

---

***AutoVue Integration SDK  
&  
Sample Integration for  
Filesys DMS***

***Installation Guide***

---

# Contents

<b>INTRODUCTION .....</b>	<b>1</b>
<b>SYSTEM REQUIREMENTS .....</b>	<b>2</b>
<b>INSTALLATION PREREQUISITES .....</b>	<b>3</b>
Download the Eclipse WTP 1.5.2 .....	3
Eclipse WTP installation for Solaris .....	3
<b>INSTALLATION .....</b>	<b>5</b>
Install the AutoVue Integration SDK .....	5
Create a New Dynamic Web Project .....	5
Importing a WAR File .....	6
Create a New Dynamic Web Project .....	9
Create Tomcat server .....	15
Add Tomcat Server Runtime to the Project Library .....	19
Define Run Configuration .....	22
Configure the Project .....	23
Configure the web.xml file .....	23
Configure the Applet File .....	24
Configure the jVue.html file .....	25
Debug Mode Configuration .....	25
Running the Sample Integration .....	26
<b>TROUBLESHOOTING .....</b>	<b>30</b>
<b>All Systems .....</b>	<b>30</b>
Setting the Compiler Compliance Level .....	30
Fixing the Bootstrap Problem .....	33
Fixing the localhost Problem .....	34
Fixing the File Not Found Problem .....	34
Fixing Tomcat Server Publishing Failure Problem .....	35
<b>Linux Systems .....</b>	<b>35</b>
Installing Java for Firefox .....	35
<b>FEEDBACK .....</b>	<b>37</b>
General Inquiries .....	37
Sales Inquiries .....	37
Customer Support .....	37



---

# Introduction

This document describes the procedure for building and running a dynamic Web project in the Eclipse Web Tools Platform (WTP) IDE for the AutoVue Integration SDK and the Filesys DMS sample. To create and run the new dynamic Web project, complete the following steps:

- 1 Install the AutoVue Integration SDK.
- 2 [Import a WAR file](#) or [Create a New Dynamic Web Project](#)
- 3 [Copy AutoVue Integration SDK and Filesys DMS application files in the project workspace](#)
- 4 [Create Tomcat server](#)
- 5 [Add Tomcat runtime to project library](#)
- 6 [Create run configuration](#)
- 7 [Run the project](#)

This document also covers [troubleshooting](#).

# System Requirements

The recommended system hardware configuration is:

- A system supporting the JDK/JRE listed below with at least 512MB of main memory (1GB recommended).
- At least 400MB of free disk space to install the software components and examples.

The installation instructions cover the following operations systems:

- Windows® operating systems 2000 and 2003.
- Redhat Linux Enterprise 4
- Solaris 10

## **Windows 2000 and 2003**

- AutoVue Client-Server Edition 19.1
- SUN JDK 1.5.0
- Eclipse Web Tools Platform (WTP) 1.5.2
- Servlet Engine Jakarta Tomcat 5.5.20
- Mozilla Firefox 2.0 for Windows or Microsoft Internet Explorer 6.0 Sp2.

## **Redhat Linux Enterprise 4**

- AutoVue Client-Server Edition
- Sun jdk 1.5.0 for Linux
- Eclipse Web Tools Platform (WTP) 1.5.2
- Servlet Engine Jakarta Tomcat 5.5.20
- Mozilla Firefox 2.0 for Linux

## **Solaris 10**

- AutoVue Client-Server Edition 19.1
- SUN JDK 1.5.0
- Eclipse IDE 3.2.1 and Web Tools
- Servlet Engine Jakarta Tomcat 5.5.20
- Mozilla Firefox 2.0 for Solaris

# Installation Prerequisites

This manual assumes you are familiar with Java development and with basic Web application development concepts, such as deployment descriptors and WAR archives. Understanding XML language is beneficial, but not mandatory.

The software products listed in the [System Requirements](#) must be installed and configured on your system according to the manufacturer's instructions.

Useful commands on UNIX and Linux:

- To extract files from a TAR file, use the tar command: `tar -xvk <tarfile>`
- To unzip GZIP files, use the gunzip command: `gunzip <gzip file>`
- To unzip ZIP files, use the unzip command: `unzip <zip file>`

## Download the Eclipse WTP 1.5.2

If you do not have Eclipse WTP installed on your machine, you can download the WTP-all-in-one module from <http://www.eclipse.org/webtools/> and unzip it in your working directory.

The Eclipse Web Tools Platform extends the Eclipse platform with tools for developing J2EE Web applications. The WTP all-in-one module includes the complete set of software you need to start using WTP immediately. This package already has WTP combined with the complete set of prerequisites. You do not need any thing else.

## Eclipse WTP installation for Solaris

There is no Eclipse WTP all-in-one module for Solaris. Take the following extra steps.

- 1 Download the following files:
  - The current version of the Eclipse driver: `eclipse-SDK-3.2.1-solaris-gtk.zip`
  - The EMF driver used in this build: `emf-sdo-xsd-SDK-2.2.1.zip`
  - The GEF driver used in this build: `GEF-SDK-3.2.1.zip`
  - Java EMF Model Runtime driver used in this build: `JEM-SDK-1.2.1.zip`
  - The Eclipse Web Tools Platform Project used in this build: `wtp-sdk-R-1.5.2-200610261841.zip`

- 2 Complete the installation in the following sequence:
  - a. eclipse-SDK-3.2.1-solaris-gtk.zip
  - b. emf-sdo-xsd-SDK-2.2.1.zip
  - c. GEF-SDK-3.2.1.zip
  - d. JEM-SDK-1.2.1.zip
  - e. wtp-sdk-R-1.5.2-200610261841.zip
- 3 Run Eclipse WTP (<eclipse directory>/eclipse).



# Installation

Take the following installation steps:

- 1 [Install the AutoVue Integration SDK \(page 5\)](#)
- 2 [Create a New Dynamic Web Project \(page 5\)](#)
- 3 [Create Tomcat server \(page 15\)](#)
- 4 [Add Tomcat Server Runtime to the Project Library \(page 19\)](#)
- 5 [Define Run Configuration \(page 22\)](#)
- 6 [Define Run Configuration \(page 22\)](#)
- 7 [Configure the Project \(page 23\)](#)

## Install the AutoVue Integration SDK

The distribution contains three folders: `wins32`, `linux` and `solaris`. Each of these folders contains the installer of the AutoVue Integration SDK 19.1 for the corresponding platform:

- For **Windows** (2000 or 2003), go to the **win32** folder and launch the **setupwin32.exe** file.
- For **Linux** (Redhat Linux Enterprise 4), go to the **linux** folder and launch the **setuplinux.bin** file. Make sure that you have the privileges to run this file. Otherwise, run the command: **chmod +x setuplinux.bin**
- For **Solaris**, go to the **solaris** folder and launch the **setupsolarisSparc.bin** file. Make sure that you have the privileges to run this file. Otherwise, run the command: **chmod +x**

**setupsolarisSparc.bin**

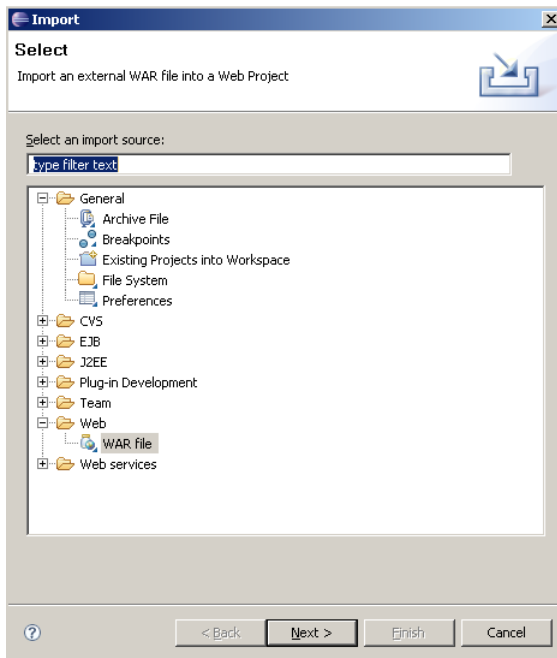
## Create a New Dynamic Web Project

The Sample Integration for Filesys DMS is a dynamic web application. In this section we show you two ways to create the Filesys project. The first one consists of [importing a WAR file](#). The second one consists of [creating a dynamic Web project](#) in the Eclipse WTP IDE.

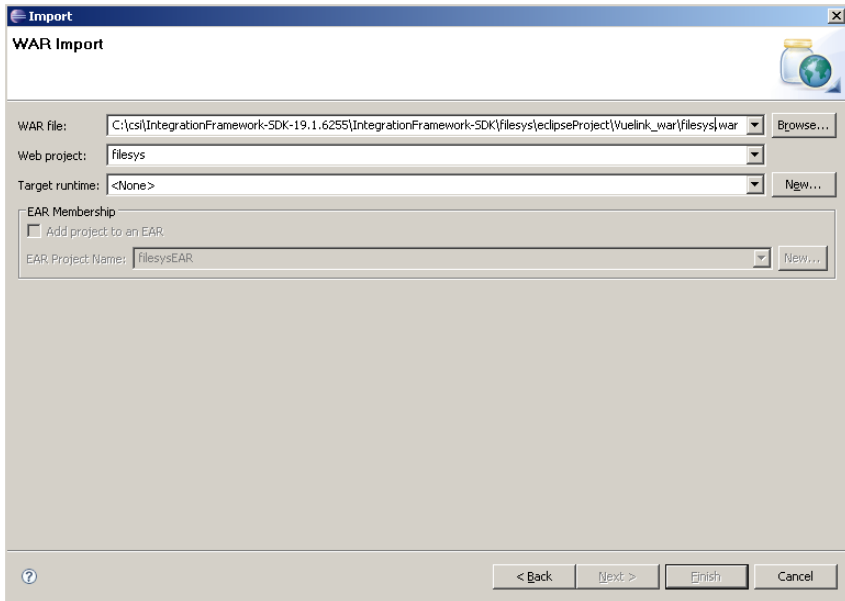
## Importing a WAR File

Take the following steps to create the Filesys web project easily:

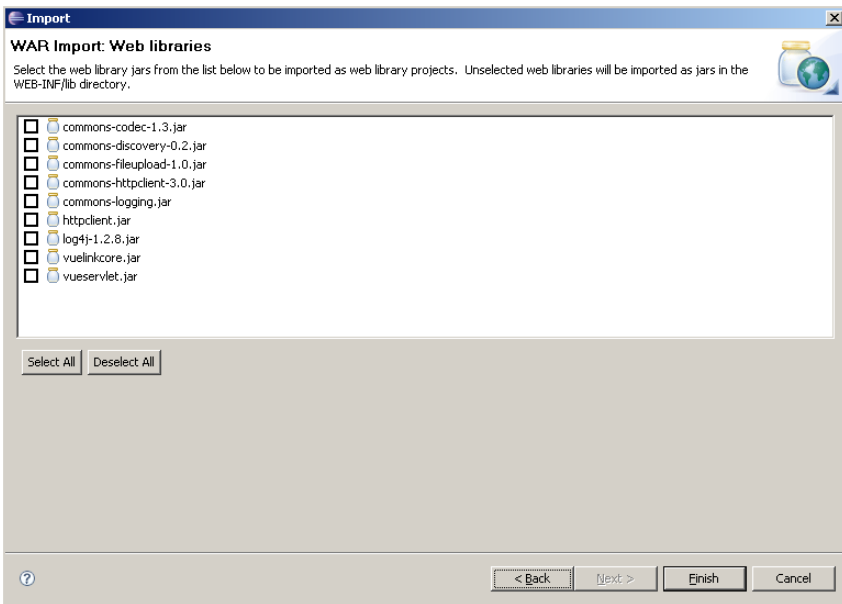
- 1 Select **File > Import**.
- 2 In the **Import** dialog, select **WAR file**, then click **Next**.



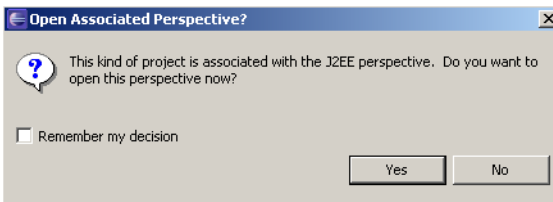
- 3 Click **Browse** to import the **fileSYS.war** file, which is located under **fileSYS\ eclipseProject\VueLink\_war**.



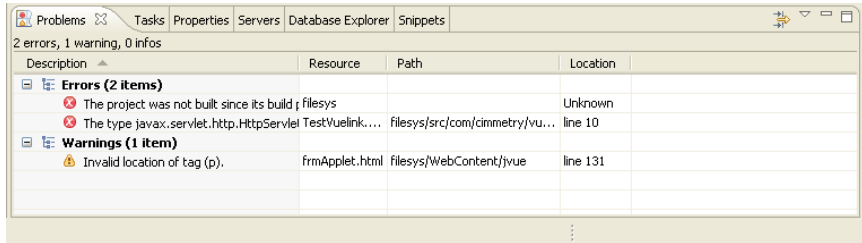
- 4 Click **Finish** to populate the Web project.



- 5 Unzip the **<Autovue Integration SDK installation directory>/fileSYS/database/fileSYSRepository.zip** file in a location of your choice (**<fileSYS data repository>**). The **<fileSYS data repository>** is the repository that will contain the FileSYS DMS data, such as documents, XRefs, markups, and so on.
- 6 Click **Yes** if Eclipse asks you to open the J2EE perspective for this project.



**Note:** At this stage there are still some problems, as shown in the following figure. To fix these problems, take the installation steps in the rest of this document.



If you run into compiler and compliance problems, see section troubleshooting.

This concludes the first approach to creating a dynamic web page. Section [Create a New Dynamic Web Project \(page 9\)](#) shows a different approach; how to create a web page using the Eclipse WTP IDE.

To continue the installation, skip sections [Create a New Dynamic Web Project \(page 9\)](#) and [Copy Filesys DMS Application Files \(page 14\)](#), and go directly to section [Create Tomcat server \(page 15\)](#).

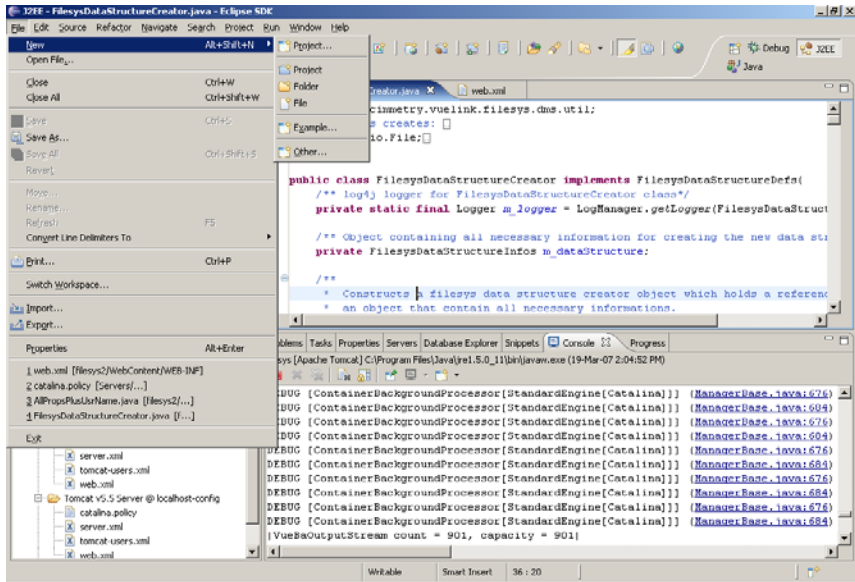
## Create a New Dynamic Web Project

This section describes the procedure for creating a dynamic web project. This procedure is also documented in the Eclipse help environment and it is accessible from the Help Content window.

To create a new dynamic Web project, take the following steps:

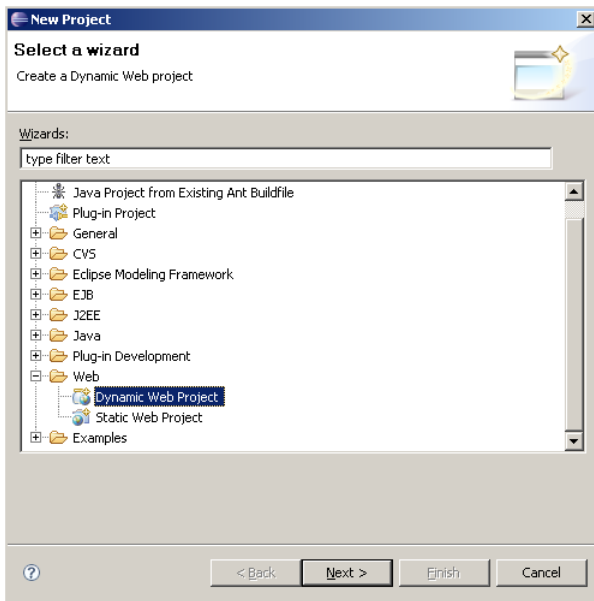
- 1 Open the J2EE perspective.
- 2 Select **File > New > Project** from the main menu.

The New Dynamic Web Project wizard starts.



3 Follow the project wizard prompts:


- a. Open the **Web** folder, select **Dynamic WEB Project**, then click **Next**.



The Dynamic Web Project window appears.

- b. Enter “**fileSYS**” in the **Project name** field, then click **Next**. You can choose a different location from the default one. In this case, clear the **Default Use** checkbox and **browse** to the new location.

### Dynamic Web Project

Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application. 

Project name:

**Project contents:**

Use default

Directory:

**Target Runtime**

**Configurations**

Hint: Get started quickly by selecting one of the pre-defined project configurations.

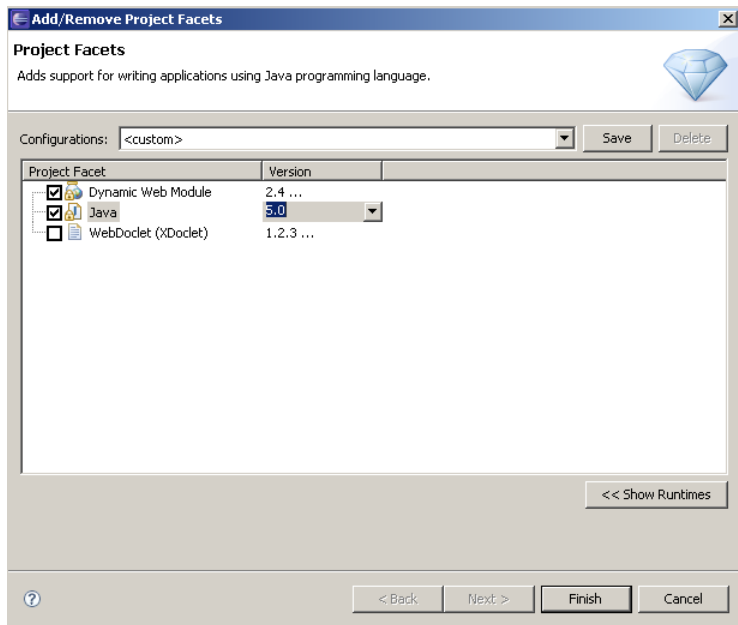
**EAR Membership**

Add project to an EAR

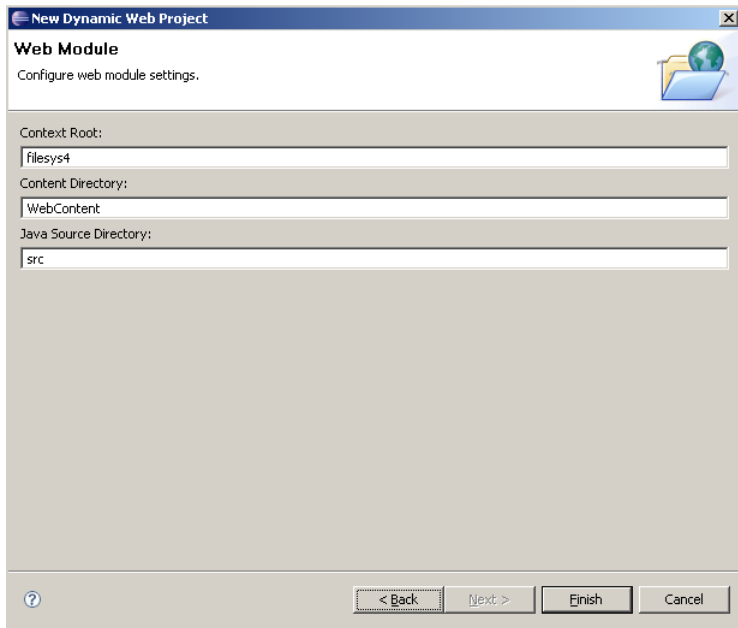
EAR Project Name:



4 Set the **Java** version to **5.0**, then click **Next**.



The Web Module settings display.



5 Click **Finish**.

**Note:** If Eclipse asks you to open the J2EE perspective for this project, click **Yes**.

## Copy Filesys DMS Application Files

To copy the AutoVue Integration SDK and Filesys DMS application files in the workspace project, complete the steps below. All the ZIP files are in the /filesys folder.

- 1 Unzip the **filesys\ eclipseProject\VueLink\_zip\src.zip** file in the **<working directory>\workspace\filesys** folder.
- 2 Unzip the **filesys\ eclipseProject\VueLink\_zip\jvue.zip** file in the **<working directory>\workspace\filesys\WebContent** folder.
- 3 Unzip the **filesys\ eclipseProject\VueLink\_zip\WEB-INF.zip** file in the **<working directory>\workspace\filesys\WebContent** folder.  
**Note:** This action will overwrite the existing WEB-INF directory.
- 4 Unzip **filesys\database\filesysRepository.zip** file in the location of your choice (**<filesys data repository>**).

**Note:**

- The <working directory> is the directory containing the Eclipse workspace directory.
  - The <filesystems data repository> is the directory containing the Filesys DMS data (documents, xrefs, markups, and so on.)
- 5 Return to Eclipse IDE, select the **fileSYS** project, then press **F5** to refresh the project.

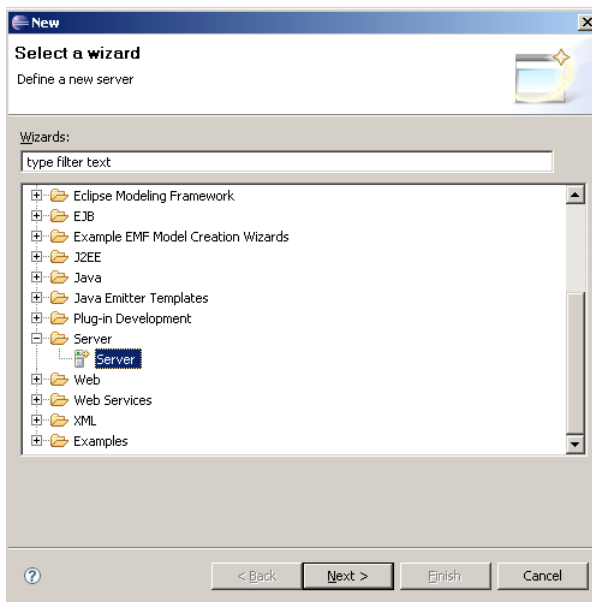
## Create Tomcat server

The Eclipse WTP supports the Apache Tomcat run-time environment, running locally. With Tomcat, you can run the AutoVue Integration SDK project that contains servlets. To set up a server to test the project on Apache Tomcat, complete the following:

You can create a server to identify the runtime environment that you want to use to test your AutoVue Integration SDK project.

To create the Tomcat server in the Eclipse environment, take the following steps:

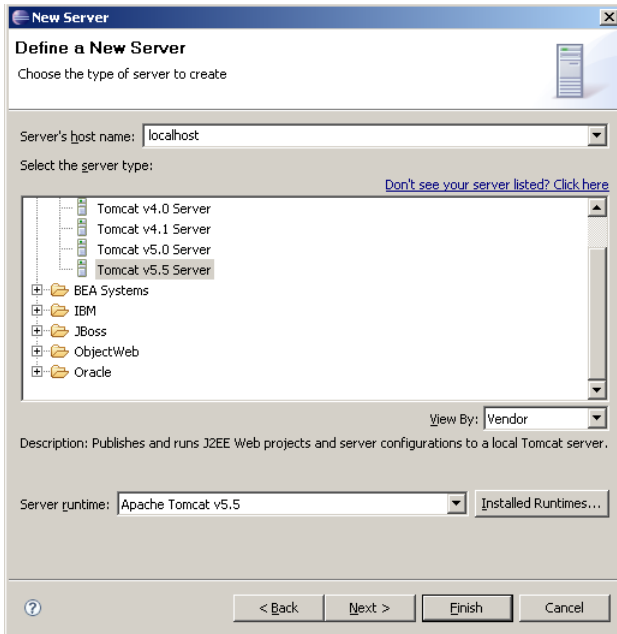
- 1 Select **File > New > Other** from the menu.
- 2 Expand the **Server** folder, then select **Server**.



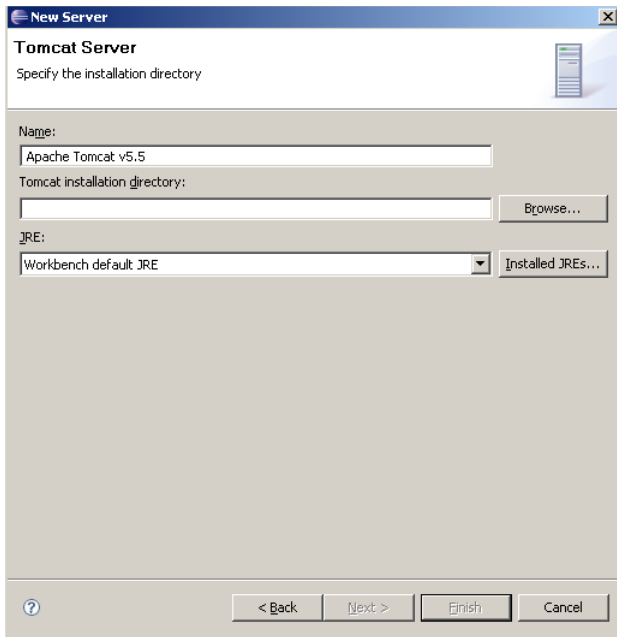
3 Click **Next**.

The **Define a New Server** wizard opens. This wizard lets you define a new server that contains information required to point to a specific runtime environment for local or remote testing, or for publishing to an application server.

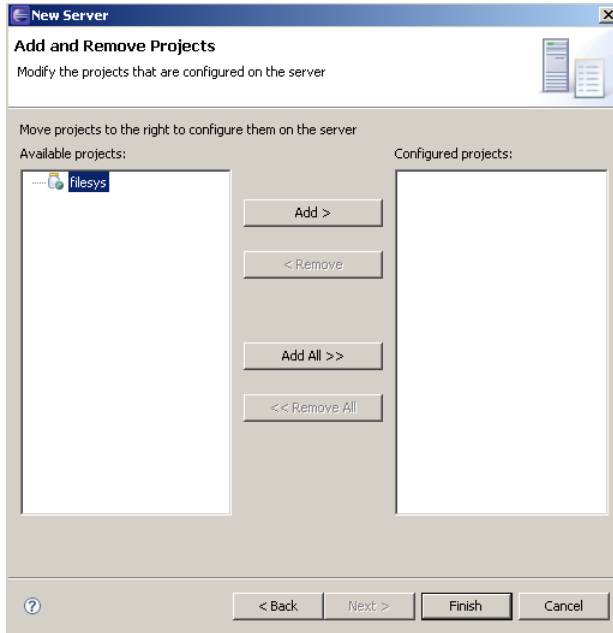
4 Choose the type of server to create. Select **Apache > Tomcat v5.5 Server**. Enter “localhost” in the **Server’s host name** field and click **Next**.



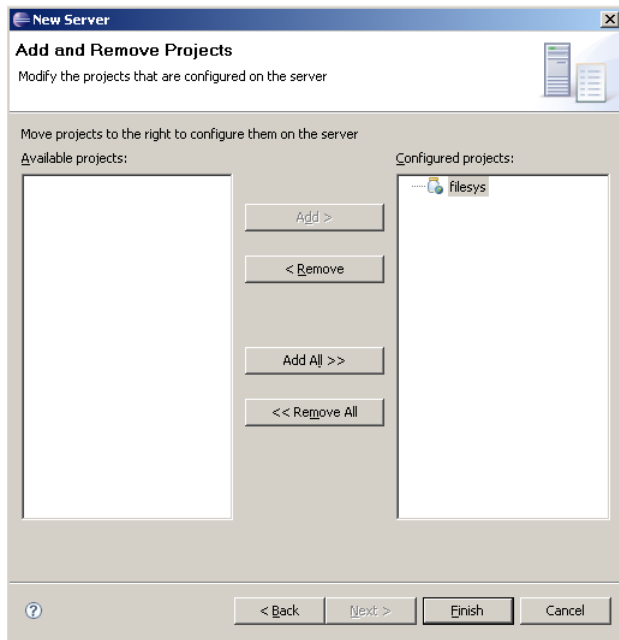
5 Browse to and select the Tomcat location on your system, then click **Next**.



- 6 Select the **fileSYS** project from the **Available projects** list, then click **Add**.



The project is added to the configured projects list.



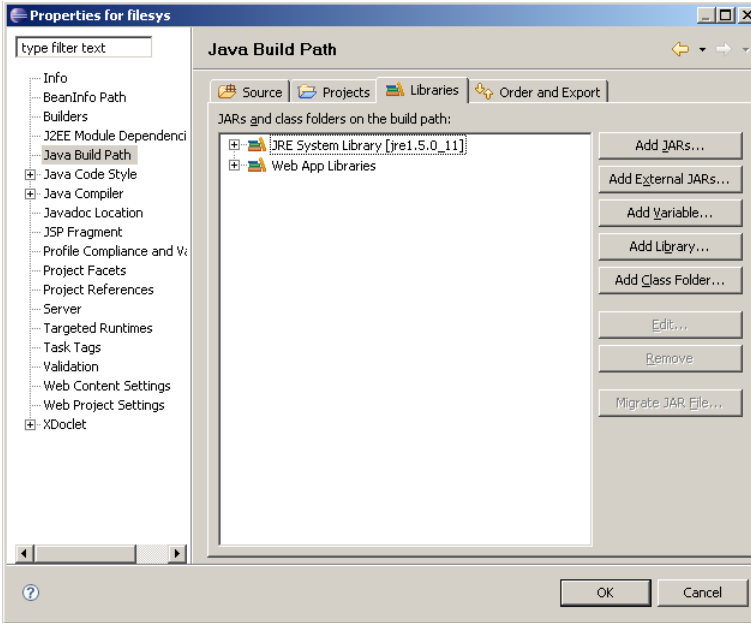
7 Click **Finish**.

## Add Tomcat Server Runtime to the Project Library

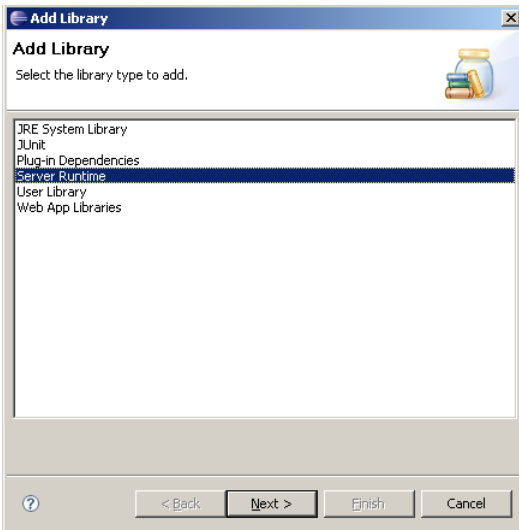
To add Tomcat server runtime to the Filesys project library, take the following steps:

- 1 Select the **fileSYS** project in the project view.
- 2 Right-click and select **properties**.  
The Properties for fileSYS window appears.
- 3 Select **Java Build Path** from the tree.
- 4 Click the **Libraries** tab.

- 5 Click the **Add Library** button.

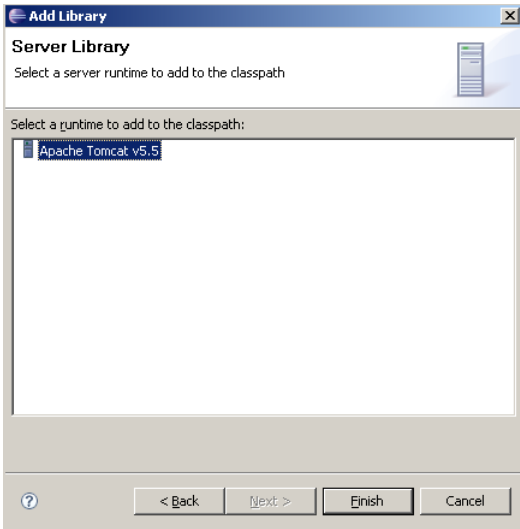


- 6 In the **Add Library** window that appears, select **Server Runtime**, then click **Next**.

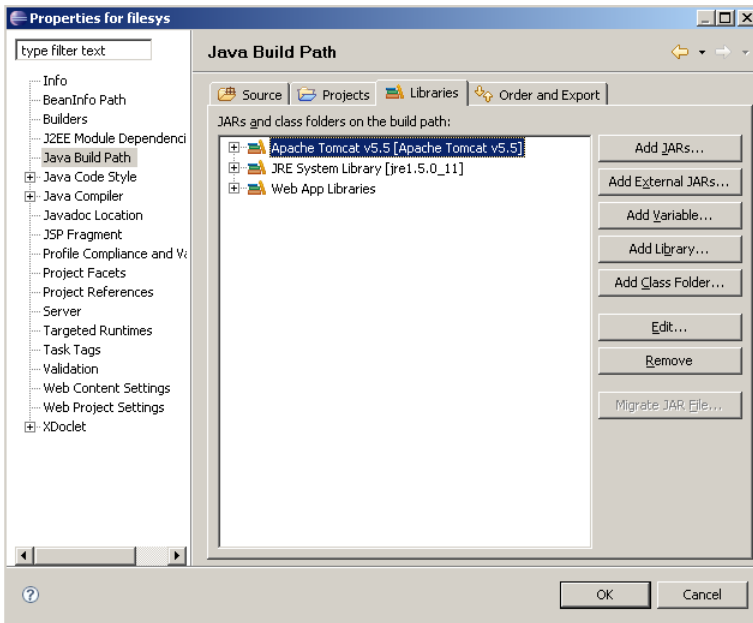




7 Select **Apache Tomcat v5.5**, then click **Finish**.



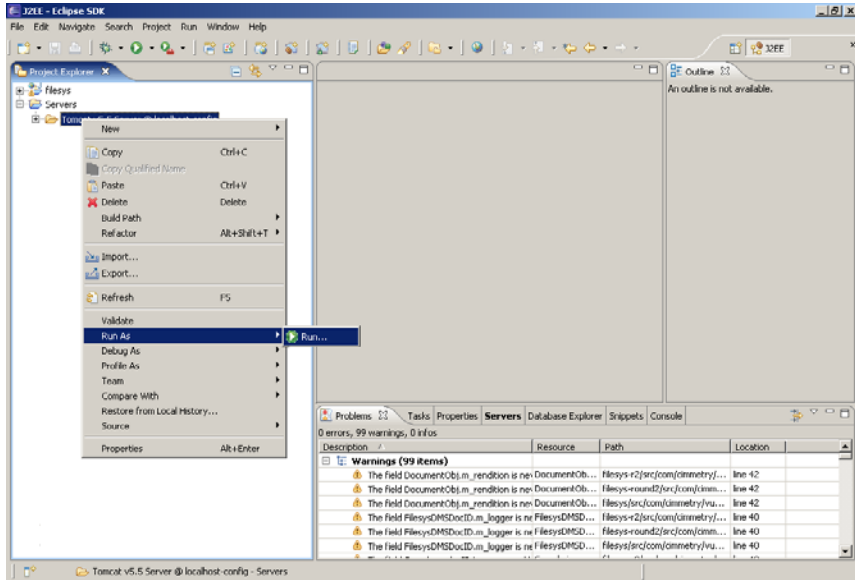
8 In the Properties for filesystem window, click **OK**.



# Define Run Configuration

You can use the workbench to test the **filesys** project artifacts on a server.

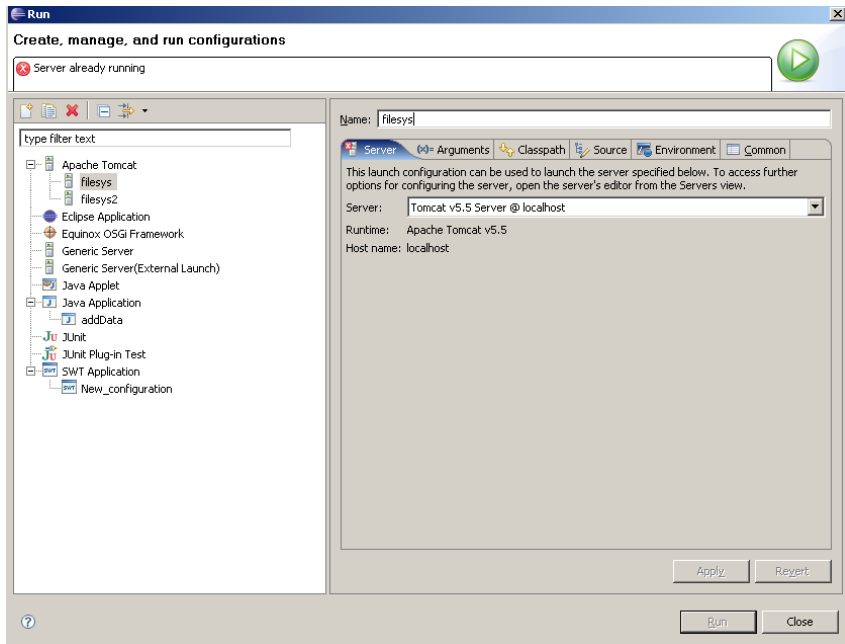
- 1 In the **Project Explorer** view from the J2EE perspective, expand **Servers**.
- 2 Right-click **Tomcat v5.5 Server**, then select **Run As > Run** from the menu.



The Server selection wizard appears.

- 3 In the view on the right, right-click **Apache Tomcat**, then select **New** from the menu.

- 4 Enter the project name, **filesys**, in the **Name** field.



- 5 Select **Tomcat v5.5 Server** from the **Server** drop-down menu.
- 6 Click **Apply**, then click **Close**.

## Configure the Project

To successfully run the Filesys project, complete the following steps:

- 1 [Configure the web.xml file \(page 23\)](#)
- 2 [Configure the Applet File \(page 24\)](#)
- 3 [Configure the jVue.html file \(page 25\)](#)

## Configure the web.xml file

- 1 With your preferred file editor, open the **web.xml** file located at **<working directory>/workspace/filesys/webContent/WEB-INF** and set the new Filesys DMS data in two places: FilesysVuelink servlet init-parameters and the ListDirServlet servlet init-parameters, as shown in the following steps.

- 2 Locate the FilesysVuelink servlet init-parameters block:

```
<init-param>
<param-name>RootDir</param-name>
<param-value>"Target location of the data repository"</param-value>
</init-param>
```

Change the location of the data repository (shown in **bold**) to the location you had previously chosen (**<fileys data repository>**).

- 3 Locate the ListDirServlet servlet init-parameters block:

```
<init-param>
<param-name>RootDir</param-name>
<param-value>"Target location of the data repository"</param-value>
</init-param>
```

Change the location of the data repository (shown in **bold**) to the location you had previously chosen (**<fileys data repository>**).

- 4 Locate the VueServlet servlet mapping block:

```
<init-param>
<param-name>JVueServer</param-name>
<param-value>"jVue Server host:port"</param-value>
</init-param>
```

Change the jVueServer hostname and the set the port to **5099**.

## Configure the Applet File

- 1 With your preferred file editor, open the **frmApplet.html** file located at **<working directory>/workspace/WebContent/jvue/**.
- 2 Locate the following block:

```
<PARAM NAME="JVUESERVER" VALUE="http://
_APPSERVER_HOST_:_APPSERVER_PORT_/fileys/servlet/
VueServlet;socket://_JVUESERVER_HOST_:_JVUESERVER_PORT_">
```

- a. Change the **\_APPSERVER\_HOST\_** parameter and set the **\_APPSERVER\_PORT\_** parameter to **8080**.

- b. Change `_JVUESERVER_HOST_` parameter and set the `_JVUESERVER_PORT_` to **5099**.
- 3 Locate the following block:

```
<PARAM NAME="DMS" VALUE="http://  
_APPSERVER_HOST_:_APPSERVER_PORT_/filesystem/servlet/  
com.cimmetry.vuelink.filesys.FilesysVuelink">
```

Change the `_APPSERVER_HOST_` parameter and set the `_APPSERVER_PORT_` parameter to **8080**.

## Configure the jVue.html file

- 1 With your preferred file editor, open the `jVue.html` file located at `<working directory>/workspace/WebContent/jvue/`.
- 2 Locate the following block:

```
<frame src="http://_APPSERVER_HOST_:_APPSERVER_PORT_/filesystem/  
servlet/com.cimmetry.vuelink.filesys.dms.gui.ListDirServlet" name="Directory"  
marginheight="0" marginwidth="0" />
```

Change the `_APPSERVER_HOST_` parameter and set `_APPSERVER_PORT_` to **8080**.

## Debug Mode Configuration

- 1 With your preferred file editor, open the `log4j.properties` file located at `<working directory>/workspace/WebContent/WEB-INF/lib`.
- 2 In line 11, set the location and the filename of your log4j logging file. Copy the `log4j.properties` to `<Tomcat Directory>/common/ classes`.

```
10 # setting the logging file  
11 log4j.appender.R.File=<working directory>/<logs directory>/<logfile>.log
```

**Note:** The `log4j.properties` must be in the `WEB-INF` folder when you start the server.

The sample integration for Filesys DMS uses the apache log4j package for logging. Default configurations are set in log4j.properties inside WEB-INF\lib folder of the sample integration application.

**Note:** You can change the level and location of output by modifying this file. The following figure shows the different levels of logging available:

Will Output Messages Of Level

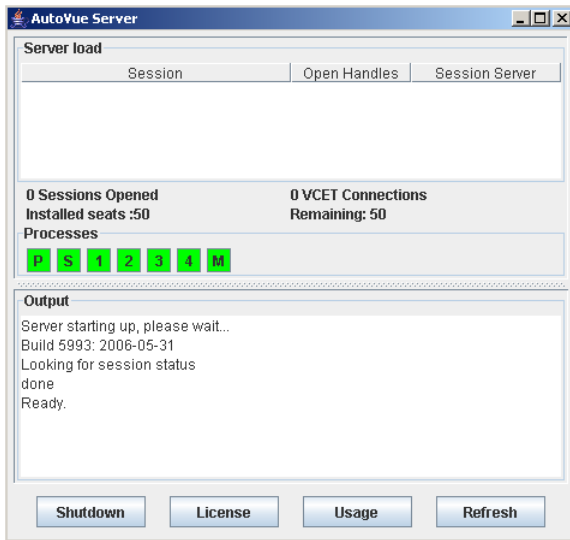
	DEBUG	INFO	WARN	ERROR	FATAL
Debugger Level	DEBUG	INFO	WARN	ERROR	FATAL
DEBUG	Green	Green	Green	Green	Green
INFO	Red	Green	Green	Green	Green
WARN	Red	Red	Green	Green	Green
ERROR	Red	Red	Red	Green	Green
FATAL	Red	Red	Red	Red	Green
ALL	Green	Green	Green	Green	Green
OFF	Red	Red	Red	Red	Red

For example, if you want to elevate the log to DEBUG level, then set **log4j.logger.com.cimmtry.vuelink=DEBUG** inside the previously-mentioned file.

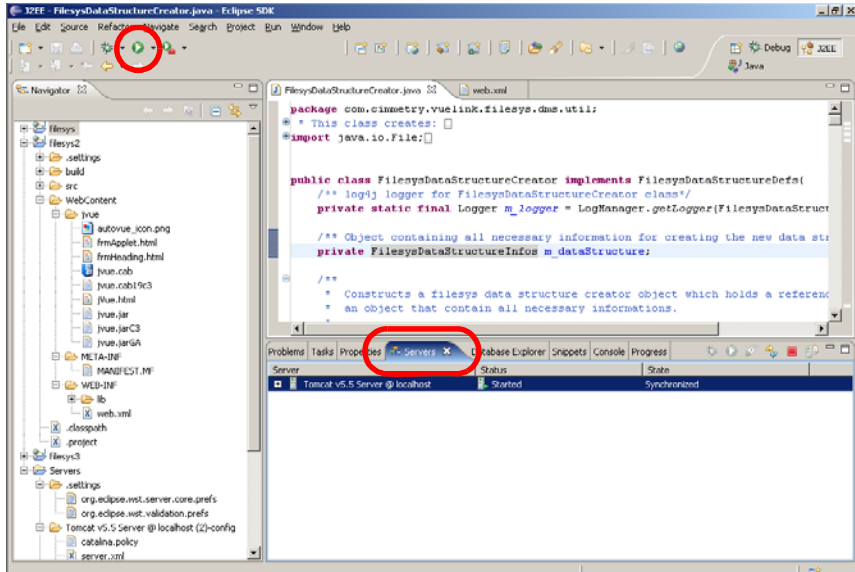
For more information on log4j capabilities, refer to log4j documentation.

## Running the Sample Integration

- 1 Startup the AutoVue Server.

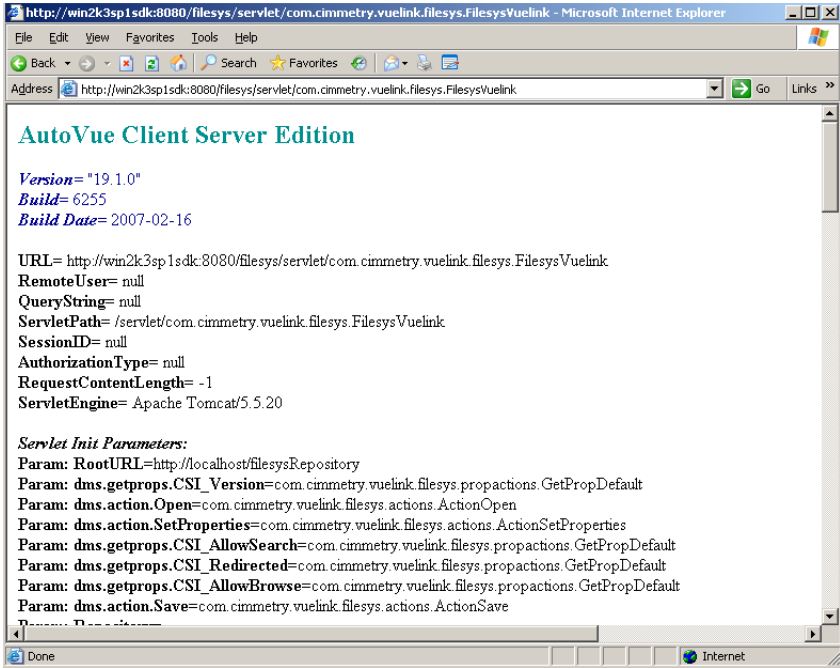


- In the **Servers** view of the **filesys** project (inside Eclipse IDE), run the Tomcat server.



- To verify that the Servlet (`com.cimmetry.vuelink.filesys.FilesysVuelink`) is running properly, launch your Web browser and enter the URL pointing to the Servlet alias name, which you assigned when installing `filesys\ eclipseProject\Vuelink_war\filesys.war` into the application server. Here is an example of a URL: **`http://win2k3sp1sdk:8080/filesys/servlet/com.cimmetry.vuelink.filesys.FilesysVueLink`**  
The following figure shows a sample response if the sample integration is running properly. If you do not get a response similar to the one shown in the

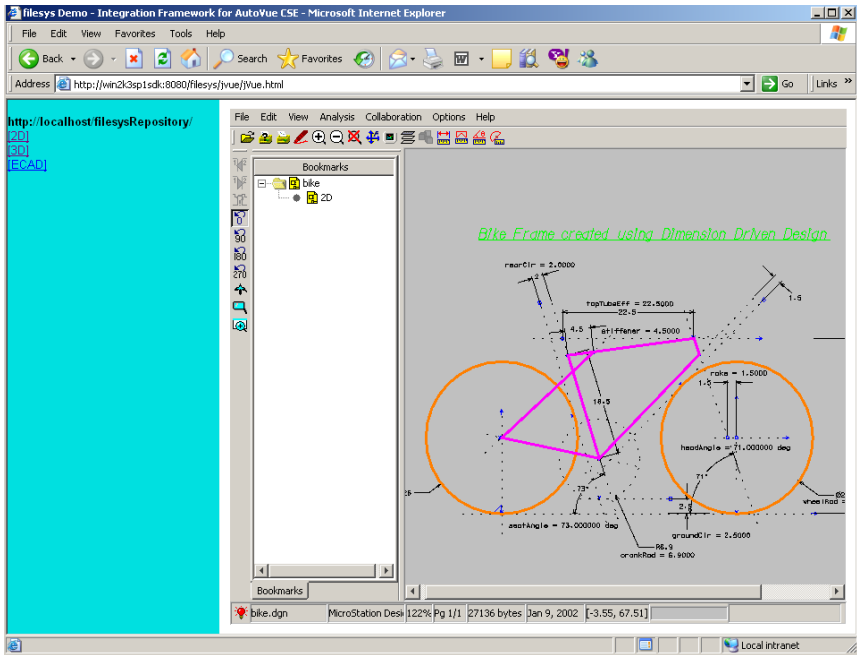
following figure, verify that the Servlet is installed and deployed properly and that your application server is running and is functioning properly.



- 4 Launch an Internet browser and enter the jVue.html address:  
**http://<VueApplet host:8080/filesys/jvue/jVue.html**



Two frames display in the browser. The frame on the left is an interface that lets you navigate the Filesys DMS repository. The frame on the right displays the AutoVue applet.



# Troubleshooting

This section involves troubleshooting for [All Systems \(page 30\)](#) and for [Linux Systems \(page 35\)](#).

## All Systems

This section addresses the following issues:

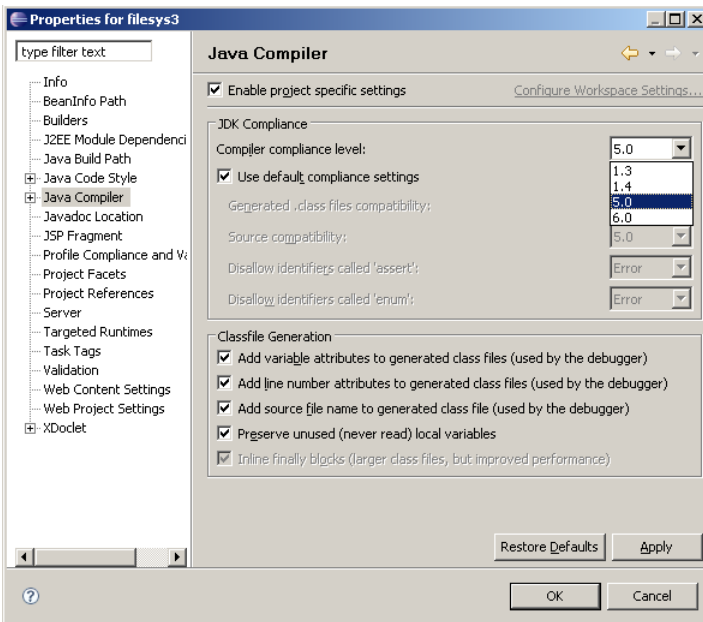
- [Setting the Compiler Compliance Level \(page 30\)](#)
- [Fixing the Bootstrap Problem \(page 33\)](#)
- [Fixing the localhost Problem \(page 34\)](#)
- [Fixing the File Not Found Problem \(page 34\)](#)
- [Fixing Tomcat Server Publishing Failure Problem \(page 35\)](#)

## Setting the Compiler Compliance Level

Take the steps in this section only if you run into compiler compliance level problems. You can set the right compliance level of the compiler from the **Properties for fileys** window.

- 1 In the view on the left, click **Java Compiler**.

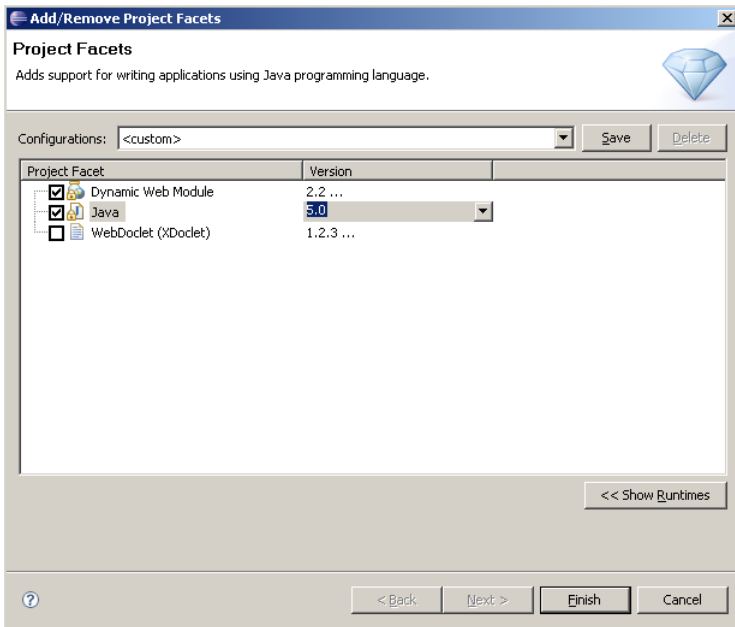
## 2 Set the **Compiler compliance level** to **5.0**.



Check the compatibility between the compiler compliance level and the filesys project facets. You can change the version of a facet in a J2EE project by editing the facets for the project. Changing the Java compiler version of a J2EE project involves changing the version of the **Java** facet.



The **Add/remove Project Facets** window appears.



- 4 Set the **Java version** to **5**.
- 5 Click **OK**.

**Note:** To indicate that you want the Workbench to perform incremental builds whenever resources are saved, select the **Build automatically** checkbox from the **Project** menu.

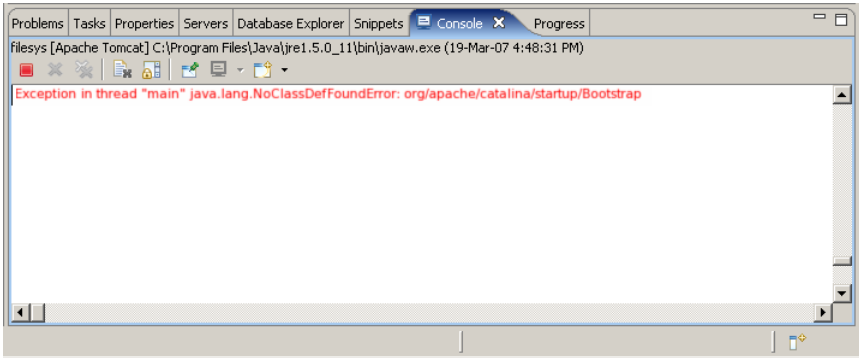
## Fixing the Bootstrap Problem

If you run Tomcat and there is the following error message:

```
Exception in Thread "Main" java.lang.NoClassDefFoundError: org/apache/Catalina/startup/Bootstrap
```

You can fix the problem by adding two JAR files to the ClassPath Server:

- bootstrap.jar from TOMCAT\_HOME/bin
- tools.jar from JAVA\_HOME/lib

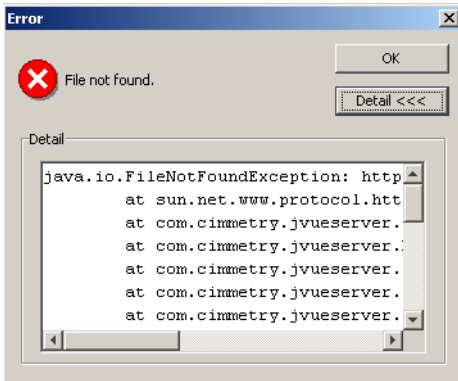


## Fixing the localhost Problem

When you start Tomcat server from Eclipse and then you try the link: **http://localhost:TomcatPort**. It may display **404 error**. This is the correct behavior, because in the Tomcat server the configuration property **catalina.home** points to your Tomcat installation, but **catalina.base** points to a **.metadata.plugins\org.eclipse.wst.server.core\tmp?**, which is a directory under your workspace. Thus, you are running a separate instance of Tomcat. The **webapps** directory under the **.metadata...\tmp?** directory contains only an “empty” ROOT **webapp** plus any web projects you have added to the server. This is why you get the **404 error** message.

## Fixing the File Not Found Problem

If you get the error message **File not found** when you try to view a file, check that you have done all the required modifications correctly in **vue.html**, **frApplet.html**, and **web.xml** files.



## Fixing Tomcat Server Publishing Failure Problem

When you start the Tomcat server and see the error message **Publishing failure**, the Tomcat server configuration at **/server/Tomcat 5.5 Server @localhost-config** is probably missing. To fix it, delete the Server and re-install it, following the steps in the installation procedure.

## Linux Systems

### Installing Java for Firefox

On Linux, Firefox requires JRE 1.4.2 or later. Firefox is compiled with **gcc 3.2.3**, so a compatible version of the Java plugin must be used. JRE 1.4.2 or later contains a compatible plugin. To install this plugin take the following steps:

- 1 Open a terminal.
- 2 Change to your Firefox plugins directory.
- 3 Issue the following command:  
**In -s /usr/java/j2re1.5.0\_10/plugin/i386/ns7/libjavaplugin\_oji.so**

**Note:** Make sure that the link is done by root user in order to be valid for all users.





# Feedback

Cimmetry Systems products are designed according to your needs. We would appreciate your feedback, comments or suggestions. Contact us by fax, e-mail or telephone. There is a feedback button on our Web page that activates an easy-to-use feedback form. Let us know what you think.

## General Inquiries

**Telephone:** +1 514-735-3219

**Fax:** (514) 735-6440

**E-mail:** [info@cimmetry.com](mailto:info@cimmetry.com)

**Web Site:** <http://www.cimmetry.com>

## Sales Inquiries

**Telephone:** +1 514-735-3219 or 1-800-361-1904

**Fax:** (514) 735-6440

**E-mail:** [sales@cimmetry.com](mailto:sales@cimmetry.com)

## Customer Support

**Telephone:** +1 514-735-9941

**Web Site:** <http://www.cimmetry.com/support>

