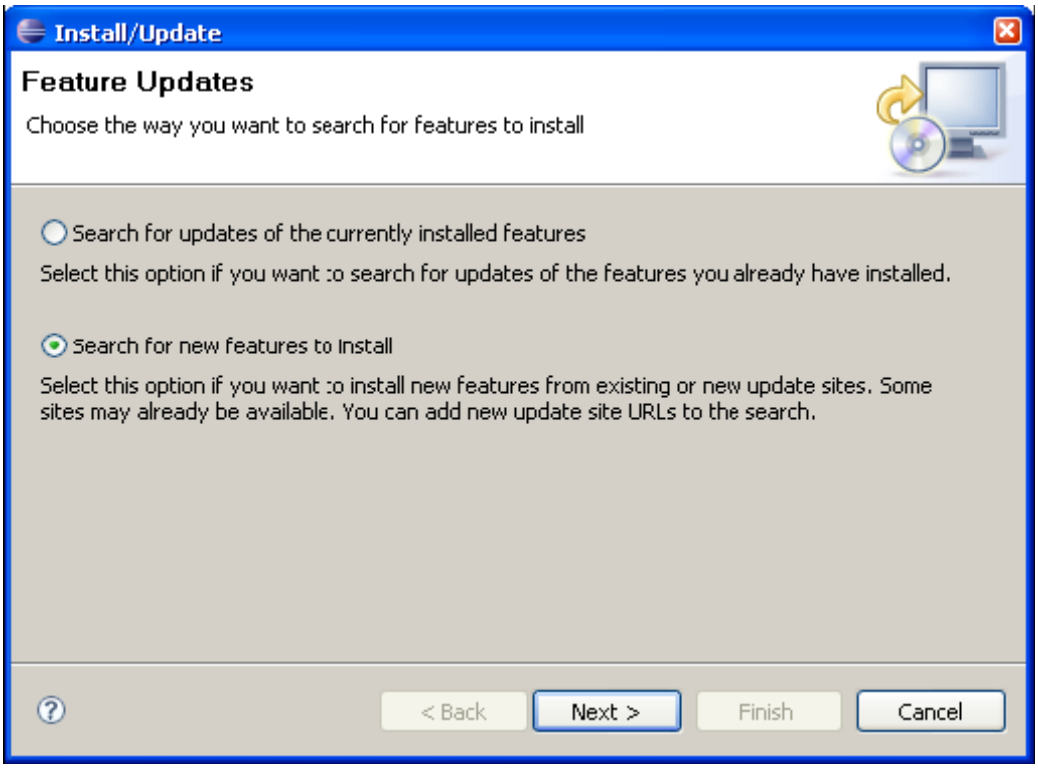
Eclipse JVI Plugin

This is a demonstration of how to use the JVI plugin for eclipse. It shows you how to install and configure the JVI plugin and how to run the example.

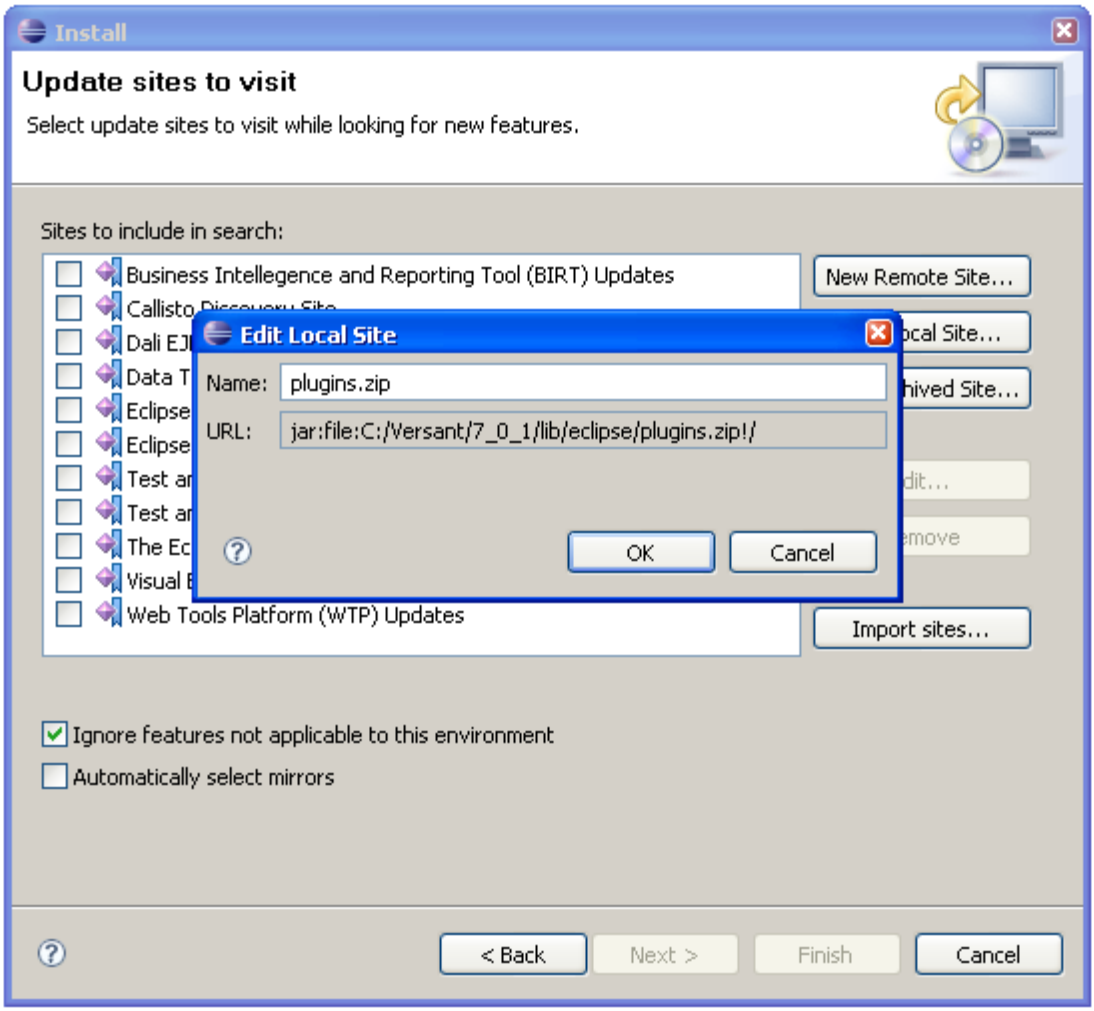
Plugin Installation

The Versant JVI Eclipse plugins are contained in the file plugins.zip in the <VERSANT_ROOT>/lib/eclipse directory in your Versant installation.
Follow the normal Eclipse software update procedures (from the Eclipse **Help | Software Updates | Find and Install...** menu) to install the Versant JVI plugins.

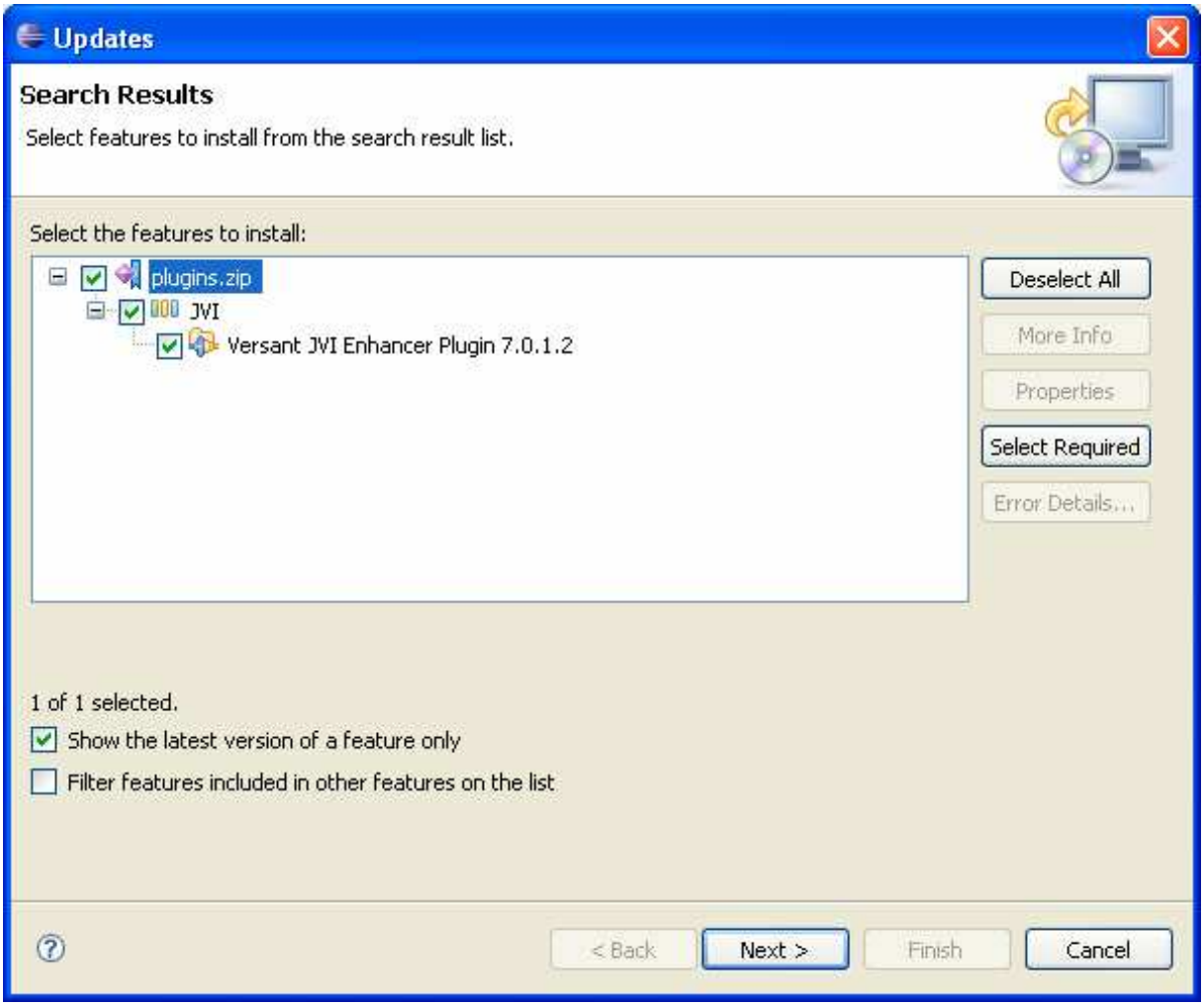


Choose **Search for new features to install** and click **Next >.**

In the next dialog, choose **New Archived Site...** and navigate to the file <VERSANT_ROOT>/lib/eclipse/plugins.zip and click **OK.** The path is added to the sites that Eclipse will use to search for new features.


Install
Update sites to visit
Select update sites to visit while looking for new features.
Sites to include in search:
Business Intellegence and Reporting Tool (BIRT) Updates
Web Tools Platform (WTP) Updates
New Remote Site...
Import sites...
Edit Local Site
Name:
plugins.zip
URL:
jar:file:C:/Versant/7_0_1/lib/eclipse/plugins.zip!/
OK
Cancel
Ignore features not applicable to this environment
Automatically select mirrors
< Back
Next >
Finish
Cancel

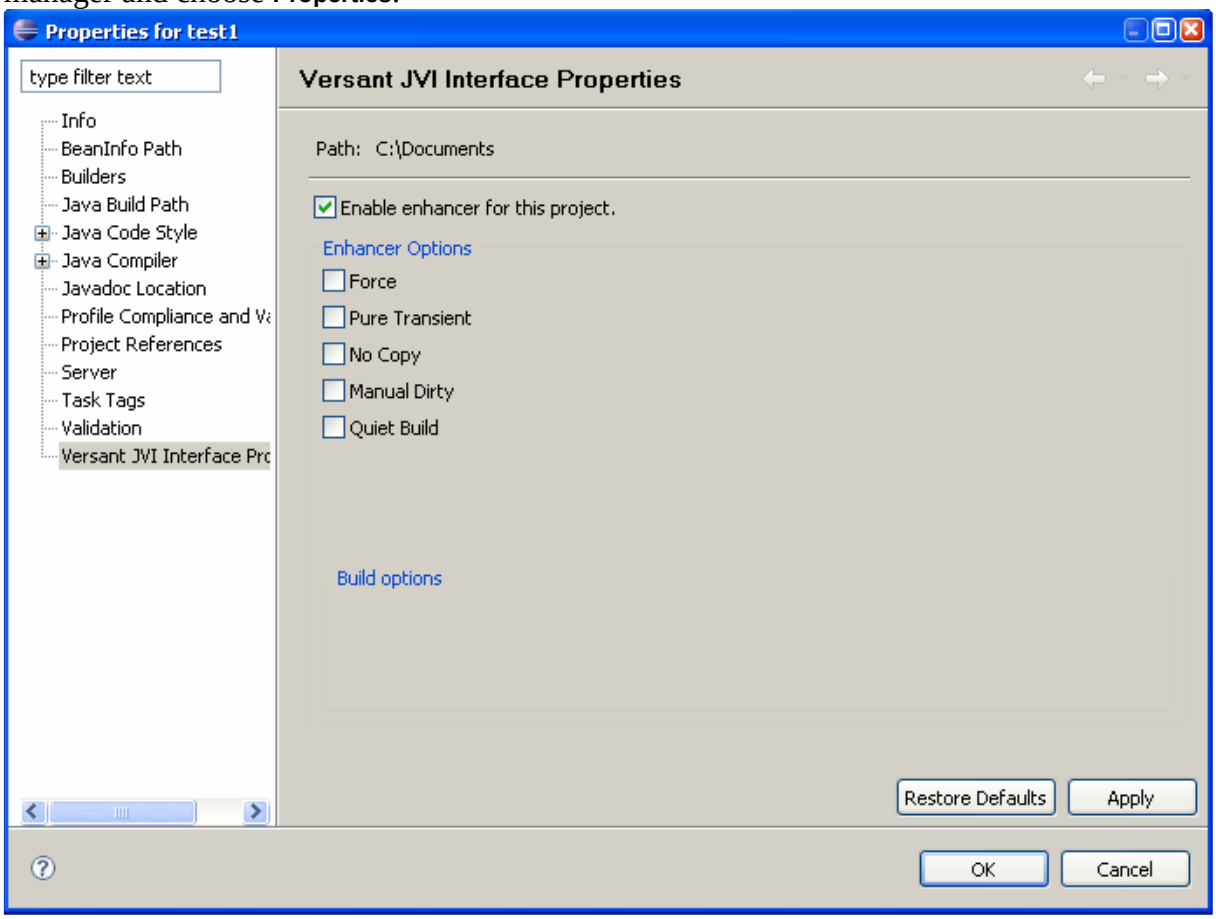
Click **Finish** and wait for the next dialog to come up. It will show you the available plugins to install. Choose to install the JVI enhancer plugin and click **Next >**.



Select the plugins you want to install and click **Next >**. Use the defaults for the rest of the installation.

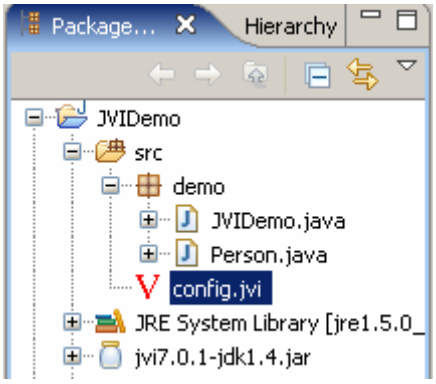
Using the JVI plugin

Open the project you're working on. Then right click on the project within the package manager and choose **Properties.**

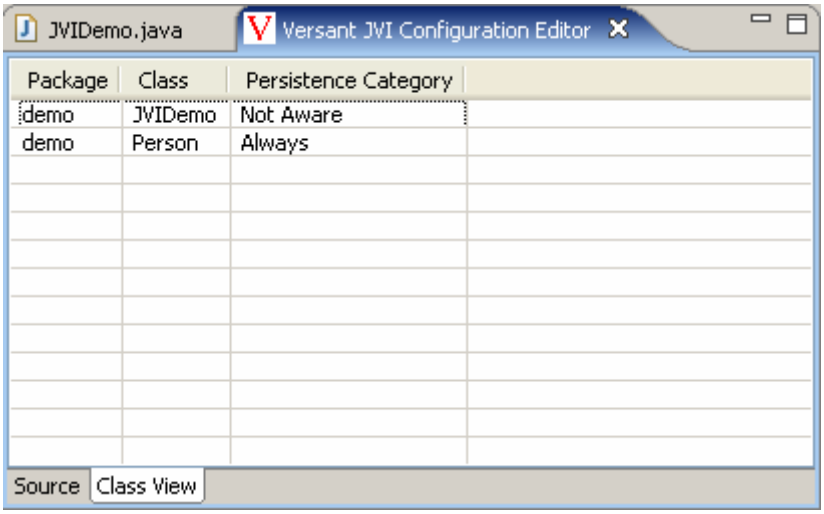


Select **Enable enhancer for this project** and leave the dialog.

Afterwards, create a new file for your current project and call it `config.jvi`. The JVI plugin will recognize the file and a special icon will be mapped to it.



Now you can configure your classes for persistency enhancement by double clicking on the `config.jvi` file. The file will be opened in the editor with tow tabs, the **Source** tab and the **Class View** tab. Choose the **Class View** tab and you will see all the project classes listed.



Now choose the **Persistency Category** for the class of your choice by clicking in the appropriate table cell. Save your changes.

Your project is now set up for automatically enhance your persistent classes. The project can now be compiled and run.

Running the sample application

To run the sample application start Eclipse and import the project <VERSANT_ROOT>/demo/jvi/IDE/eclipse as **Existing Projects Into Workspace**. The imported project will show up in your package explorer with the name JVIDemo.
Choose **Enable enhancer for this project** in **Project | Properties | Versant JVI Interface Properties.** In the same dialog specify the **Configuration File** as <VERSANT_ROOT>/demo/jvi/IDE/eclipse/src/config.jvi and the **Output Directory** to <VERSANT_ROOT>/demo/jvi/IDE/ eclipse /bin .
Also make sure that the build options within that dialog meet your needs and adjust them if necessary.
Afterwards reference the jar file jvi7.0.1-jdk1.4.jar as a user library to your projects build path. To do that right click on JVIDemo in the package explorer and choose **Build Path | Add External Archives.** Now choose the file as <VERSANT_ROOT>/lib/jvi7.0.1-jdk1.4.jar and click ok.

You are now ready to start the example application. Run it with a valid database name as parameter.
The example does 4 things, it:

1. inserts some data into the database and prints the content of the database.
2. deletes one object from the database and prints out the current content.
3. changes one object in database and prints out the current content.
4. cleans up the database by delete all objects.