference project for SimpleOven:

1. Select Window > Open Perspective > Modeling from the main menu bar. The perspective switches the Modeling perspective.
2. Select File > New > Example from the main menu bar. The New Example wizard opens.
3. Expand PathMATE, select SimpleOven, and click Next.
4. Type ReferenceSimpleOven in the Project name field.
5. Click Finish. ReferenceSimpleOven appears in the Model Explorer.

PROCEDURE: Create a Java Project

To create an Eclipse project to build the generated Java implementation of SimpleOven:

1. Enable the Java Development capability. Select Window > Preferences from the main menu bar. Select Workbench / Capabilities from the tree on the left side of the dialog. Check Java Developer in the capabilities on the right side of the dialog.

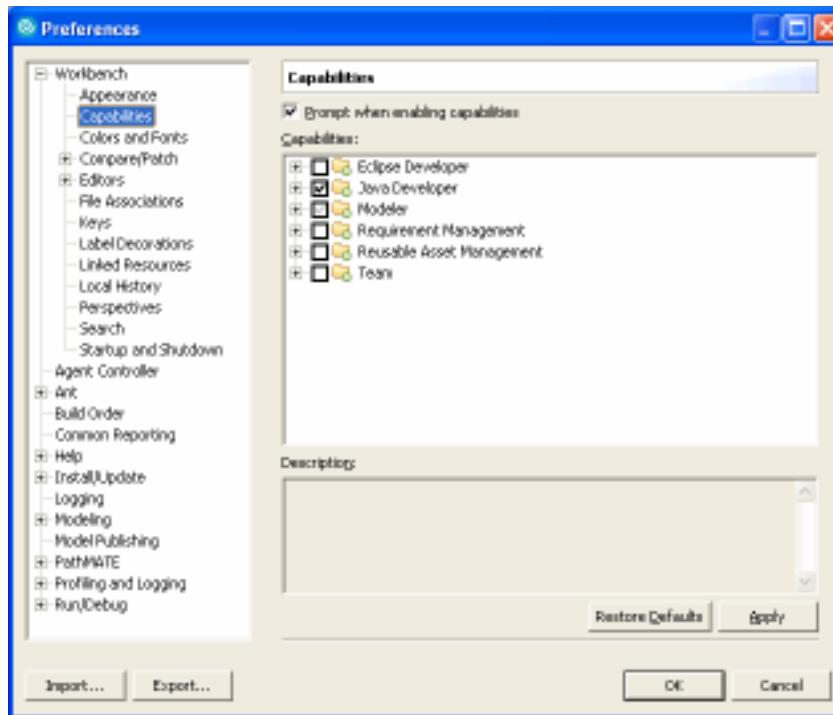
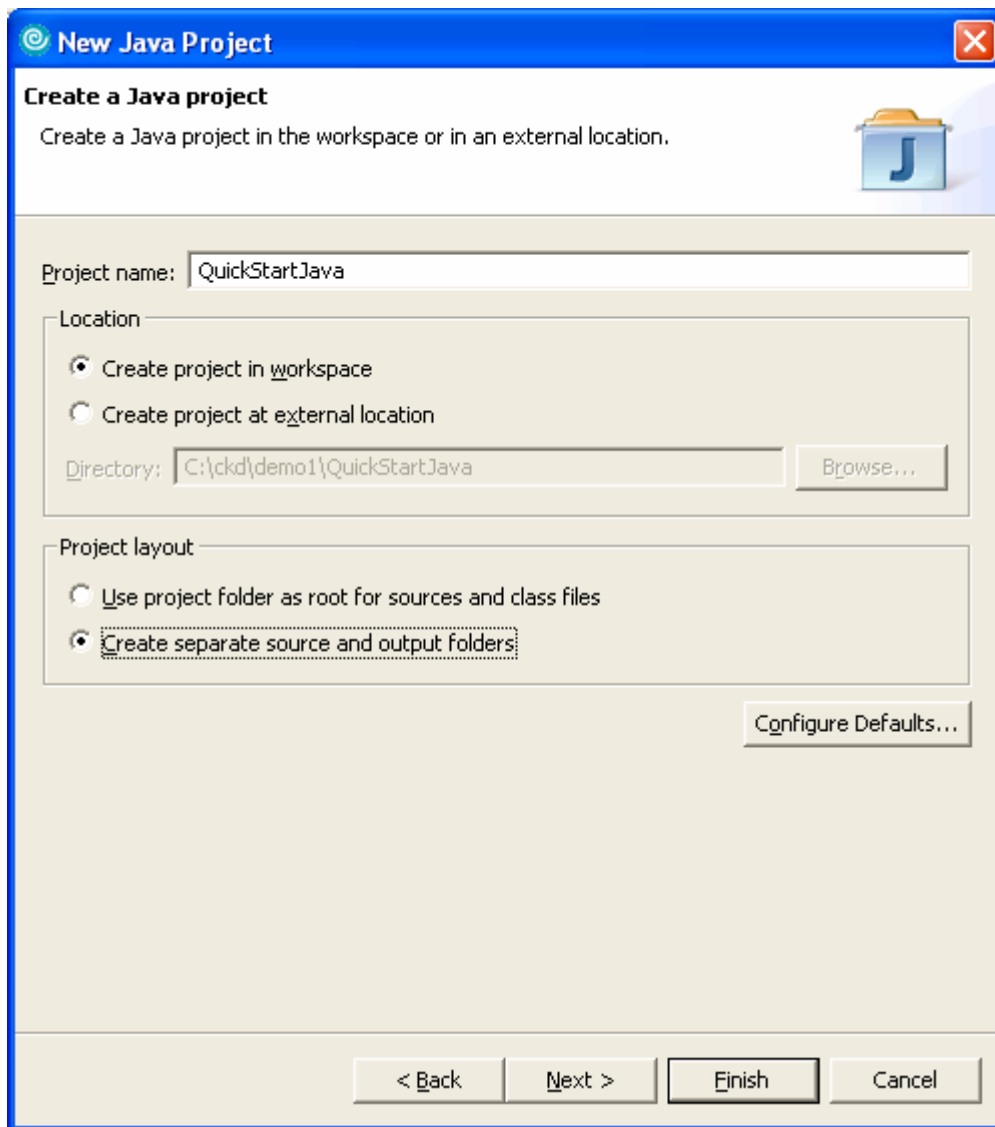


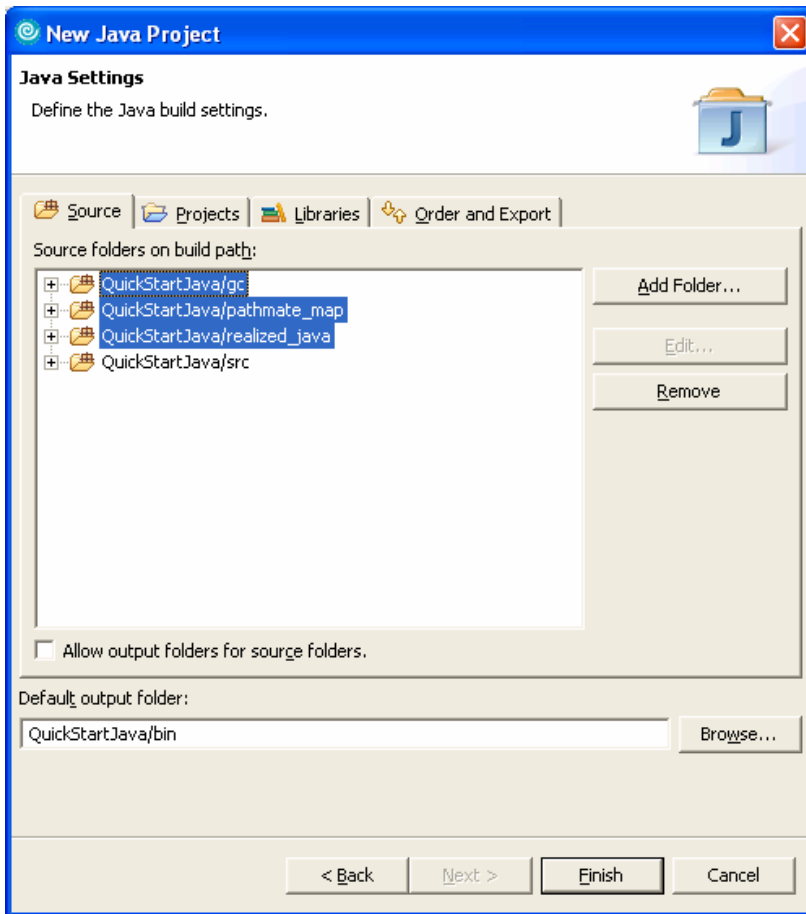
Figure 1 - Enabling the Java Developer Capability

2. Press Ok.
3. From the main RSM menu select File > New > Project.
4. The New Project Wizard Appears.
5. Select Java / Java Project from Wizards.
6. Press Next. The Create a Java Project page appears.
7. Enter the name QuickStartJava.
8. Locate the Project layout section. Click the Create Separate Source and Output folders radio button.



9. Press Next. The Java Settings page appears.
10. At the source tab, press the Add Folder button. The Source Folder Selection dialog appears.
11. Select QuickStartJava from the tree.
12. Press the Create New Folder button. The New Folder dialog appears.
13. Enter gc in the Folder name field.
14. Press OK. The gc source folder is currently empty but the transformation engine will place the generated code there.
15. Select QuickStartJava from the tree on the Source Folder Selection dialog.
16. Press the Create New Folder button again. The New Folder dialog appears.

17. Enter pathmate_map in the Folder name field. The pathmate_map source folder links in the PathMATE Java mechanisms and instrumentation mechanism packages.
18. Press the Advanced button. If you don't see the Advanced button, press Cancel on the New Folder dialog. The New Folder dialog is dismissed. On the Source Folder Selection dialog, make sure that QuickStartJava at the root of the tree is selected. Press the Create New Folder button. Enter pathmate_map in the Folder name field. Proceed with the next step.
19. Check Link to folder in the file system.
20. Press the Browse button. The Browse for Folder dialog appears.
21. Select the directory c:\pathmate\design\java.
22. Press OK.
23. Select QuickStartJava from the tree on the Source Folder Selection dialog.
24. Press the Create New Folder button again. The New Folder dialog appears.
25. Enter realized_java in the Folder name field. The realized_java source folder links in the implementation of the realized domain ExternalDeviceControl domain.
26. Press the Advanced button.
27. Check Link to folder in the file system.
28. Press the Browse button. The Browse for Folder dialog appears.
29. Locate the root directory of your workspace. Then locate the directory SimpleOvenReference/java underneath the workspace.
30. Press OK to dismiss the New Folder Dialog.
31. Press Ok to dismiss the Source Folder Selection dialog.
32. Upon completion of this step the Java Settings page should look like this:



33. Press Finish.

34. A confirmation dialog may appear asking if you want to switch to the Java perspective. Press Yes. The Eclipse perspective changes to Java.

35. The new QuickStartJava project will appear in the Package Explorer. The project will have build problems because the code has not been generated yet.

PROCEDURE: Change Markings to Enable Spotlight Instrumentation

1. In the Package Explorer select ReferenceSimpleOven / Java / properties.txt.
2. Right click and select Copy.
3. Select the QuickStartJava project.
4. Right click and select Paste.
5. Double click on the QuickStartJava / properties.txt. The properties.txt file appears in the Editor pane.
6. Add the following to the top of properties.txt :

```
Domain,SimpleOven.*,SpotlightEnabled,T
```

7. Select File > Save from the main menu to save the changes.

The *markings* file properties.txt controls many aspects of transformation, including Spotlight debugger configuration.

Task 2: Create a PathMATE Project

PROCEDURE: Create a PathMATE Project

Use Rational Software Modeler to create a PathMATE project for your system model:

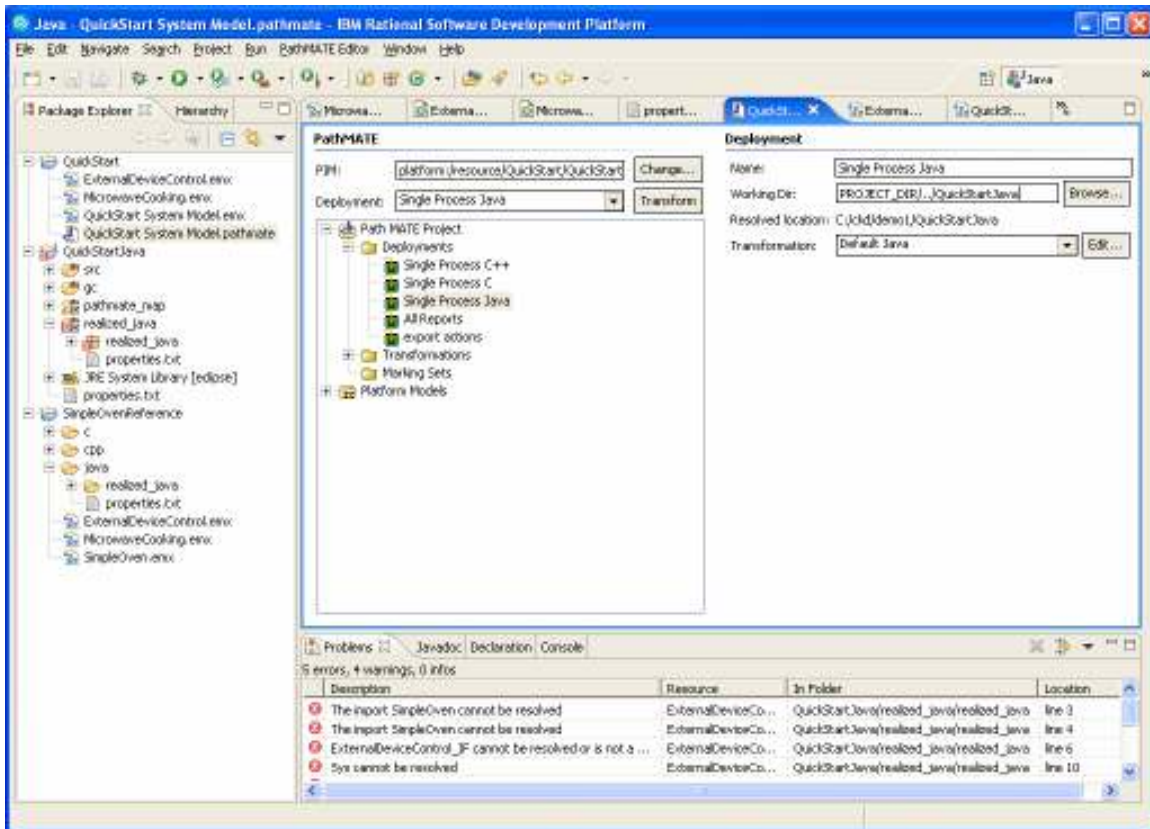
1. Select File > New > Other...

The New dialog box prompts you to select a wizard.

2. Expand PathMATE and select PathMATE Project in the list of wizards. (If the PathMATE folder is not visible, click Show All Wizards.)
3. Click Next. The New PathMATE Project dialog box opens.
4. Expand QuickStart and select the Platform Independent Model named QuickStart System Model.emx.
5. Click Finish. The QuickStart System Model.pathmate opens in the PathMATE editor pane.

The QuickStart System Model.pathmate file is not visible in the Model Explorer, but is visible in the Java and Resource perspectives, under your QuickStart project. The PathMATE editor pane is available regardless of the perspective you open.

6. In the QuickStart System Model.pathmate file, select the Single Process Java deployment from the Deployment dropdown at the top left of the editor.
7. In the PathMATE browser, select PathMATE Project > Deployment > Single Process Java.
8. In the details pane, change the Working Dir to PROJECT_DIR/./QuickStartJava. The figure below shows the PathMATE editor after changing the working directory of the deployment. Changing the working directory changes the directory where the transformation will place the generated files.
9. Select File > Save from the main menu to save the changes.



Task 3: Transform Your Model to Code

PROCEDURE: Generate Java Implementation Code

1. Open QuickStart System Model.pathmate in the editor pane.
2. Verify that the Single Process Java appears in the Deployment pick list.
3. Click Transform.
4. A progress bar appears.
5. Verify in the Console tab transformation was successful.

TIP: If PathMATE encounters errors during transformation, the error messages appear in the Console tab. You may wish to use the Problems viewer to quickly navigate to the source of the error (Window > Show View > Problems). Problems may be caused by errors in the model or incorrect entries in the properties.txt file. Correct the problems identified in the messages, then return to QuickStart System Model.pathmate in the editor pane and click Transform again.

PLEASE NOTE – the transformation maps applied in this procedure were directly controlled by the PIM markings specified in your properties.txt file, which you earlier copied from the ReferenceSimpleOven project java subdirectory.

6. After a successful transformation, the generated java will recompile automatically. All problems should be resolved and the build should be successful.
7. To run with Spotlight you will need to enable Spotlight instrumentation. In the Package Explorer view, expand QuickStartJava > pathmate_map > mechanisms.
8. Double click on PfdDefine.java. The file opens in the Editor pane. Set the NO_PATH_IE to false as shown below:

```
public static final boolean NO_PATH_IE = false;
```

9. Select File > Save from the main menu to save the changes to PfdDefine.java.

Congratulations! You now have a complete Java implementation and project file for your system!

2. Run SimpleOven with Spotlight

Run the new program with the Spotlight to visualize SimpleOven system execution at the model level, and animate aspects of the SimpleOven model within RSM directly via execution.

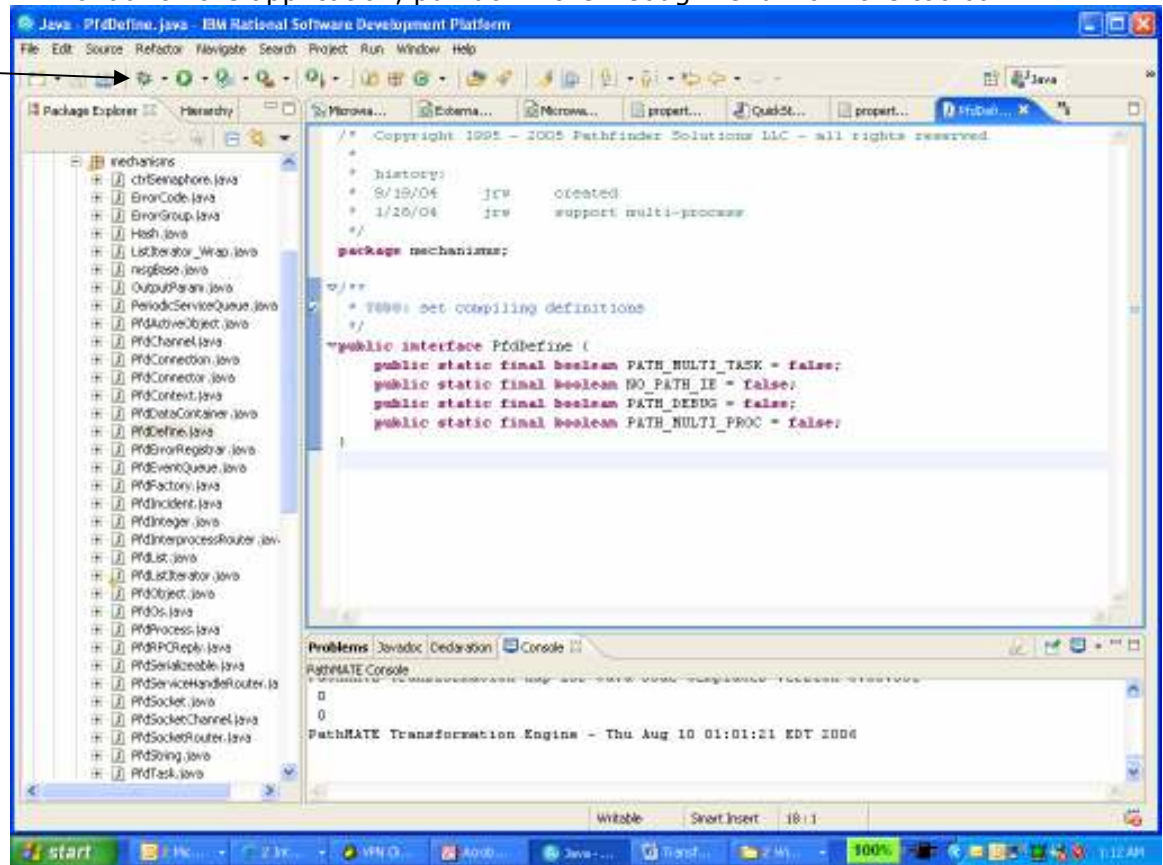
Task 1: Execution Control

Start SimpleOven and Connect to Spotlight

Launch the application and connect it to Spotlight:

1. To launch the application, pull down the Debug menu from the toolbar.

Debug menu



2. Select Debug... from the debug toolbar menu.
3. The Debug dialog appears.
4. Select Java Application from the Configurations tree.
5. Right click and select New.
6. Enter SimpleOven in the Name field.
7. Select QuickStartJava for the Project if it is not selected already.
8. In the Main Class section, press the Search button. The Choose Main Type dialog appears.
9. Select SimpleOvenApp from the Matching Types.
10. Press Ok.
11. Press the Debug button. The executable starts running and the following message is reported to the Console:

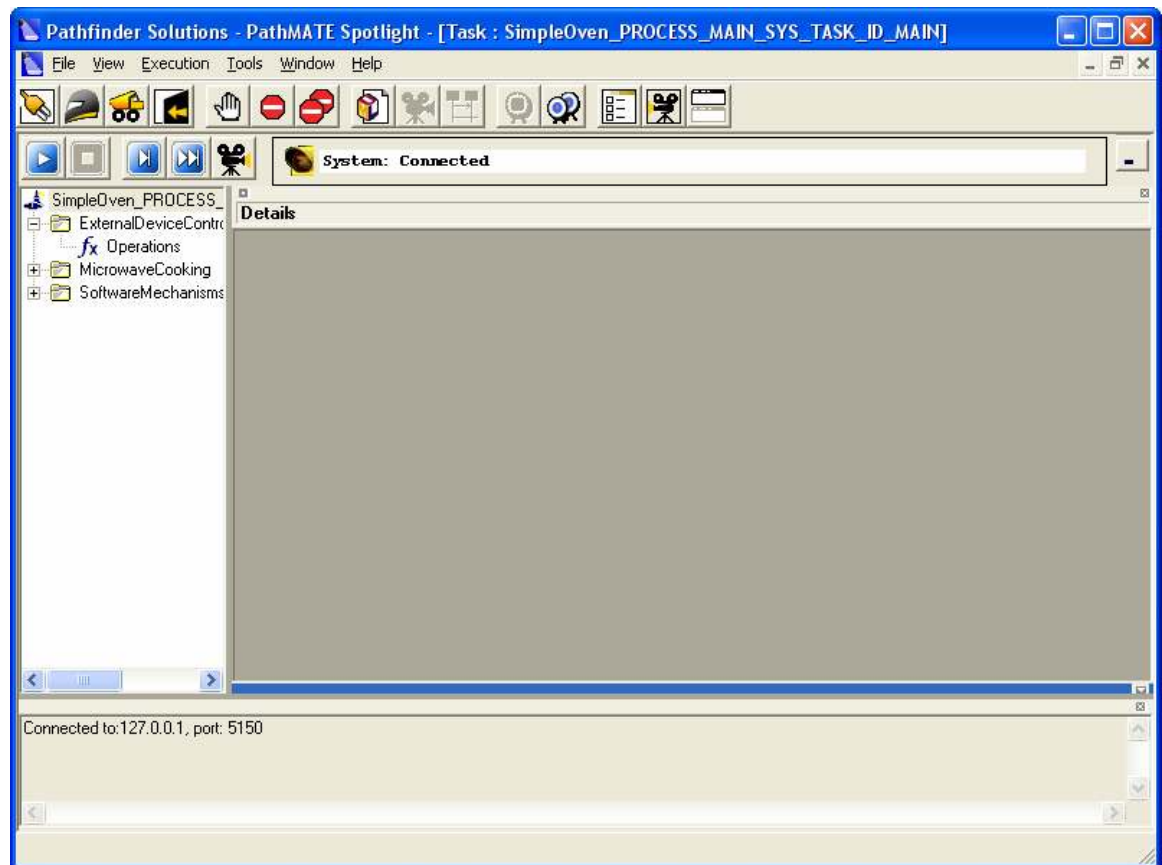
Task waiting for Spotlight connection on port 5150...

12. To launch Spotlight, click in the Navigator(in Java Perspective) or the Model Explorer (in Modeling Perspective) . Select PathMATE > Launch Spotlight from the main menu or choose the Eclipse toolbar Launch Spotlight button. Alternatively you can launch Spotlight from the Windows menu Start > Programs > Pathfinder Solution> Spotlight.

13. Once Spotlight starts click the Connect icon at the left end of the Spotlight toolbar to connect Spotlight to the target application. When Spotlight is successfully connected, the three domains in SimpleOven appear in the browser on the left, and the status bar indicates *System: Connected*. Spotlight should look like the window below.
14. NOTE: If the ExternalDeviceControl and MicrowaveCooking domains are not shown in the browser, Spotlight instrumentation is most likely not enabled for those domains. Click on the red square Terminate button on the Console view to stop SimpleOven. Close the Spotlight window. Switch to the Java perspective by selecting Window > Open Perspective > Java. Expand the QuickStartGuide project. Double click to open properties.txt. Check to see if it contains the line that enables Spotlight instrumentation:

```
Domain, SimpleOven.*, SpotlightEnabled, T
```

If the properties.txt changed, double click on the QuickStartJava > QuickStart System Model.pathmate file. Select the Single Process Java deployment and press the Transform button. The generated code is updated and the java code automatically rebuilds. Rerun SimpleOven and connect to Spotlight again.



Congratulations! You've connected the Simple Oven executable to Spotlight!