



---

## ***Visualization Environment for Rich Data Interpretation (VERDI 1.3): Developer Instructions***

U.S. EPA Contract No. EP-W-05-045, “Operation of the Center for Community Air Quality Modeling and Analysis”

---

Prepared for: William Benjey and Donna Schwede  
U.S. EPA, ORD/NERL/AMD/APMB  
E243-04  
USEPA Mailroom  
Research Triangle Park, NC 27711

Prepared by: Liz Adams and Qun He  
Institute for the Environment  
The University of North Carolina at Chapel Hill  
137 E. Franklin St., CB 6116  
Chapel Hill, NC 27599-6116

Date: June 18, 2010

## Contents

<b>1</b>	<b>Introduction.....</b>	<b>1</b>
1.1	Background .....	1
<b>2</b>	<b>Install developer environment .....</b>	<b>2</b>
2.1	Download and Install Eclipse .....	2
2.2	Download and install NSIS Installer (only for windows).....	2
2.3	Install Subclipse update 1.2x in Eclipse 3.5 .....	2
2.4	Install Java Development Kit.....	2
<b>3</b>	<b>Running Eclipse .....</b>	<b>3</b>
3.1	Import VERDI Source Code.....	5
3.1.1	Select File→Import.....	5
3.1.2	Checkout Projects from SVN.....	6
3.1.3	Create a new repository location.....	7
3.1.4	Specify location of VERDI SourceForge repository .....	8
3.1.5	Select Folders for Checkout.....	9
<b>4</b>	<b>Run VERDI within Eclipse .....</b>	<b>11</b>
<b>5</b>	<b>Prepare to build Distribution.....</b>	<b>12</b>
5.1.1	build.properties file .....	12
5.1.1.1	Microsoft Windows .....	12
5.1.1.2	Linux .....	12
5.1.2	Build_dist.xml.....	13
<b>6</b>	<b>Build VERDI .....</b>	<b>15</b>
6.1	Build using Ant .....	15
6.1.1	Microsoft Windows .....	15
6.1.1	Linux Distribution.....	17
6.2	Check Console for Error Messages.....	18
6.3	Add Java Compiler to Ant .....	18
<b>7</b>	<b>Updating source code from the repository .....</b>	<b>21</b>
7.1	Open the Synchronization Window .....	21

7.2 Synchronize with Repository using SVN .....	23
7.3 Resolve conflicts and Update.....	25

## Figures

Figure 3-1 Select a Workspace .....	3
Figure 3-2 Eclipse Starting Up .....	4
Figure 3-3 Eclipse Workbench .....	4
Figure 3-4 File Import.....	5
Figure 3-5 Click on Arrow to see workbench.....	6
Figure 3-6 Checkout Projects from SVN on Windows/Linux .....	7
Figure 3-7 Checkout Projects from SVN on Mac .....	7
Figure 3-8 Create new repository location.....	8
Figure 3-9 Checkout Code .....	9
Figure 3-10 Select Folders for Checkout .....	10
Figure 3-11 SVN Checkout includes a meter indicating % completion .....	10
Figure 3-12 SVN imported code into workspace .....	11
Figure 4-1 Run VERDI within Eclipse .....	11
Figure 5-1 Review/Edit build.properties.....	12
Figure 5-2 build_dist.xml – no editing needed .....	13
Figure 5-3 build_dist.xml: build Linux distribution section .....	14
Figure 5-4 dist.xml: nsis installer path section .....	14
Figure 6-2 Window→Show View→Ant .....	16
Figure 6-3 Double click on build.win.dist to build VERDI distribution .....	17
Figure 6-4 Console Error Message .....	18
Figure 6-5 Open Windows→Preferences .....	19
Figure 6-6 Expand Ant, Select Runtime, Select Global Entries .....	20
Figure 6-7 Add tools.tar to Ant Preferences .....	20
Figure 7-1 Show View → Other .....	21
Figure 7-2 Expand Team Folder - highlight Synchronize – click ok .....	22
Figure 7-3 Synchronize Window added to bottom of Workspace.....	22
Figure 7-4 Click on Synchronize Symbol to bring up Pop-up.....	23
Figure 7-5 Synchronize Pop-up – select SVN .....	24
Figure 7-6 Synchronize SVN – select all.....	24
Figure 7-7 Check for Updates and Conflicts .....	25
Figure 7-7 Check for Updates and Conflicts .....	25



# 1 Introduction

## 1.1 ***Background***

This manual contains instructions on how developers can set up, run, build, and obtain updates from the software repository for Visualization Environment for Rich Data Interpretation (VERDI). Developers are encouraged to develop and contribute code for VERDI. Developers are requested to submit a bugzilla request for enhancement and to submit test datasets, documentation and proposed code to CMAS for testing prior to committing code to the repository.

Initial development of VERDI was done by the Argonne National Laboratory for the U.S. Environmental Protection Agency (EPA) and its user community. Argonne National Laboratory's work was supported by the EPA through U.S. Department of Energy contract DE-AC02-06CH11357. Further development has been performed by the University of North Carolina Institute for the Environment under U.S. EPA Contract No. EP-W-05-045 and EP-W-09-023, by Lockheed Corporation under U.S. EPA contract No. 68-W-04-005, and Argonne National Laboratory. VERDI is licensed under the Gnu Public License (GPL) version 3, and the source code is available through verdi.sourceforge.net. VERDI is supported by the Community Modeling and Analysis System (CMAS) Center under U.S. EPA Contract No. EP-W-09-023. The CMAS Center is located within the Institute for the Environment at the University of North Carolina at Chapel Hill.

## **2 Install developer environment**

To install this software on Windows, you may need administrator privileges. You should exit all programs before installing software.

### **2.1 Download and Install Eclipse**

- a. <http://www.eclipse.org/>
- b. Download and install Eclipse Classic 3.5.1
  - i. Windows: install to the directory C:\Program Files\
  - ii. Linux/Mac: install to local directory

### **2.2 Download and install NSIS Installer (only for windows)**

- a. <http://nsis.sf.net>
- b. Download NSIS installer and install
  - iii. install to the directory C:\Program Files\

### **2.3 Install Subclipse update 1.2x in Eclipse 3.5**

- a. <http://subclipse.tigris.org/servlets/ProjectProcess?pageID=p4wYuA>
- b. Follow a link for zipped downloads:  
<http://subclipse.tigris.org/servlets/ProjectDocumentList?folderID=2240>
- c. Download the site-1.2.4.zip, Zipped update Site for Subclipse 1.2.4 (the website states that this is obsolete, but if you use a more recent version, it will not provide the Import>Other>Checkout Projects from SVN functionality that is needed.)
- d. Extract this zip file and save into the Eclipse Directory.
  - a. Windows: You may need to authorize the overwriting of files.
  - b. Linux/Mac: copy files in Plugin directory to Plugin directory under Eclipse.

### **2.4 Install Java Development Kit**

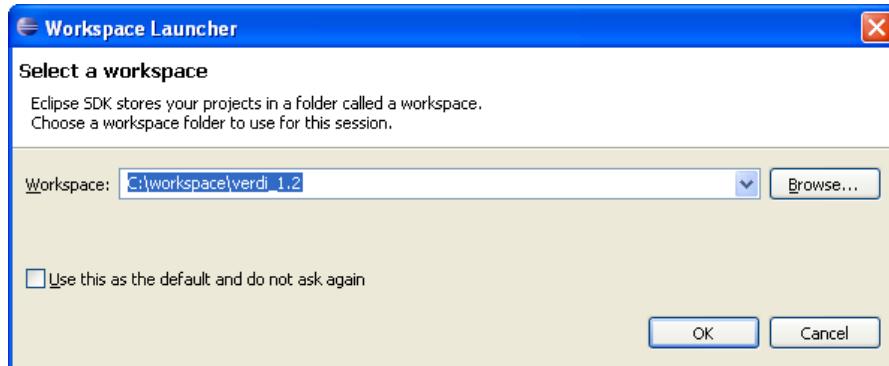
- a. <http://java.sun.com/javase/downloads/index.jsp#jdk>
- b. Install the latest release of JDK that is available from the above link.
- c. For Windows:
  - i. Add the Java compiler to your path on the Windows Machine.
  - ii. From the [desktop](#), right-click **My Computer** and click **Properties**.

- iii. In the System Properties [window](#), click on the **Advanced tab**.
  - iv. In the Advanced section, click the **Environment Variables button**.
  - v. Finally, in the Environment Variables window (as shown below), highlight the **Path** variable in the Systems Variable section and click the **Edit** button.
  - vi. Add the location where the most recent JDK available on your computer. For example: C:\Program Files\Java\jdk1.6.0\_17\bin
- d. For Linux
- a. Add the compiler to a local directory
- e. For Mac
- i. JDK is already installed on OSX
    - a. Check version using command: `javac -version`
    - b. Check path using command: `which javac`
  - ii. Install Java3D to support the contour plot on Mac for VERDI
    - a. <http://www.downloadjava3d.com/mac.php>

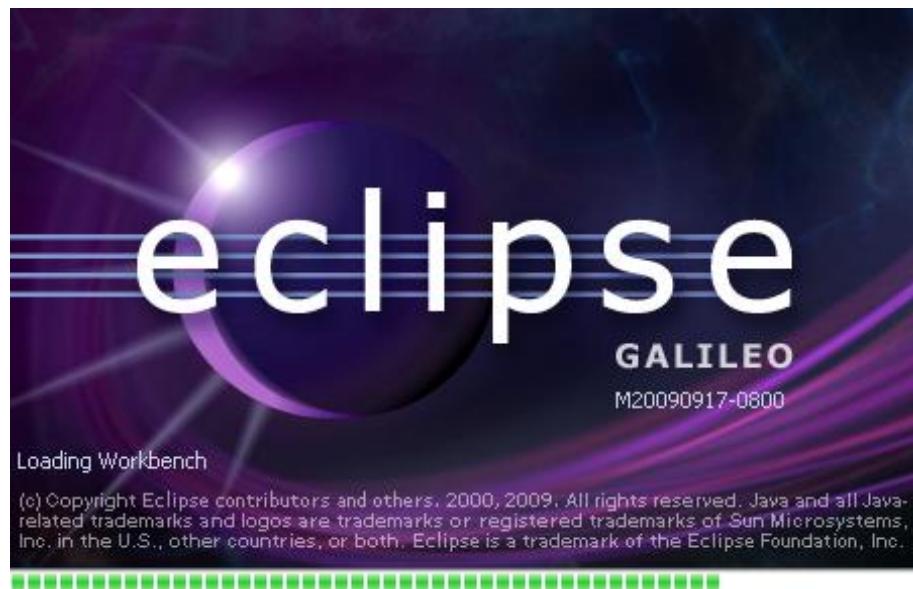
### 3 Running Eclipse

Start Eclipse by going to the C:\Program Files\eclipse directory and double clicking on eclipse.exe. Specify a location for a new workspace folder, for example, C:\workspace\verdi\_1.2 as is shown in Figure 3-1. Eclipse will create the directory automatically.

**Figure 3-1 Select a Workspace**

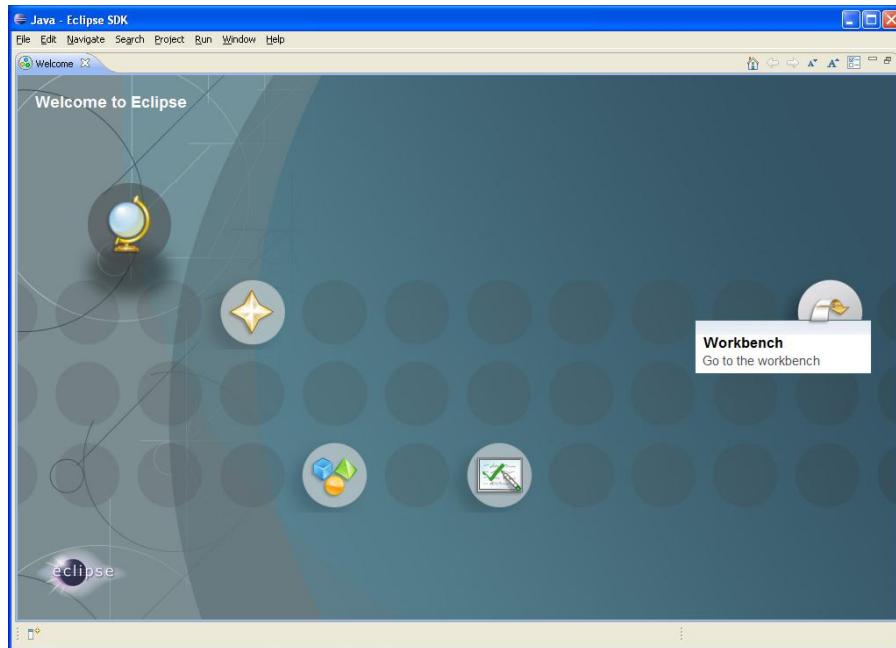


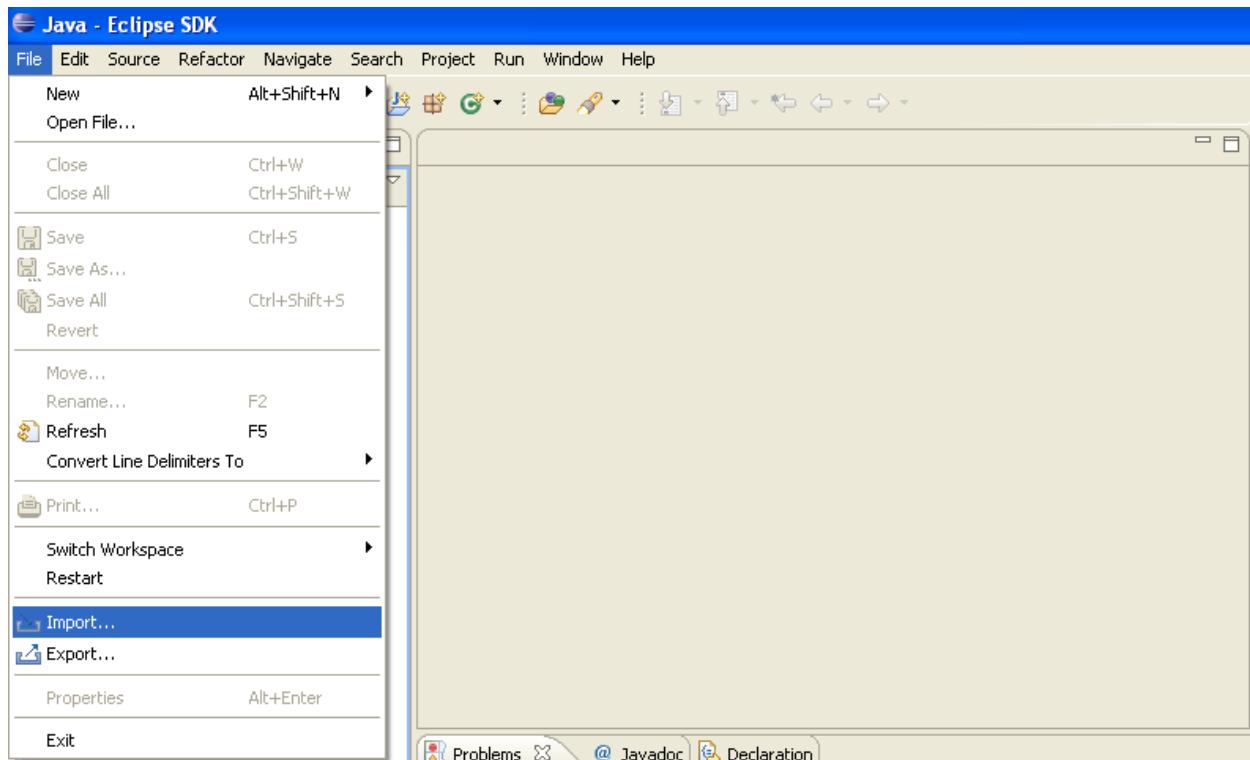
**Figure 3-2 Eclipse Starting Up**



To enter the developer workspace, click on the arrow at the right hand side of the Welcome screen (Figure 3-3). The Eclipse workbench contains several windows to allow the user to view source code, edit, and build within a single developer environment (Figure 3-4).

**Figure 3-3 Eclipse Workbench**



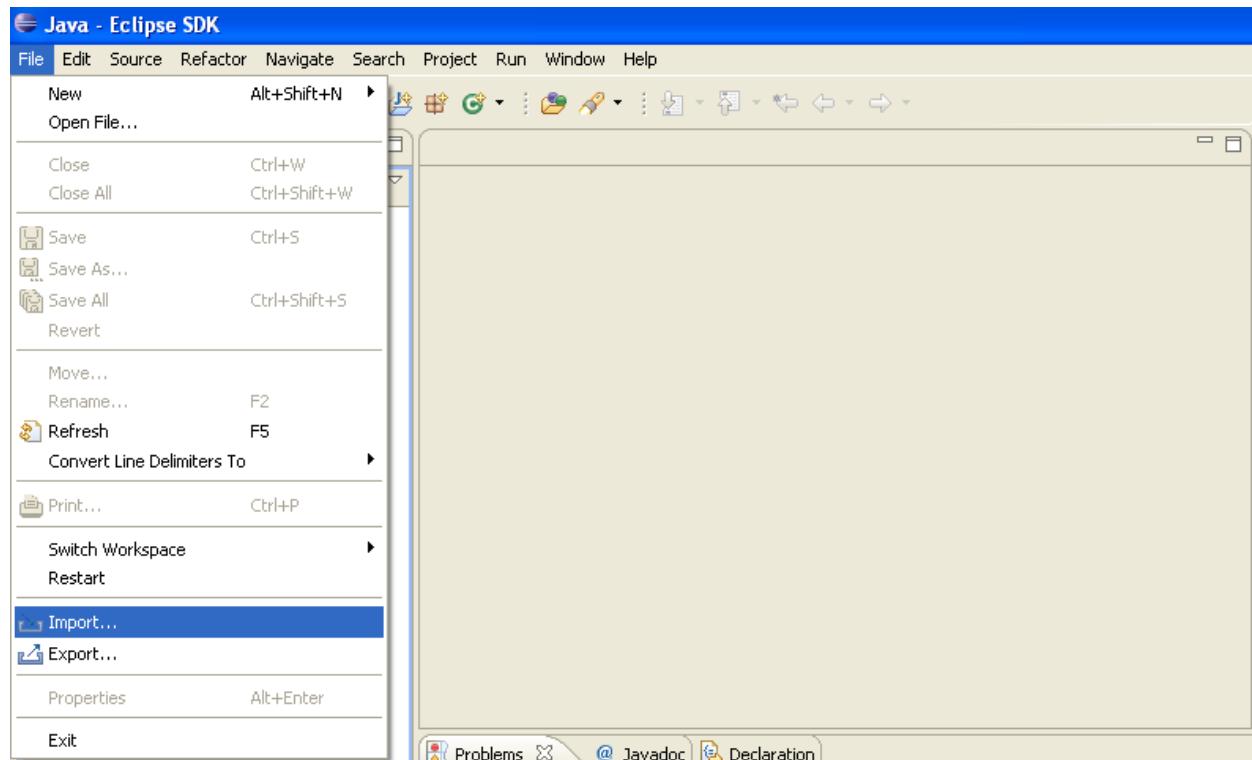
**Figure 3-4 File Import**

## 3.1 Import VERDI Source Code

### 3.1.1 Select File→Import

To import the VERDI source code use the mouse to select **File**→**Import** (Figure 3-5). This will generate a pop-up window titled Import. Expand the **Other** Folder by clicking on the plus sign next to the folder titled **Other** (Figure 3-6). On a Mac, click on and select the SVN wizard (Figure 3-7). **Checkout Projects from SVN** by clicking on it, and then click next.

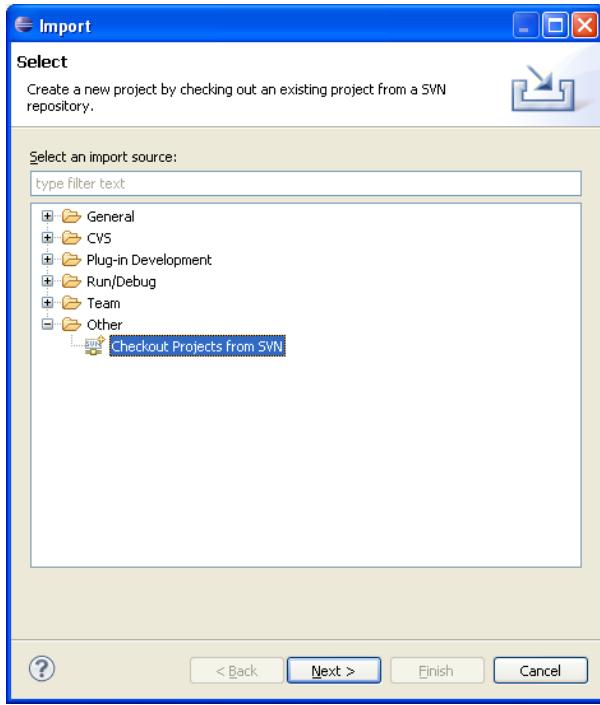
**Figure 3-5 Click on Arrow to see workbench**



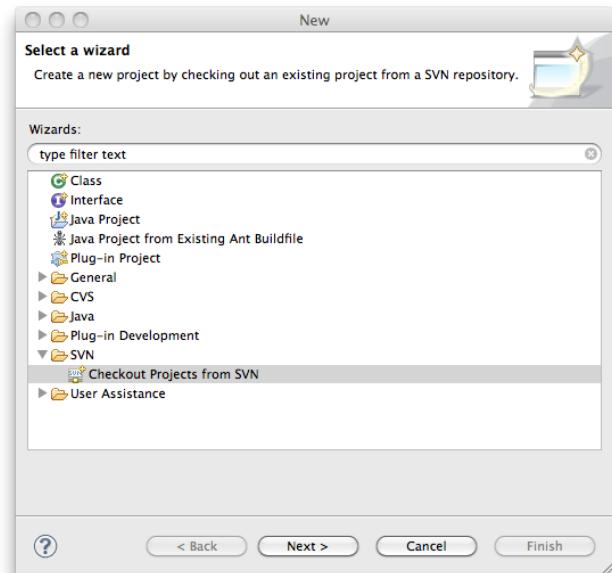
### 3.1.2 Checkout Projects from SVN

Select Checkout Projects from SVN and then click next (Figure 3.6 & Figure 3.7).

**Figure 3-6 Checkout Projects from SVN on Windows/Linux**



**Figure 3-7 Checkout Projects from SVN on Mac**



### 3.1.3 Create a new repository location

Use the mouse to highlight the button next to **Create a new repository location** (Figure 3-6), then click **next**.

**Figure 3-8 Create new repository location**



### 3.1.4 Specify location of VERDI SourceForge repository

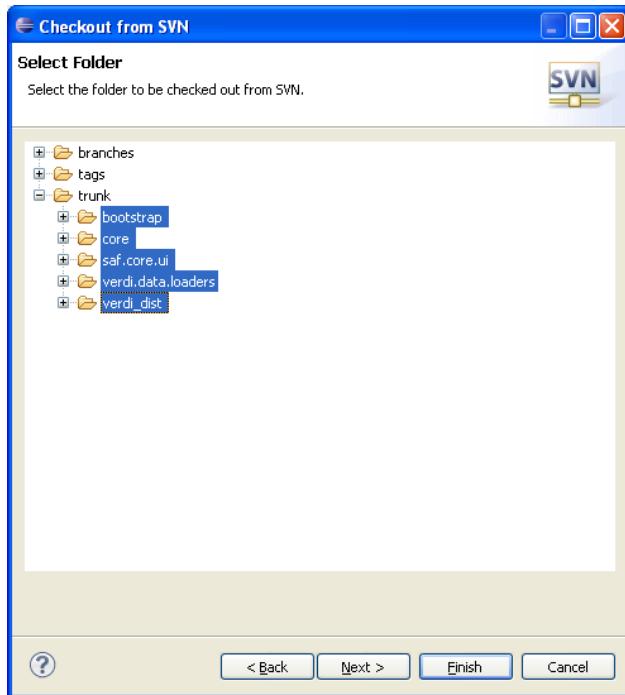
Copy and paste the url: <https://verdi.svn.sf.net/svnroot/verdi> then click next

**Figure 3-9 Checkout Code**

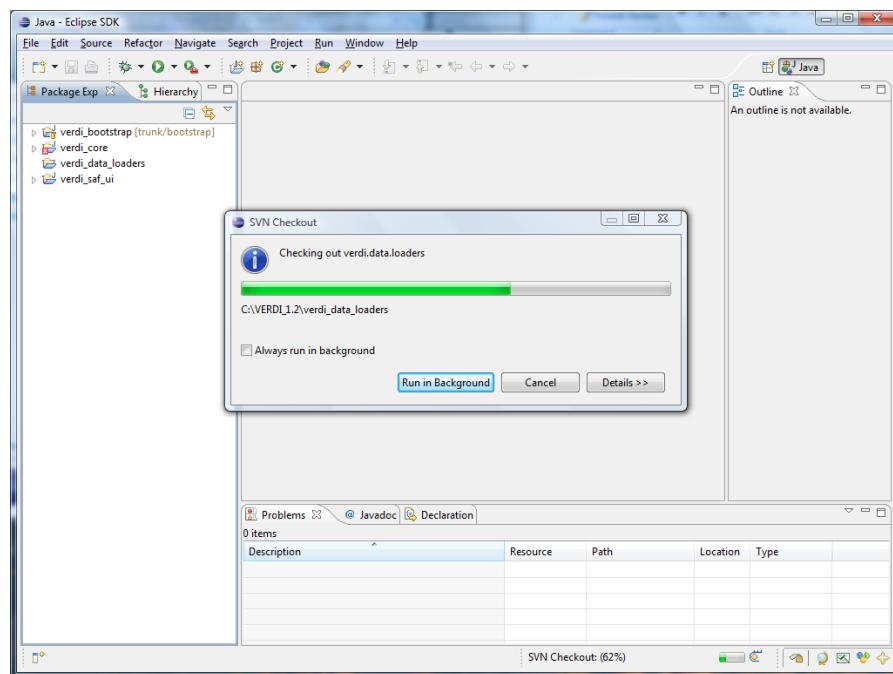
### 3.1.5 Select Folders for Checkout

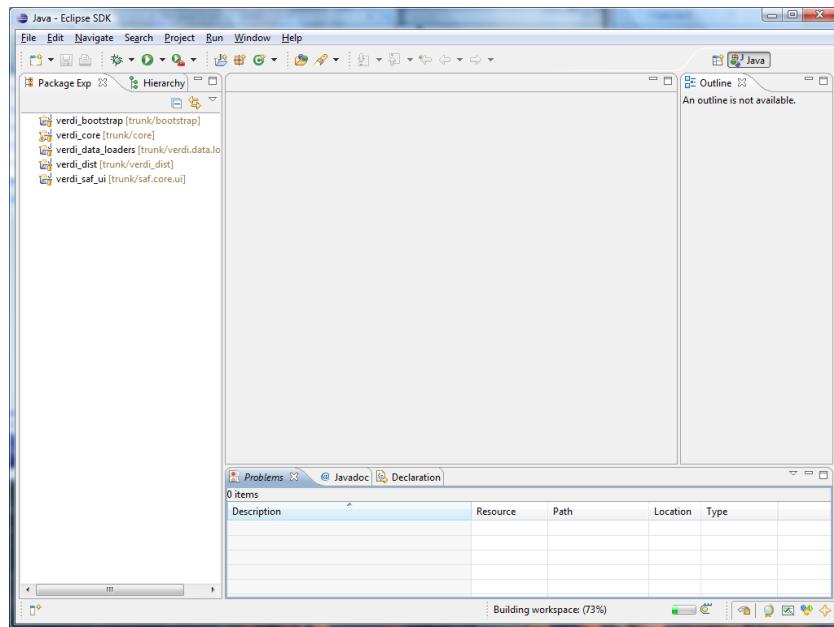
Click on Trunk, use the mouse and the shift key to highlight the following group of subfolders: bootstrap, core, saf.core.ui, verdi.data.loaders, and verdi\_dist, then click finish (Figure 3-9). Eclipse will checkout the latest version of VERDI from the repository. The SVN checkout routine provides the option to run in the background, and also provides a meter indicating the percentage of completion during the checkout process (Figure 3-10). A message will be displayed in a console window at the bottom of the workspace if there is an error. The console will display an error if you use a directory that already exists, and that may have permission problems in copying files to the workspace directory. The workspace and the directory where the VERDI software has been installed should not share the same location. Figure 3-11 shows the code has been successfully imported into the workspace.

**Figure 3-10 Select Folders for Checkout**



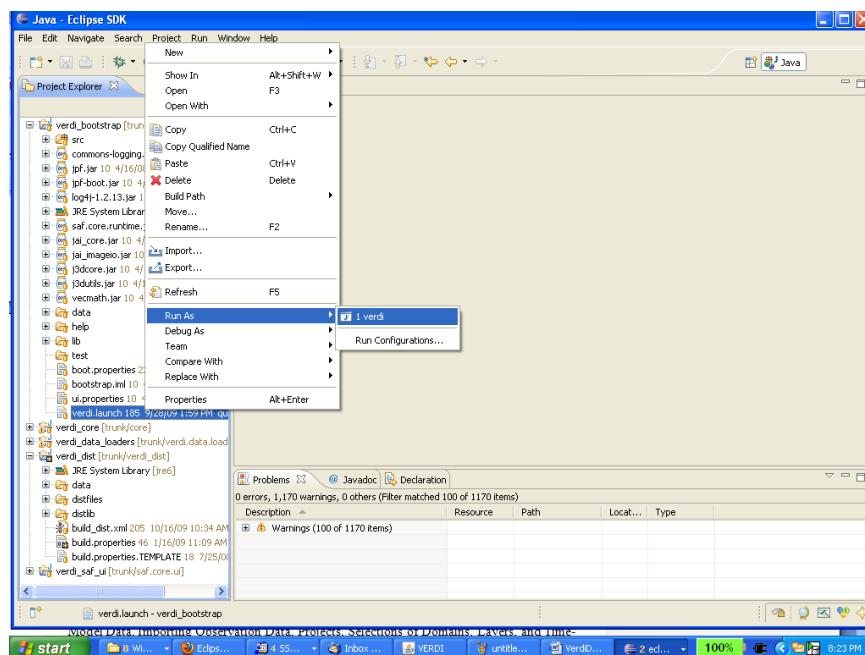
**Figure 3-11 SVN Checkout includes a meter indicating % completion**



**Figure 3-12 SVN imported code into workspace**

## 4 Run VERDI within Eclipse

You can run Verdi using the Verdi.launch script in Verdi\_bootstrap. Right click on Verdi\_launch, select Run As → Verdi (Figure 4-1)

**Figure 4-1 Run VERDI within Eclipse**

## 5 Prepare to build Distribution

Once VERDI has been checked out of the repository, the folders will be displayed in the Project Explorer Window on the Workbench.

### 5.1.1 build.properties file

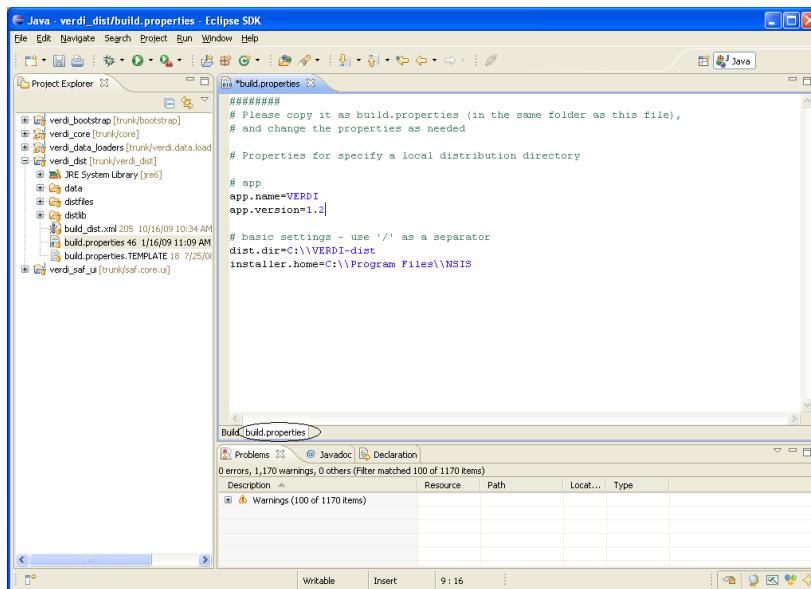
#### 5.1.1.1 Microsoft Windows

If you are building for the Windows platform, open and edit the build.properties file by double clicking on build.properties and click on the tab labeled build.properties to see it in the text editor (Figure 5-1). Edit the build.properties file to specify the directory where Eclipse will build the VERDI distribution and to specify the directory where the NSIS installer software is on your local computer.

#### 5.1.1.2 Linux

Copy the build.properties.TEMPLATE to build.properties. Edit the build.properties file to specify the local directory where Eclipse will build the distribution. Right click on the build.properties file and select Save.

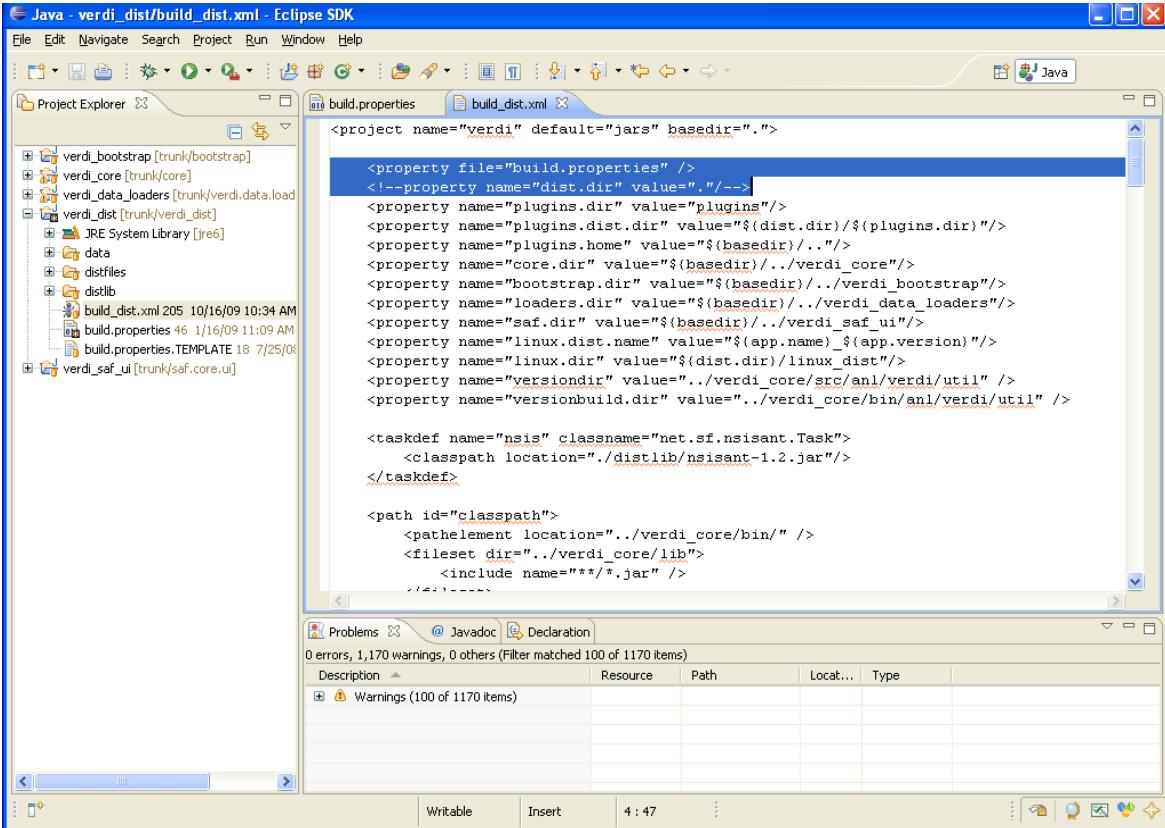
**Figure 5-1 Review/Edit build.properties**



### 5.1.2 Build\_dist.xml

The build\_dist.xml provides the instructions for how to build both the Linux and Windows® distributions of VERDI. The build\_dist.xml file obtains the local directory settings from the build.properties file. It is also possible to make the changes to specify these directories in the build.xml, but the build.properties file has been created to clearly identify what settings are dependent on the local directory configurations, and hopefully will reduce errors that might be incurred by a user editing the build\_dist.xml file. The section of the build\_dist.xml that can be edited to specify the local directories in this section, by commenting out the build.properties file is shown in Figure 5-2. The section of the build\_dist.xml that contains instructions to build the linux distribution is highlighted in Figure 5-3. Figure 5-4 shows the section of the build\_dist.xml file that provides the paths for the installer used to build the Windows® distribution.

**Figure 5-2 build\_dist.xml – no editing needed**



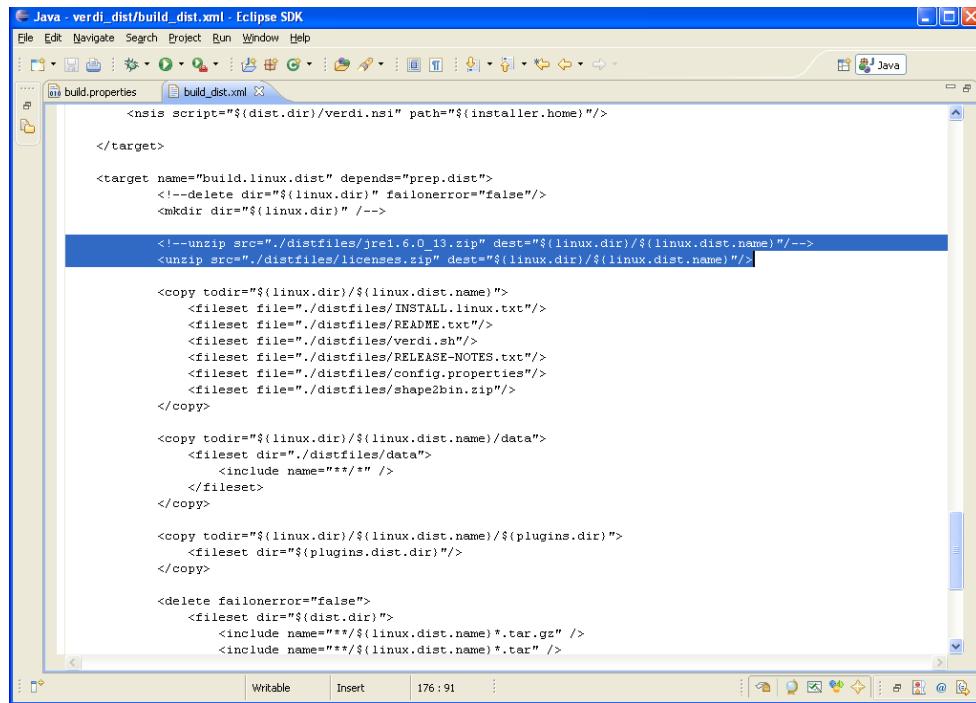
```

<project name="verdi" default="jars" basedir=".">
    <!--property name="dist.dir" value=".//-->
    <property file="build.properties" />
    <!--property name="plugins.dir" value="plugins"/>
    <property name="plugins.dist.dir" value="${dist.dir}/$(plugins.dir)"/>
    <property name="plugins.home" value="${basedir}.."/>
    <property name="core.dir" value="${basedir}../verdi_core"/>
    <property name="bootstrap.dir" value="${basedir}../verdi_bootstrap"/>
    <property name="loaders.dir" value="${basedir}../verdi_data_loaders"/>
    <property name="saf.dir" value="${basedir}../verdi_saf_ui"/>
    <property name="linux.dist.name" value="${app.name}_${app.version}"/>
    <property name="linux.dir" value="${dist.dir}/linux_dist"/>
    <property name="versiondir" value="../verdi_core/src/anl/verdi/util" />
    <property name="versionbuild.dir" value="../verdi_core/bin/anl/verdi/util" />

    <taskdef name="nsis" classname="net.sf.nsisant.Task">
        <classpath location=".//distlib/nsisant-1.2.jar"/>
    </taskdef>

    <path id="classpath">
        <pathelement location=".//verdi_core/bin/" />
        <fileset dir=".//verdi_core/lib">
            <include name="**/*.jar" />
        </fileset>
    </path>

```

**Figure 5-3 build\_dist.xml: build Linux distribution section**


The screenshot shows the Eclipse IDE interface with the title bar "Java - verdi\_dist/build\_dist.xml - Eclipse SDK". The main window displays the XML code for the build process. The code defines a target named "build.linux.dist" which depends on "prep.dist". It includes tasks for unzipping files, creating directories, copying files from "distfiles" to the distribution directory, and finally creating NSIS scripts for the installer.

```

<nsis script="${dist.dir}/verdi.nsi" path="${installer.home}" />
</target>

<target name="build.linux.dist" depends="prep.dist">
    <!--delete dir="${linux.dir}" failonerror="false"/>
    <mkdir dir="${linux.dir}" /-->
    <!--unzip src="./distfiles/jre1.6.0_13.zip" dest="${linux.dir}/${linux.dist.name}" /-->
    <unzip src="./distfiles/licenses.zip" dest="${linux.dir}/${linux.dist.name}" />

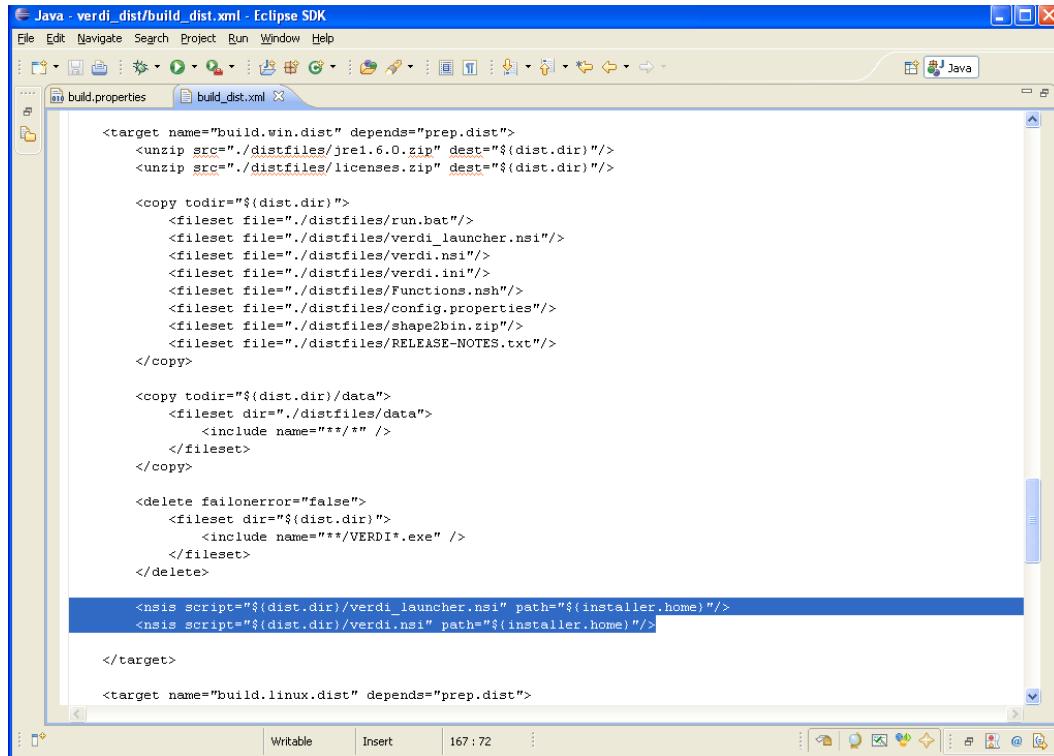
    <copy todir="${linux.dir}/${linux.dist.name}">
        <fileset file="./distfiles/INSTALL.linux.txt"/>
        <fileset file="./distfiles/README.txt"/>
        <fileset file="./distfiles/verdi.sh"/>
        <fileset file="./distfiles/RELEASE-NOTES.txt"/>
        <fileset file="./distfiles/config.properties"/>
        <fileset file="./distfiles/shape2bin.zip"/>
    </copy>

    <copy todir="${linux.dir}/${linux.dist.name}/data">
        <fileset dir="./distfiles/data">
            <include name="**/**" />
        </fileset>
    </copy>

    <copy todir="${linux.dir}/${linux.dist.name}/${plugins.dist.dir}">
        <fileset dir="${plugins.dist.dir}" />
    </copy>

    <delete failonerror="false">
        <fileset dir="${dist.dir}">
            <include name="**/${linux.dist.name}.tar.gz" />
            <include name="**/${linux.dist.name}.tar" />
        </fileset>
    </delete>

```

**Figure 5-4 dist.xml: nsis installer path section**


The screenshot shows the Eclipse IDE interface with the title bar "Java - verdi\_dist/build\_dist.xml - Eclipse SDK". The main window displays the XML code for the nsis installer path section. It includes targets for building Windows and Linux distributions, with specific sections for unzipping files, copying files, and defining NSIS scripts for the installer.

```

<target name="build.win.dist" depends="prep.dist">
    <unzip src="./distfiles/jre1.6.0.zip" dest="${dist.dir}" />
    <unzip src="./distfiles/licenses.zip" dest="${dist.dir}" />

    <copy todir="${dist.dir}">
        <fileset file="./distfiles/run.bat"/>
        <fileset file="./distfiles/verdi_launcher.nsi"/>
        <fileset file="./distfiles/verdi.nsi"/>
        <fileset file="./distfiles/verdi.ini"/>
        <fileset file="./distfiles/Functions.nsh"/>
        <fileset file="./distfiles/config.properties"/>
        <fileset file="./distfiles/shape2bin.zip"/>
        <fileset file="./distfiles/RELEASE-NOTES.txt"/>
    </copy>

    <copy todir="${dist.dir}/data">
        <fileset dir="./distfiles/data">
            <include name="**/**" />
        </fileset>
    </copy>

    <delete failonerror="false">
        <fileset dir="${dist.dir}">
            <include name="**/VERDIT.exe" />
        </fileset>
    </delete>

    <nsis script="${dist.dir}/verdi_launcher.nsi" path="${installer.home}" />
    <nsis script="${dist.dir}/verdi.nsi" path="${installer.home}" />

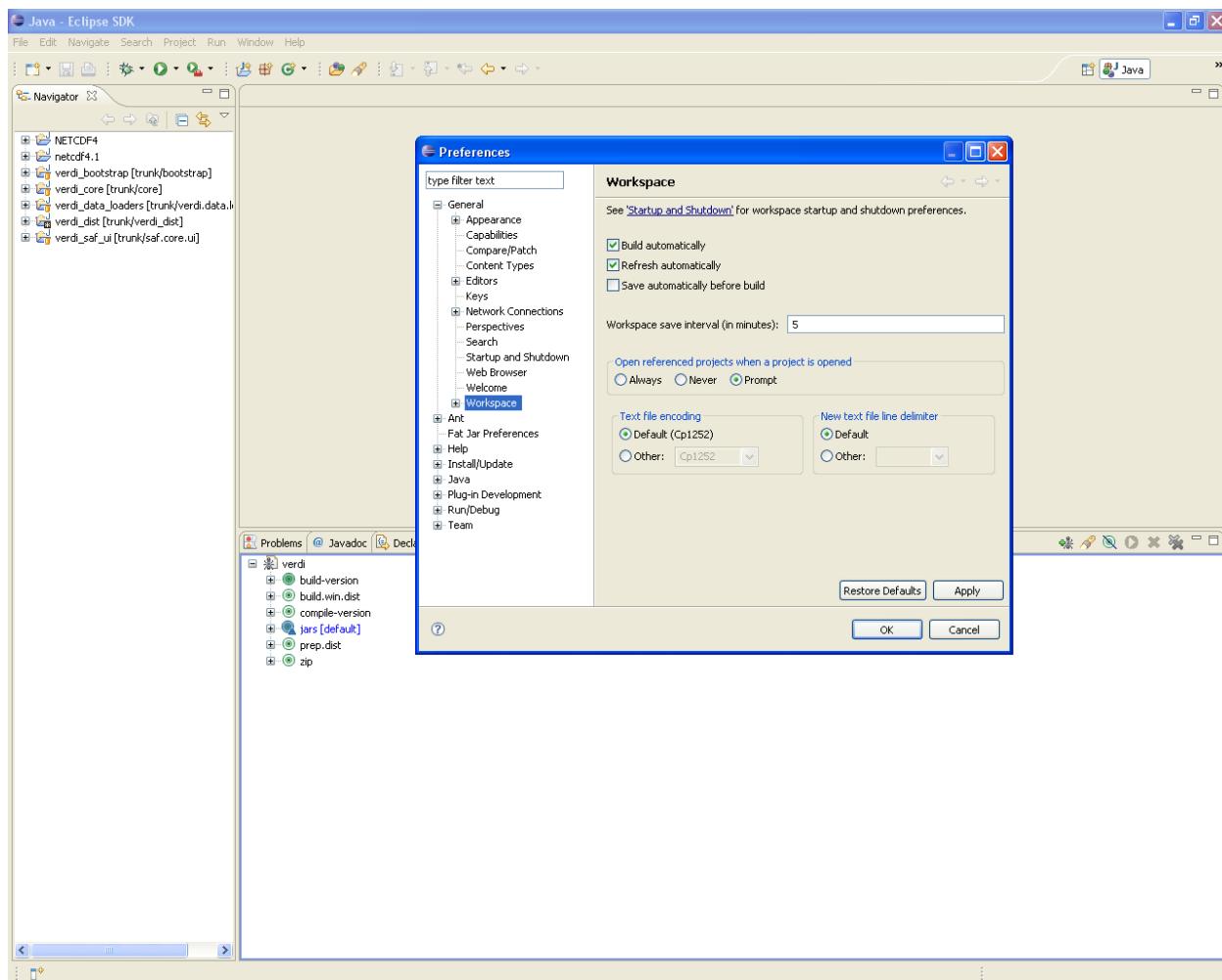
```

## 6 Build VERDI

### 6.1 Build using Ant

Before building the releases, a few settings have to be done through Eclipse menu options Window→Preferences.... Once the Preferences dialog is open (Figure 6.1), select ‘General’ then ‘Workspace’ item. In the ‘Workspace’ panel, select the check boxes for ‘Build automatically’ and ‘Refresh automatically’. Click ‘OK’ button to close the preferences dialog.

**Figure 6-1 Window→Preferences**



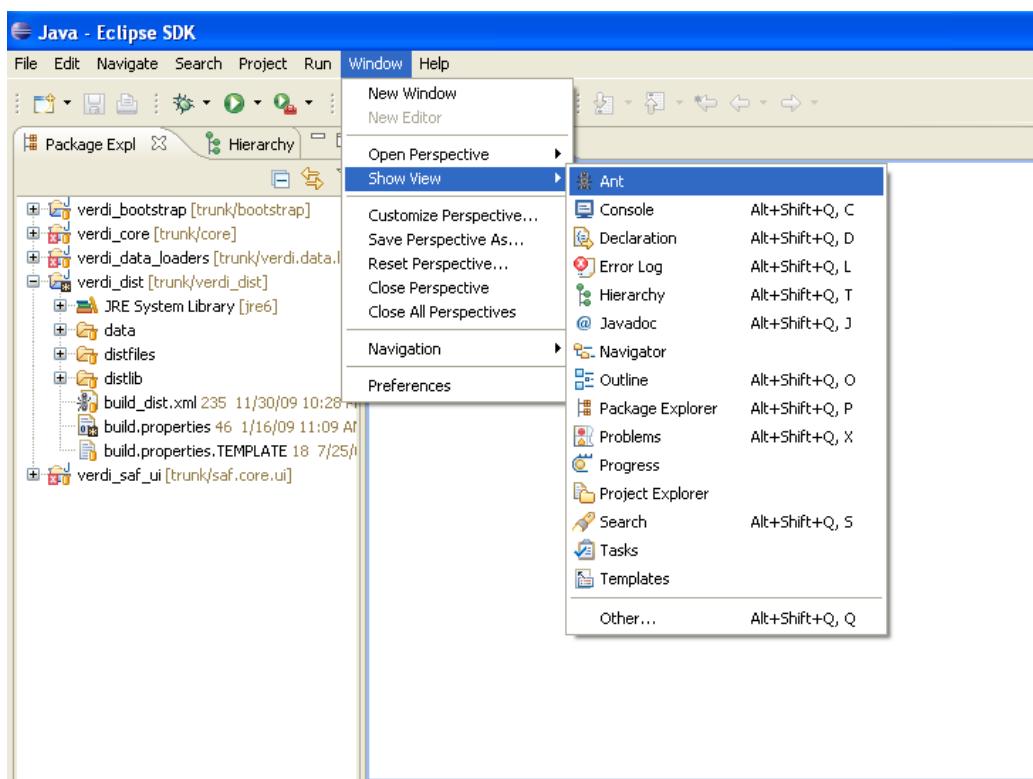
#### 6.1.1 Microsoft Windows

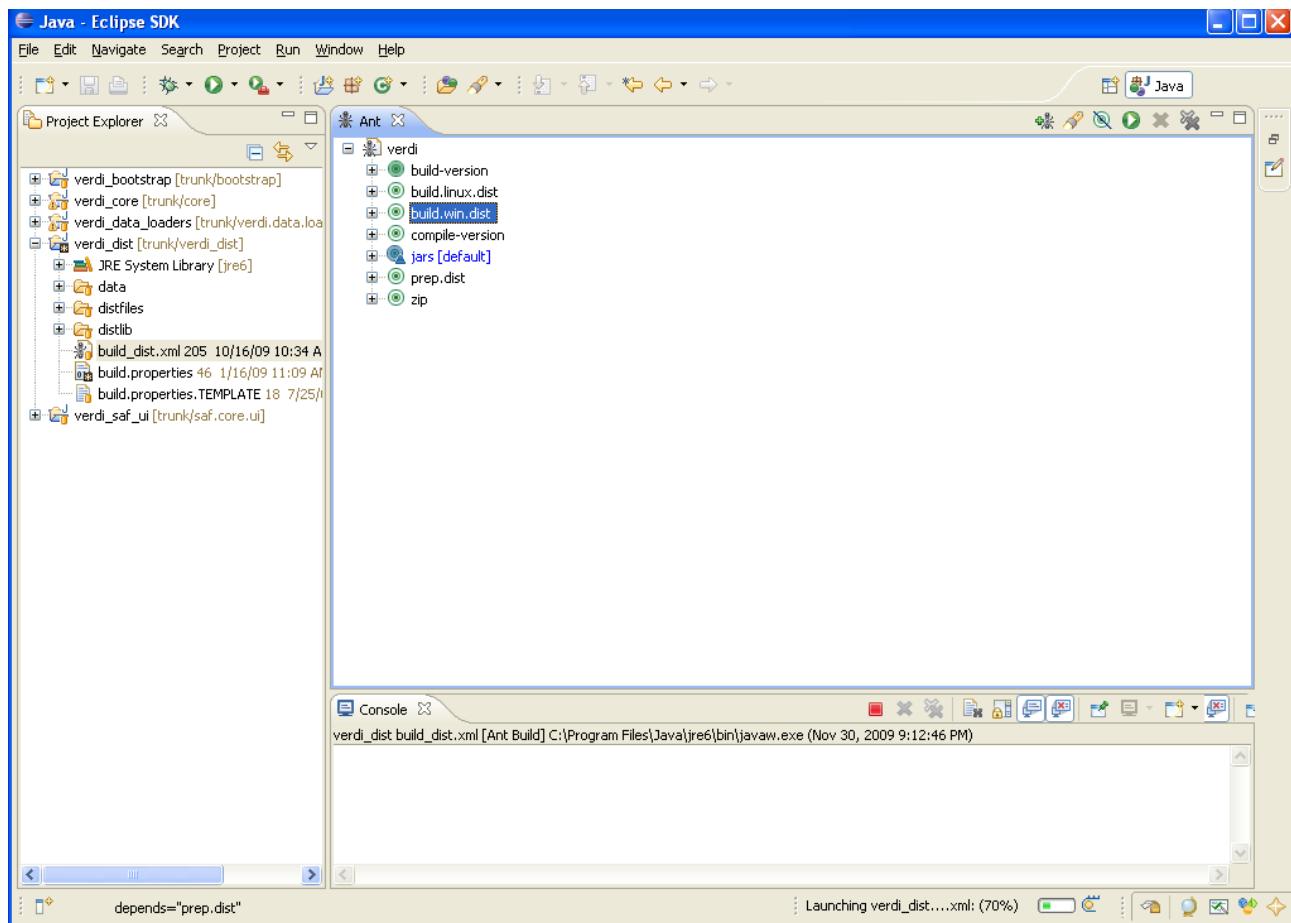
The Windows® distribution can be built using the scripts in the distlib folder within trunk/verdi\_dist on a Windows machine. Select the eclipse menu options Window→Show

View→Ant to create a subwindow for Ant (Figure 6-2). Then drag the build\_dist.xml into the Ant window. Click on the plus button next to verdi to open and display the contents.

1. Double click on build-version to label the build with the current version number
2. Double click on compile-version to compile all the source code of current version
3. Double click on build.win.dist to build the VERDI distribution for a Windows machine. (Figure 6-3)

**Figure 6-2 Window→Show View→Ant**



**Figure 6-3 Double click on build.win.dist to build VERDI distribution**

### 6.1.1 Linux Distribution

Linux distribution can be built by using Ant to run build.linux.dist on a linux machine. The trunk/verdi\_dist folder contains the build\_dist.xml script. In the trunk/verdi\_dist/distfiles folder, there should contain a fully unfolded jre1.6.3\_13 package. Select the eclipse menu options Window→Show View→Ant to create a subwindow for Ant (Figure 6-2). Then drag the build\_dist.xml into the Ant window. Click on the plus button next to verdi to open and display the contents.

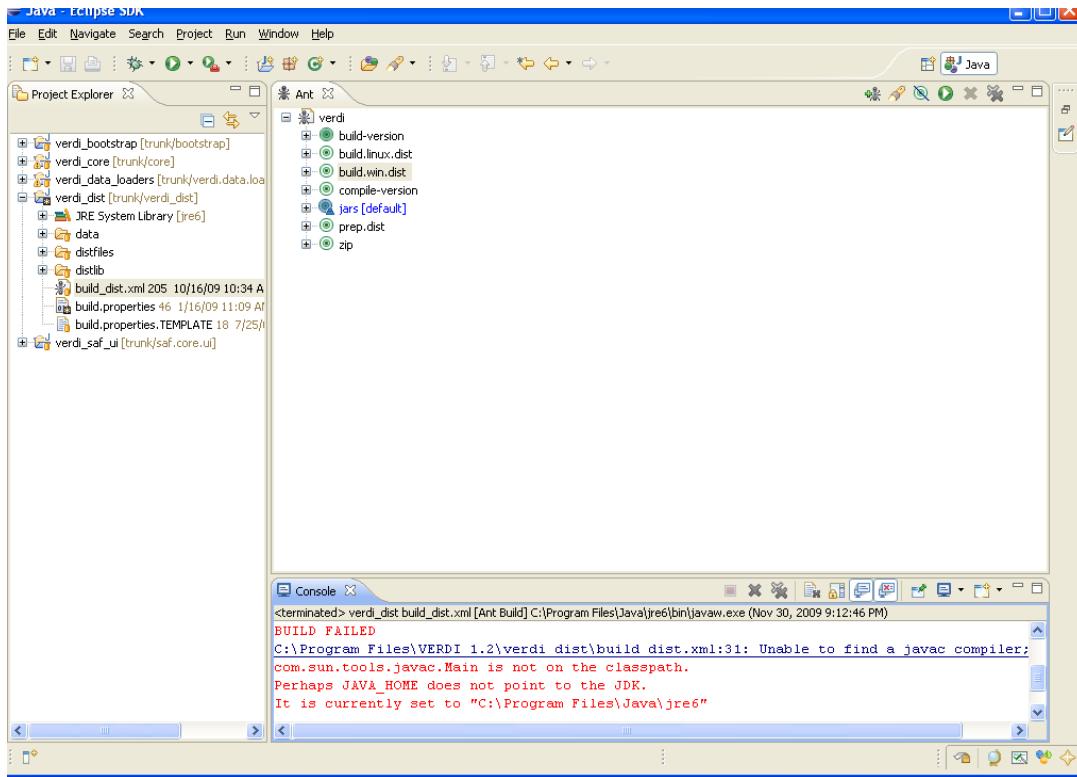
1. Double click on build-version to label the build with the current version number
2. Double click on compile-version to compile all the source code of current version

3. Double click on build.linux.dist to build the VERDI distribution for a linux machine.

## 6.2 Check Console for Error Messages

Error messages will appear in the console underneath the Ant console.

Figure 6-4 Console Error Message



If you obtain this error, add the Java compiler to your path on the Windows Machine (see Section 2.2 and Section 6.2)

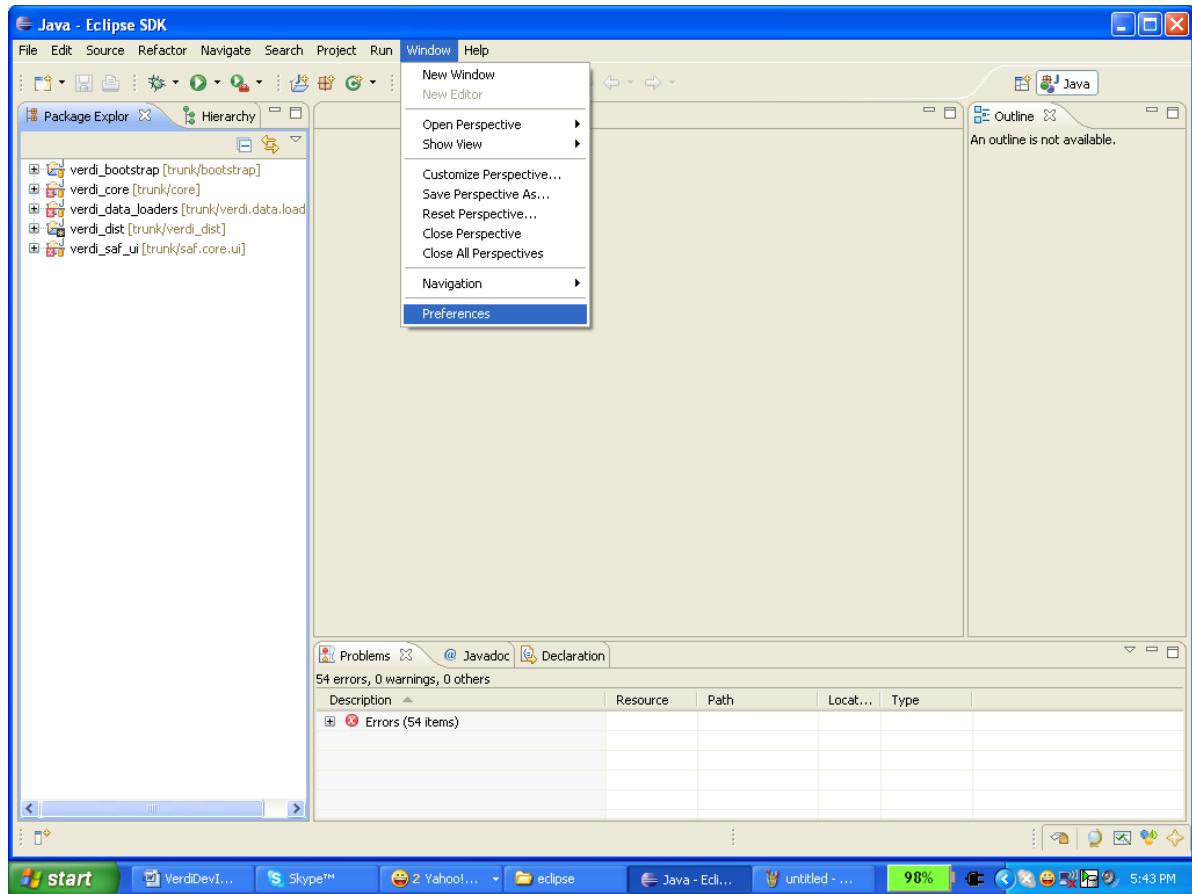
## 6.3 Add Java Compiler to Ant

To allow the Ant compiler to find the compiler, you will also need to change the Ant Preferences to add tools.jar as an external jar as follows.

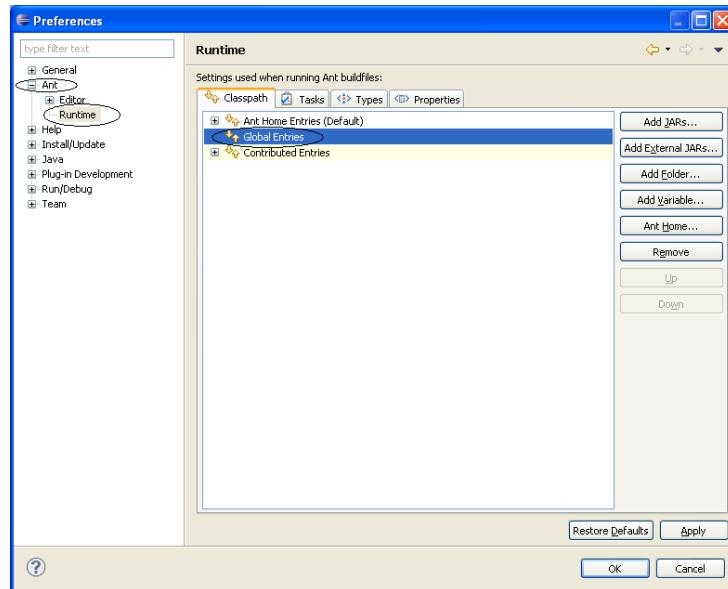
1. In Eclipse main menu select Window→Preferences (Figure 6-5)
2. In the Preference Window select Ant→Runtime (Figure 6-6)

3. Click on the Classpath tab, select Global Entries
4. Click on Add External JARs
5. Locate the tools.tar under the lib folder on the JDK local installation directory then click OK, and click OK again. (Figure 6-7)

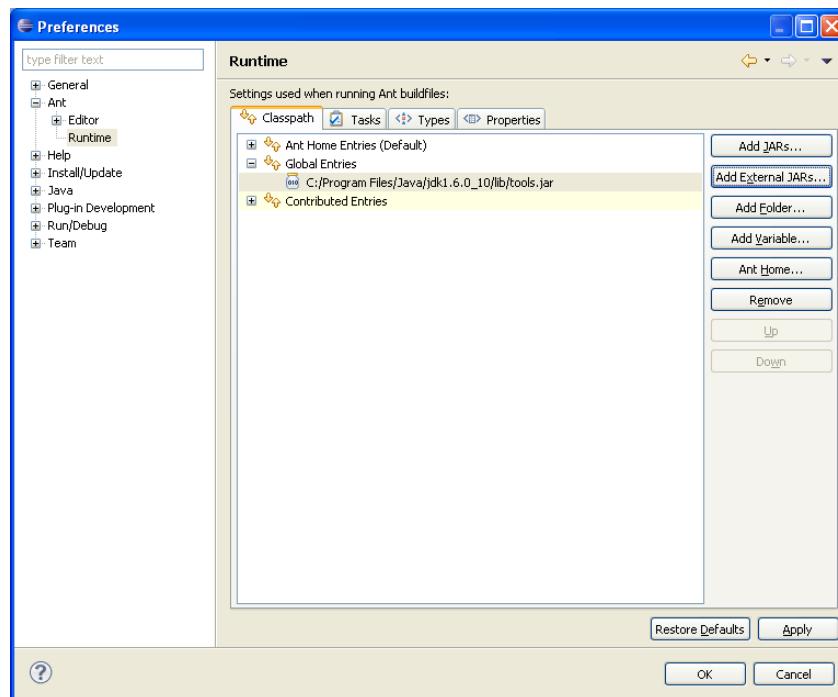
**Figure 6-5 Open Windows→Preferences**



**Figure 6-6 Expand Ant, Select Runtime, Select Global Entries**



**Figure 6-7 Add tools.tar to Ant Preferences**



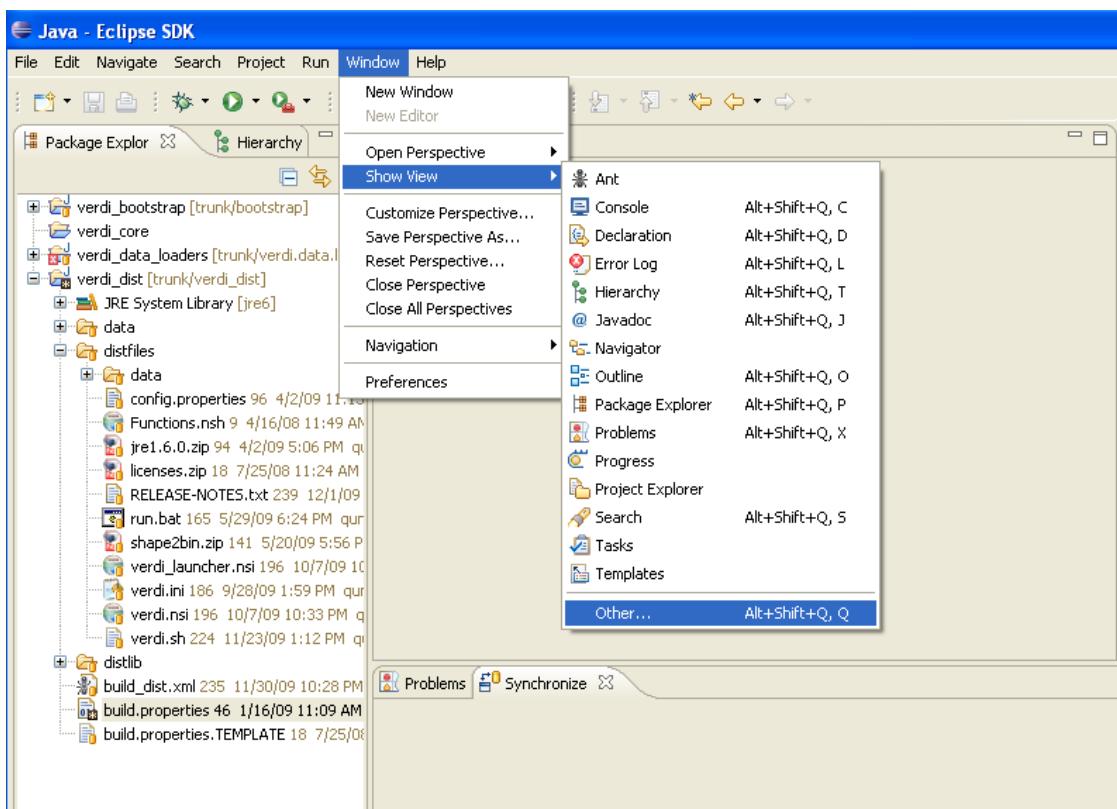
## 7 Updating source code from the repository

If you have previously obtained the code from the repository and want to obtain any bug fixes or new features that developers have committed to the repository, the synchronization window allows you to do this.

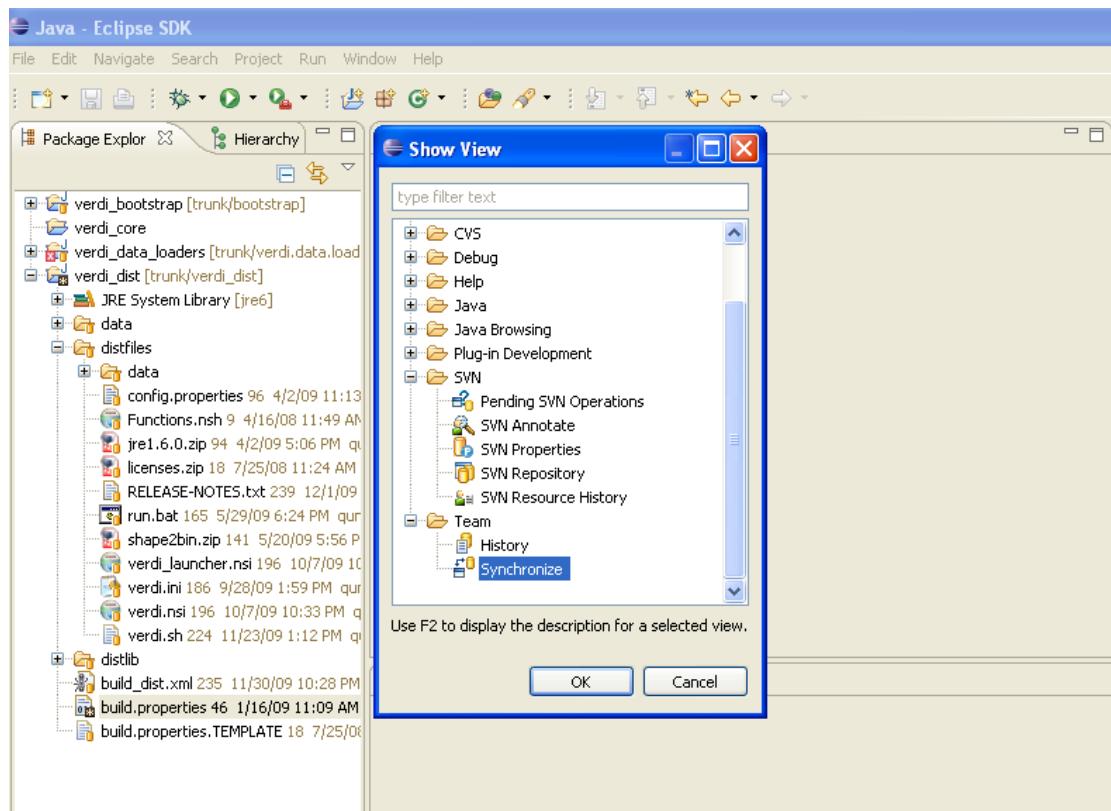
### 7.1 Open the Synchronization Window

Select Window→Show View→Other to open the **Show View** Pop-up Window (Figure 5-1). Expand the Team Folder by clicking on the plus symbol and then highlight the word Synchronize and click OK (Figure 7-2).

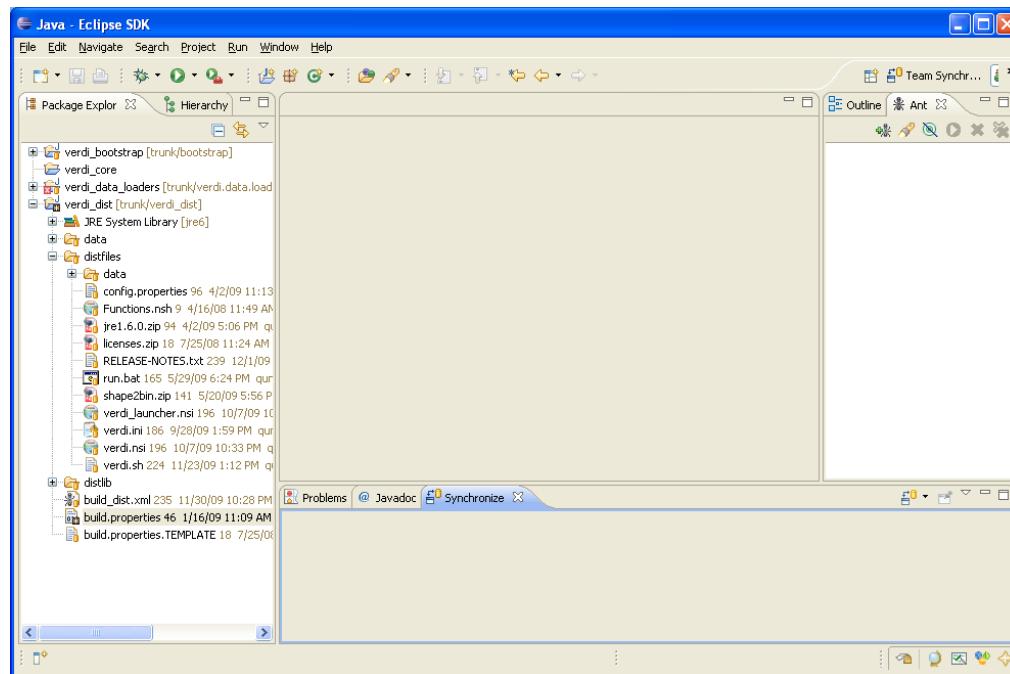
**Figure 7-1 Show View → Other**



**Figure 7-2 Expand Team Folder - highlight Synchronize – click ok**



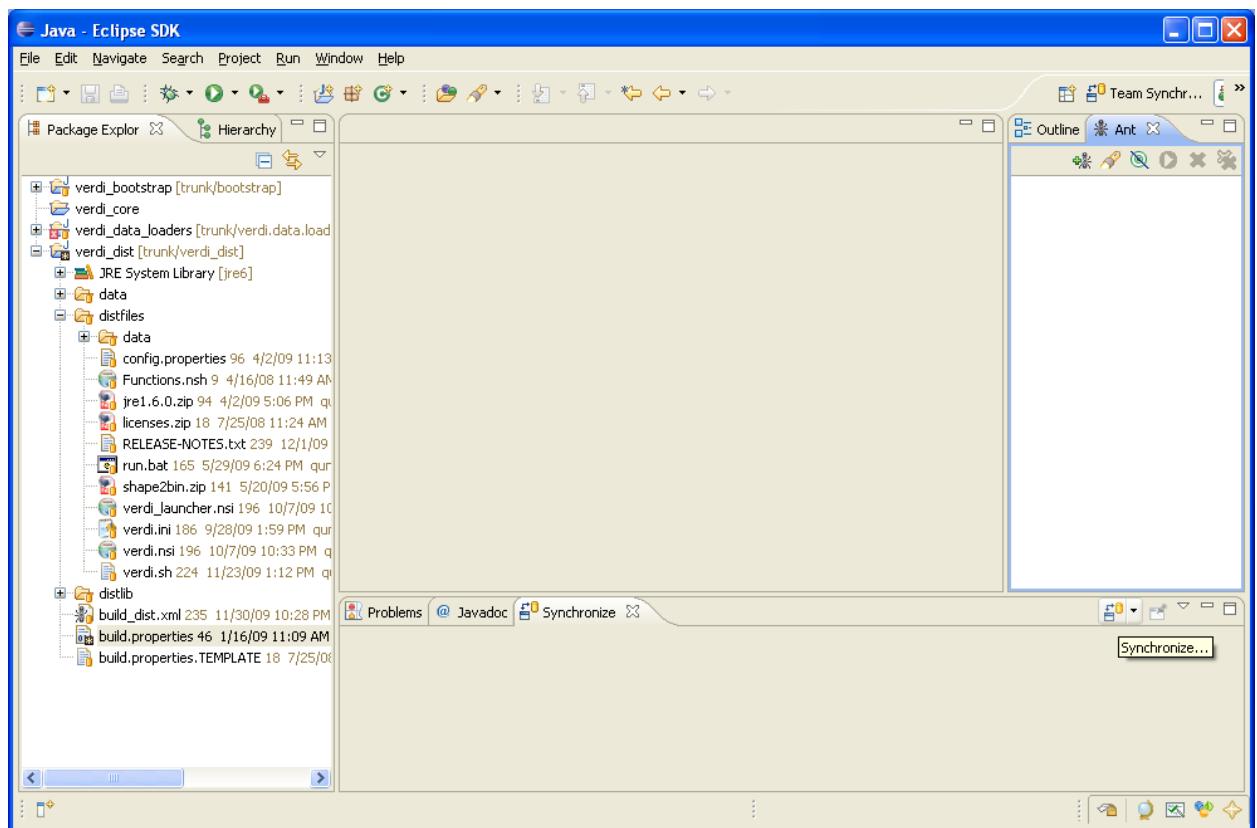
**Figure 7-3 Synchronize Window added to bottom of Workspace**



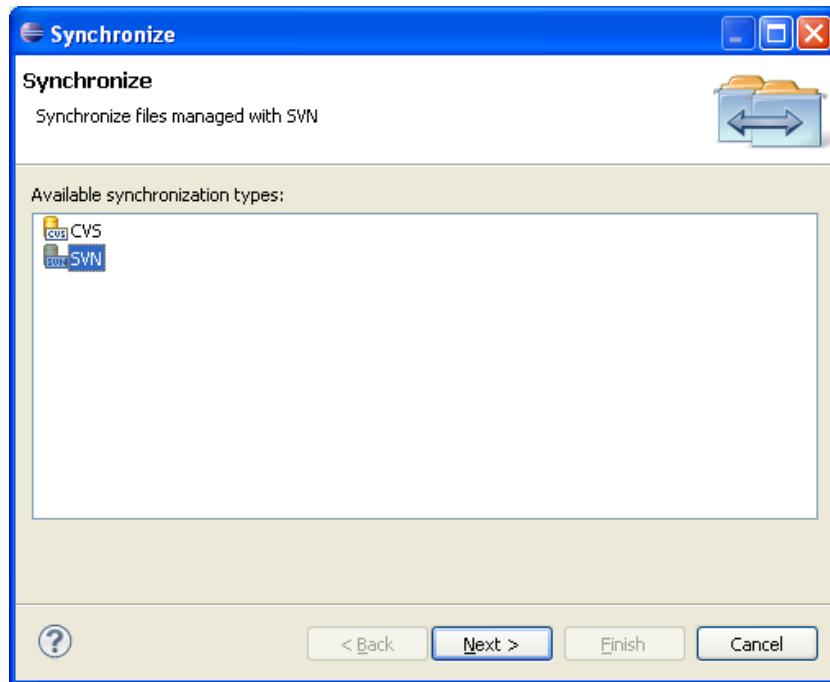
## 7.2 Synchronize with Repository using SVN

The Synchronize window contains a synchronize symbol at the upper right corner. Click on the synchronize symbol to open the Synchronize popup window. Click on SVN to select the subversion software package manager, then click NEXT. A Synchronize SVN pop-up window will appear listing the packages that are available as resources for synchronization. Click the Select All button to select all the packages, then click FINISH. A pop-up window labeled Confirm Open Perspective will ask if you would like to change from the Java Perspective to the Team Synchronization Perspective. If you opt to change perspectives, there is a right arrow button in the upper right hand corner of the workspace to switch back to the Java Perspective once you are finished reviewing the code in the Team Synchronization Perspective.

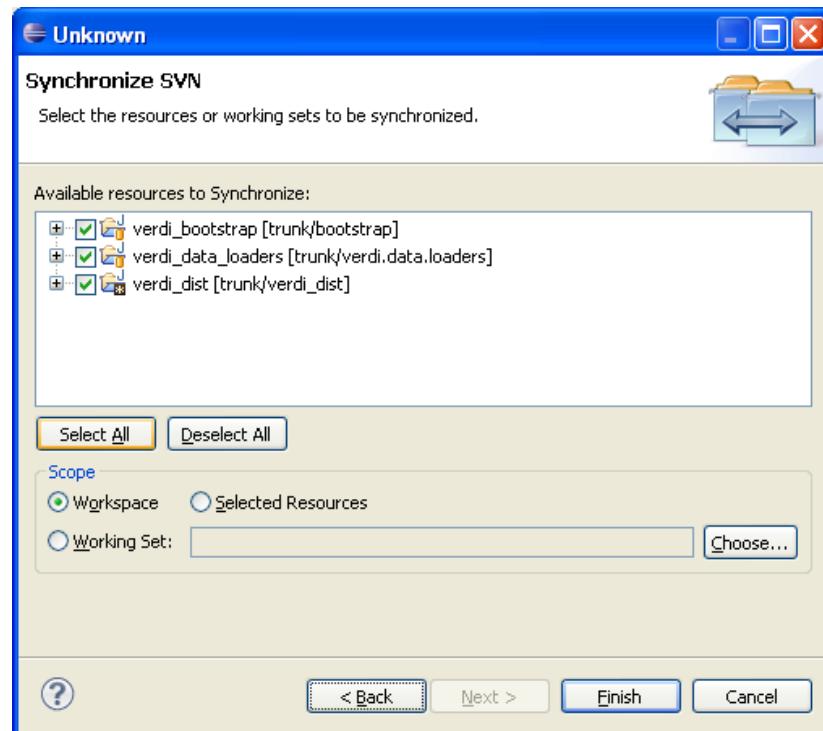
**Figure 7-4 Click on Synchronize Symbol to bring up Pop-up**



**Figure 7-5 Synchronize Pop-up – select SVN**



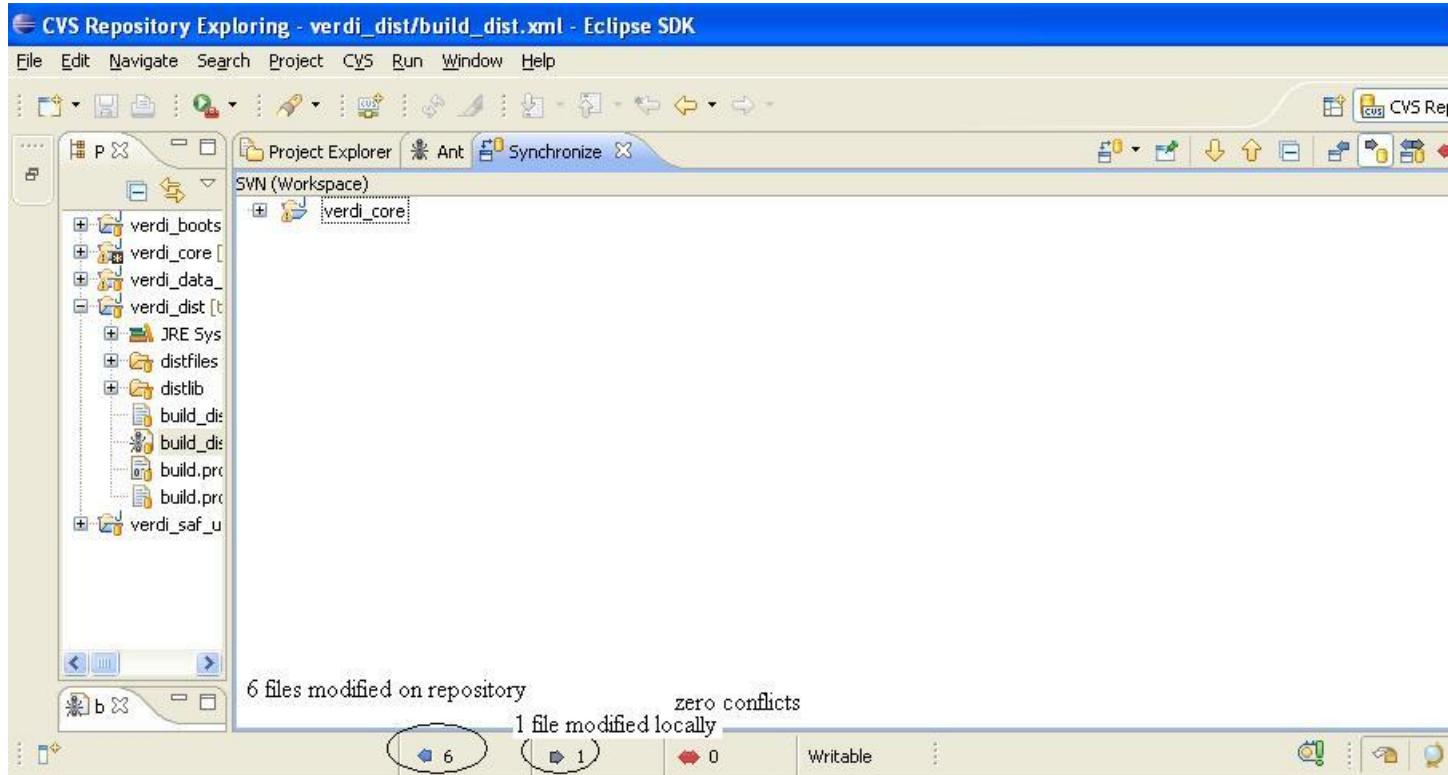
**Figure 7-6 Synchronize SVN – select all**



### 7.3 Resolve conflicts and Update

After synchronization, there will be a blue incoming arrow at the bottom of eclipse window that reports the number of files on the repository that are different than your local workspace, a green outgoing arrow, and red double conflict arrow. If you click on the blue arrow, and then right click on the Verdi\_core, a pop-up menu will allow you to select update, to bring the updates to the code down to your local workspace (see Figure 7-8).

**Figure 7-7 Check for Updates and Conflicts**



**Figure 7-7 Check for Updates and Conflicts**

## VERDI Developer Instructions 1.2

