Overture Maven Eclipse plugin development Guide

Kenneth Lausdahl

April 2009

1 Environment

In order to compile the eclipse plug-ins of the Overture Tool the environment has to be setup. This includes Java, Maven, Eclipse and Eclipse test framework.

1.1 Development tools

The following tools must be installed to enable compilation of the overture tool eclipse plug-ins:

1.1.1 JDK 1.5 or later

Java JDK 1.5 or later must be installed since the overture tool is written in Java.

1.1.2 Maven2

The Maven, Java project management tool and build automation. Note that:

- You MUST use Maven version 2.0.9 or later.
- Set the M2_HOME environment variable to point to the maven root directory.
- Set JAVA_HOME to the folder where JDK is installed.
- Add %M2_HOME%\ bin directory to your $Path.
- Start a terminal and check whether mvn -v works!
- It should reply with Maven version: 2.0.9.

The Maven user setting file is located in directory /.m2/ be default it contains user settings for Maven which apply for all projects. The settings file is named "settings.xml" a description of how to use this file in this project is described in section **??**.

1.1.3 Eclipse Classic 3.4.2

Eclipse Classic version 3.4.2 must be installed it can be fetched form `http://www.eclipse.org/downloads` Additional to eclipse some plug-ins must be installed in Eclipse.

- Dynamic Languages Toolkit
  - `http://download.eclipse.org/technology/dltk/updates-dev/1.0`
  - Core feature is needed
- Maven 2 Eclipse plugin

  - `http://m2eclipse.sonatype.org/update/`

- Eclipse Test and Performance Tools Platform Project (TPTP)
  - `http://www.eclipse.org/tptp/home/downloads/`

- Subclipse
  - `http://subclipse.tigris.org/update_1.4.x/`
  - Select the complete package
  - Guide at twiki `http://www.overturetool.org/twiki/bin/view/Main/InstallSubclipse`

1.2 Required test tools

Maven needs a eclipse installation to run tests within, this cannot be the eclipse used to development. Download eclipse classic 3.4.2 and place it in: Your user home:

1.2.1 Eclipse classic 3.4.2

- Dynamic Languages Toolkit
  - `http://download.eclipse.org/technology/dltk/updates-dev/1.0`
  - Core feature is needed

- Eclipse Test and Performance Tools Platform Project (TPTP)
  - `http://www.eclipse.org/tptp/home/downloads/`
  - Using "Ganymede" update site the "Testing and Performance" can be selected and installed.

Location of Eclipse for testing The Maven plug-in pst plug-in which enables Maven to compile Eclipse plug-ins requires an Eclipse version with TPTP installed. This Eclipse version must be installed into the following location:

- Windows your Eclipse directory would be C:\Documents and Settings\username\eclipse
- Unix/Linux/BSD it would be /home/username/eclipse
- Mac OS X it would be /Users/username/eclipse

1.3 Maven Settings

Maven settings is a XML file named "Settings.xml". It is located in the root of the local Maven repository. Default location is:

```
${user.home}/.m2/settings.xml
```

Setting up the project some things have to be specified:

1. Set the default profile
2. Set VDM tools path (Optional - only needed if using vdmt)
3. Set the eclipse startup jar
4. Add a repository where all eclipse dependencies can be resolved from

If you are not familiar with maven then use the predefined settings.xml files for your environment provided together with this document. In the following sub sections snippets of the required settings are shown. Fore more information about maven 2 settings see: `http://maven.apache.org/ref/2.0.9/maven-settings/settings.html`. A overview of the sttings file can also be seen in section **??** on page **??**.

Setting the default profile When a profile is added like in the example shown in section **??** on page **??**, the default profile can be set using this tag:

```
<activeProfiles>
  <activeProfile>default</activeProfile>
</activeProfiles>
```

Adding properties to the profile Here the path to VDM Tools is set and the eclipse startup jar is specified.

```
<properties>

  <user.vdmtoolscmdpath>
    c:\Program Files\The VDM++ Toolbox v8.2b\bin\vppde.exe
  </user.vdmtoolscmdpath>

  <user.eclipseStartup>
    plugins\org.eclipse.equinox.common_3.4.0.v20080421-2006.jar
  </user.eclipseStartup>

</properties>
```

Adding repositories to the profile In order for Maven to find the eclipse artifacts the repositories containing this artifacts need to be specified.

```
<repositories>
  <repository>
    <id>lausdahl.com</id>
    <name>Lausdahl Snapshots</name>
    <releases>
      <enabled>true</enabled>
      <updatePolicy>daily</updatePolicy>
      <checksumPolicy>ignore</checksumPolicy>
    </releases>
    <snapshots>
      <enabled>true</enabled>
      <updatePolicy>never</updatePolicy>
      <checksumPolicy>ignore</checksumPolicy>
      </snapshots>
      <url>http://maven2.lausdahl.com</url>
      <layout>default</layout>
  </repository>
  <repository>
       <id>sf.net</id>
       <name>Lausdahl Snapshots</name>
       <releases>
         <enabled>true</enabled>
         <updatePolicy>daily</updatePolicy>
         <checksumPolicy>ignore</checksumPolicy>
       </releases>
       <snapshots>
          <enabled>true</enabled>
          <updatePolicy>never</updatePolicy>
          <checksumPolicy>ignore</checksumPolicy>
        </snapshots>
         <url>
http://overturetraces.svn.sourceforge.net/viewvc/overturetraces
        </url>
```

```
        <layout>default</layout>
  </repository>
</repositories>
```

Adding plugin repositories to the profile In order for Maven to find the maven plugins for building eclipse plugin artifacts the repositories containing this plugin artifacts need to be specified.

```
<pluginRepositories>
  <pluginRepository>
    <id>sf.net</id>
    <name>Lausdahl Snapshots</name>
    <releases>
      <enabled>true</enabled>
      <updatePolicy>daily</updatePolicy>
      <checksumPolicy>ignore</checksumPolicy>
    </releases>
    <snapshots>
      <enabled>true</enabled>
      <updatePolicy>never</updatePolicy>
      <checksumPolicy>ignore</checksumPolicy>
    </snapshots>
    <url>http://overturetraces.svn.sourceforge.net/viewvc/overturetraces</url>
    <layout>default</layout>
  </pluginRepository>
</pluginRepositories>
```

2 Check out and build

2.1 Check out

First check out this SVN URL to a working dir:

`https://overture.svn.sourceforge.net/svnroot/overture`
Terminal command:

```
svn co https://overture.svn.sourceforge.net/svnroot/overture overture
```

Now the repository should be checked out to your working dir where you should see the trunk folder.

2.2 Build

Steps:

1. Install the core artifacts[1]
2. Generate eclipse binary-plugins
3. Import projects into eclipse
4. Adjust the manifests in eclipse
5. Build eclipse plugins command line with maven - optional

[1] A known issue exist here look up the Core Renaming issue on page **??**

2.2.1 Install core artifacts

Go to trunk/core. Then execute:

```
mvn install
```

```
[INFO]
------------------------------------------------------------------------
[INFO] Reactor Summary:
[INFO]
------------------------------------------------------------------------
[INFO] Top-level POM for OvertureTool ..................... SUCCESS [5.714s]
[INFO] The VDM++ Standard Library ......................... SUCCESS [3.872s]
[INFO] The Overture Abstract Syntax ....................... SUCCESS [6.332s]
[INFO] The VDMunit Support Library ........................ SUCCESS [2.064s]
[INFO] The VDM++ parser ................................... SUCCESS [6.001s]
[INFO] The Overture Eclipse plugins ....................... SUCCESS [0.043s]
[INFO] Bi-directional OML to UML translator ............... SUCCESS [3.212s]
[INFO] Interface for VDM Tools ............................ SUCCESS [0.061s]
[INFO] API for VDM Tools - wrapper for corba .............. SUCCESS [1.762s]
[INFO] The VDMJ Interpreter ............................... SUCCESS [14.253s]
[INFO] Combinatorial Testing of VDM++ models .............. SUCCESS [5.573s]
[INFO] showtrace .......................................... SUCCESS [1.687s]
[INFO]
------------------------------------------------------------------------
[INFO]
------------------------------------------------------------------------
[INFO] BUILD SUCCESSFUL
[INFO]
------------------------------------------------------------------------
```

Listing 1: Overture core install result.

2.2.2 Generate eclipse binary-plug-ins

All eclipse "binary-plugins" need to have all depended projects compiled and copied into its lib folder and a corresponding manifest generated.
Execute the following maven goal inside the folder: trunk/org.overturetool.eclipse/

```
mvn psteclipse:eclipse-plugin
```

```
[INFO] Reactor Summary:
[INFO]
------------------------------------------------------------------------
[INFO] Eclipse Top-level (Core) ........................... SUCCESS [5.261s]
[INFO] Eclipse Plug-ins ................................... SUCCESS [0.030s]
[INFO] org.overturetool.eclipse.plugins.stdlib ............ SUCCESS [0.161s]
[INFO] org.overturetool.eclipse.plugins.umltrans.core ..... SUCCESS [0.076s]
[INFO] org.overturetool.eclipse.plugins.editor.core ....... SUCCESS [0.002s]
[INFO] org.overturetool.eclipse.plugins.traces.core ....... SUCCESS [0.137s]
[INFO] org.overturetool.eclipse.plugins.showtrace.core .... SUCCESS [0.368s]
[INFO] org.overturetool.eclipse.plugins.umltrans .......... SUCCESS [0.003s]
[INFO] org.overturetool.eclipse.plugins.debug ............. SUCCESS [0.006s]
[INFO] org.overturetool.eclipse.plugins.launching ......... SUCCESS [0.003s]
[INFO] org.overturetool.eclipse.plugins.editor.ui ......... SUCCESS [0.014s]
[INFO] org.overturetool.eclipse.plugins.debug.ui .......... SUCCESS [0.003s]
[INFO] org.overturetool.eclipse.plugins.traces ............ SUCCESS [0.005s]
[INFO]
```

```
------------------------------------------------------------------------
[INFO]
------------------------------------------------------------------------
[INFO] BUILD SUCCESSFUL
[INFO]
------------------------------------------------------------------------
```

Listing 2: Overture eclipse binary plug-ins updated.

Now the depended projects should be compiled and placed in the lib folder under the current plug-in. Now the manifest needs to be updated inside eclipse.

2.2.3 Import maven projects in eclipse

Go to the trunk and execute:

```
mvn eclipse:eclipse
```

```
[INFO]
------------------------------------------------------------------------
[INFO] Reactor Summary:
[INFO]
------------------------------------------------------------------------
[INFO] Top-level POM for OvertureTool ..................... SUCCESS [4.075s]
[INFO] The VDM++ Standard Library ......................... SUCCESS [0.119s]
[INFO] The Overture Abstract Syntax ....................... SUCCESS [0.046s]
[INFO] The VDMunit Support Library ........................ SUCCESS [0.041s]
[INFO] The VDM++ parser ................................... SUCCESS [0.024s]
[INFO] The Overture Eclipse plugins ....................... SUCCESS [0.021s]
[INFO] Bi-directional OML to UML translator ............... SUCCESS [0.037s]
[INFO] Interface for VDM Tools ............................ SUCCESS [0.023s]
[INFO] API for VDM Tools - wrapper for corba .............. SUCCESS [0.027s]
[INFO] The VDMJ Interpreter ............................... SUCCESS [0.595s]
[INFO] Combinatorial Testing of VDM++ models .............. SUCCESS [0.324s]
[INFO] showtrace .......................................... SUCCESS [0.057s]
[INFO] Super POM for OvertureTool ......................... SUCCESS [0.016s]
[INFO] Maven tools for overture ........................... SUCCESS [0.012s]
[INFO] VDM Tools Typecheck and Java code generate for Maven .. SUCCESS [8.452s]
[INFO] Eclipse Top-level (Core) ........................... SUCCESS [0.573s]
[INFO] Eclipse Plug-ins ................................... SUCCESS [0.054s]
[INFO] org.overturetool.eclipse.plugins.stdlib ............ SUCCESS [0.709s]
[INFO] org.overturetool.eclipse.plugins.umltrans.core ..... SUCCESS [0.090s]
[INFO] org.overturetool.eclipse.plugins.editor.core ....... SUCCESS [0.071s]
[INFO] org.overturetool.eclipse.plugins.traces.core ....... SUCCESS [0.049s]
[INFO] org.overturetool.eclipse.plugins.showtrace.core .... SUCCESS [0.070s]
[INFO] org.overturetool.eclipse.plugins.umltrans .......... SUCCESS [0.071s]
[INFO] org.overturetool.eclipse.plugins.debug ............. SUCCESS [0.040s]
[INFO] org.overturetool.eclipse.plugins.launching ......... SUCCESS [0.067s]
[INFO] org.overturetool.eclipse.plugins.editor.ui ......... SUCCESS [0.064s]
[INFO] org.overturetool.eclipse.plugins.debug.ui .......... SUCCESS [0.062s]
[INFO] org.overturetool.eclipse.plugins.traces ............ SUCCESS [0.051s]
[INFO]
------------------------------------------------------------------------
[INFO]
------------------------------------------------------------------------
[INFO] BUILD SUCCESSFUL
[INFO]
------------------------------------------------------------------------
```

Listing 3: Overture eclipse project files generated.

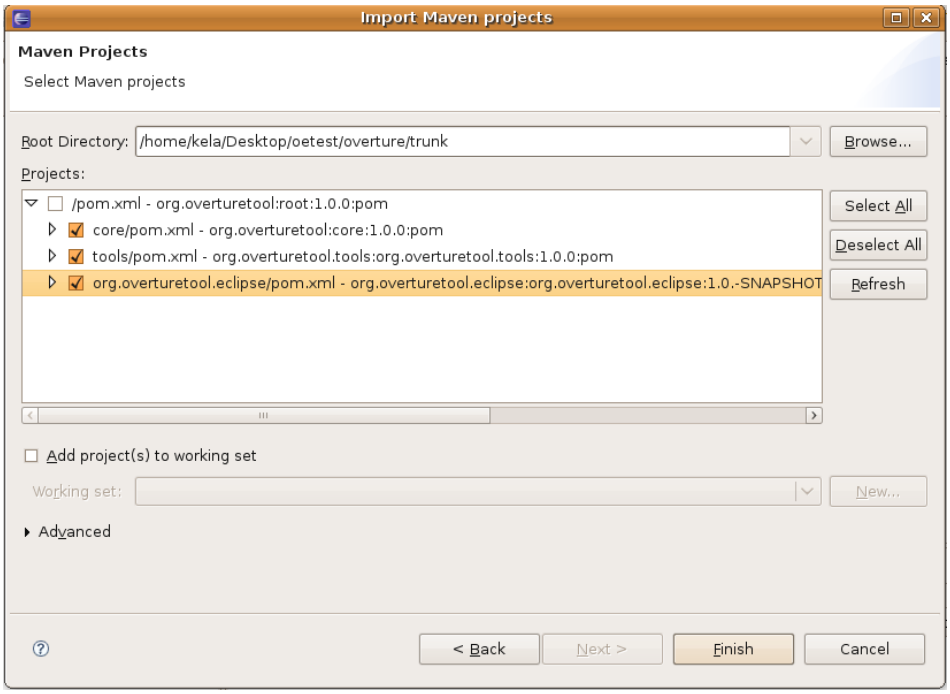

Figure 1: Import Maven Projects in Eclipse

Now all maven artifacts will have the corresponding eclipse project files to the pom file.
Open eclipse with Maven support.

Select the workspace of choice.

Select Import → General→ Maven Projects

Select the overture trunk directory

Now you should see: fig. **??**.
Do not select the root level POM named root.

Click finish It might take Eclipse some minutes to import and build the artifacts.

2.2.4 Adjust the manifests in eclipse

To make the projects build the manifests have to be updated. This is needed since the manifests are parsed at compile time and dependencies from the manifest is inserted runtime into the pom file.
At the time of writing there are the following binary eclipse plug-in projects:

- org.eclipse.plugins.stdlib
- org.eclipse.plugins.umltrans.core
- org.overture.eclipse.plugins.showtrace.core
- org.overture.eclipse.plugins.traces.core

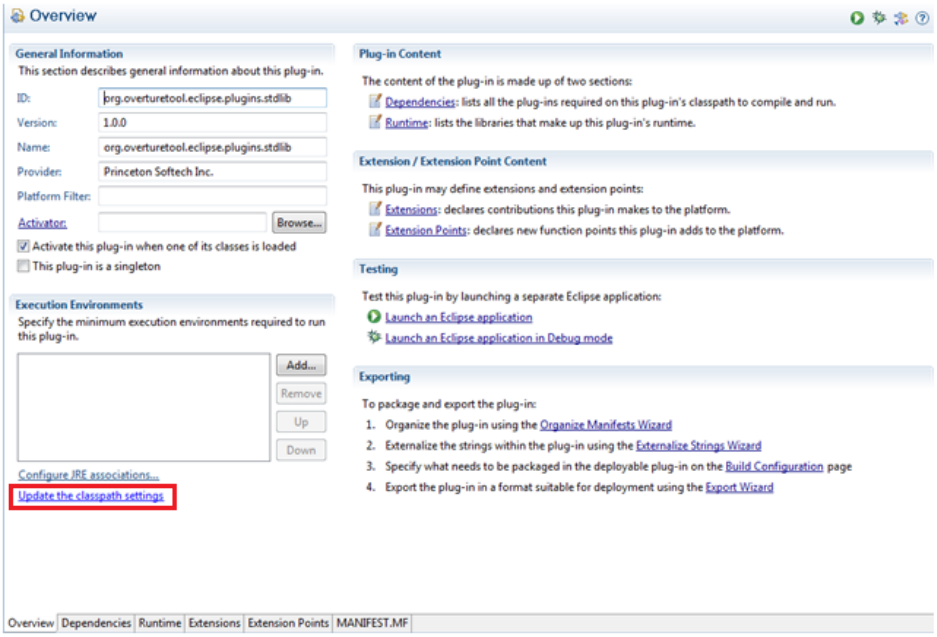

Figure 2: Update classpath in binary plug-in inside eclipse

Under each plug-in a folder called META-INF is located. Here the manifest file is stored. Open this file for each of the binary eclipse plug-in projects and de as followed:

Open the manifest and select "Overview → Update the classpath settings" - lower left corner.

The manifest editor inside eclipse can be seen in fig. **??**

Click Update the classpath settings

Now maven will build the artifacts and the errors should disappear. BUILD completed. If Eclipse still shows errors on some of the projects then do a Clean inside eclipse:

Project → Clean

2.2.5 Build eclipse plugins command line with maven - optional

Run this in the trunk/org.overturetool.eclipse folder

mvn install

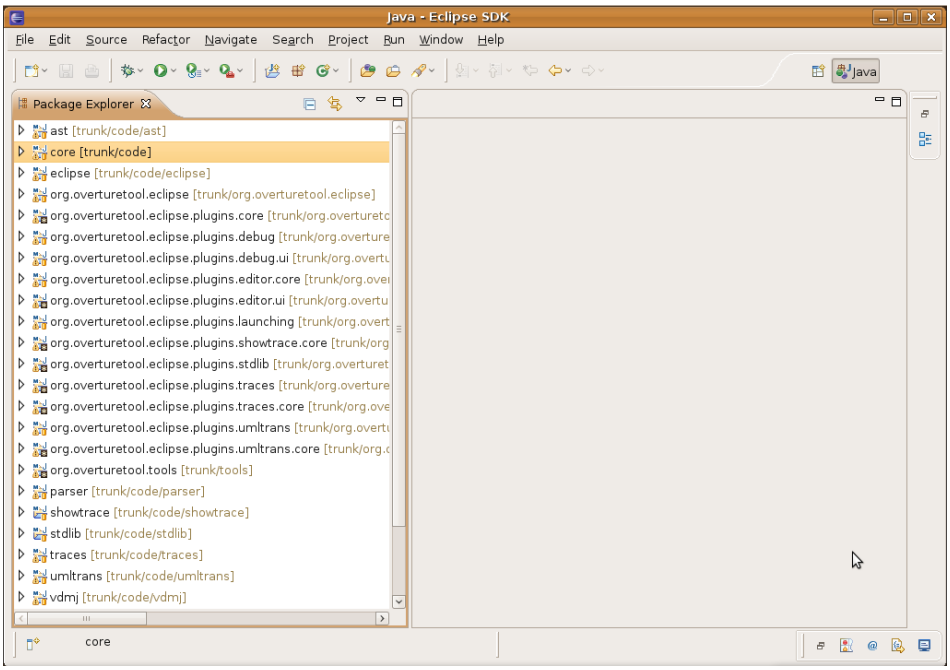

Figure 3: Eclipse plug-in build completed with no errors.

Appendix

2.3 Maven Settings file Overview

Here is a overview of the placement of the different elements in the Maven settings file:

```
<settings>
  <localRepository/>
  <interactiveMode/>
  <usePluginRegistry/>
  <offline/>
  <proxies>
    <proxy>
      <active/>
      <protocol/>
      <username/>
      <password/>
      <port/>
      <host/>
      <nonProxyHosts/>
      <id/>
    </proxy>
  </proxies>
  <servers>
    <server>
      <username/>
      <password/>
      <privateKey/>
      <passphrase/>
      <filePermissions/>
      <directoryPermissions/>
      <configuration/>
      <id/>
    </server>
```

```
</servers>
<mirrors>
  <mirror>
    <mirrorOf/>
    <name/>
    <url/>
    <id/>
  </mirror>
</mirrors>
<profiles>
  <profile>
    <activation>
      <activeByDefault/>
      <jdk/>
      <os>
        <name/>
        <family/>
        <arch/>
        <version/>
      </os>
      <property>
        <name/>
        <value/>
      </property>
      <file>
        <missing/>
        <exists/>
      </file>
    </activation>
    <properties/>
    <repositories>
      <repository>
        <releases>
          <enabled/>
          <updatePolicy/>
          <checksumPolicy/>
        </releases>
        <snapshots>
          <enabled/>
          <updatePolicy/>
          <checksumPolicy/>
        </snapshots>
        <id/>
        <name/>
        <url/>
        <layout/>
      </repository>
    </repositories>
    <pluginRepositories>
      <pluginRepository>
        <releases>
          <enabled/>
          <updatePolicy/>
          <checksumPolicy/>
        </releases>
        <snapshots>
          <enabled/>
          <updatePolicy/>
          <checksumPolicy/>
        </snapshots>
        <id/>
        <name/>
        <url/>
        <layout/>
      </pluginRepository>
```

```
    </pluginRepositories>
    <id/>
  </profile>
 </profiles>
 <activeProfiles/>
 <pluginGroups/>
</settings>
```

2.4 Settings file for Windows

```
<settings>
  <activeProfiles>
    <activeProfile>default</activeProfile>
  </activeProfiles>
  <profiles>
    <profile>
      <id>default</id>
      <activation>
        <activeByDefault>true</activeByDefault>
      </activation>
      <properties>
        <user.vdmtoolscmdpath>
          c:\Program Files\The VDM++ Toolbox v8.2b\bin\vppde.exe
        </user.vdmtoolscmdpath>
        <user.eclipseStartup>
          plugins\org.eclipse.equinox.common_3.4.0.v20080421-2006.jar
        </user.eclipseStartup>
      </properties>
      <repositories>
        <repository>
          <id>lausdahl.com</id>
          <name>Lausdahl Snapshots</name>
          <releases>
            <enabled>true</enabled>
            <updatePolicy>daily</updatePolicy>
            <checksumPolicy>ignore</checksumPolicy>
          </releases>
          <snapshots>
            <enabled>true</enabled>
            <updatePolicy>never</updatePolicy>
            <checksumPolicy>ignore</checksumPolicy>
          </snapshots>
          <url>http://maven2.lausdahl.com</url>
          <layout>default</layout>
        </repository>
        <repository>
          <id>sf.net</id>
          <name>Lausdahl Snapshots</name>
          <releases>
            <enabled>true</enabled>
            <updatePolicy>daily</updatePolicy>
            <checksumPolicy>ignore</checksumPolicy>
          </releases>
          <snapshots>
            <enabled>true</enabled>
            <updatePolicy>never</updatePolicy>
            <checksumPolicy>ignore</checksumPolicy>
          </snapshots>
          <url>http://overturetraces.svn.sourceforge.net/viewvc/overturetraces</url>
          <layout>default</layout>
        </repository>
      </repositories>
      <pluginRepositories>
```

```
      <pluginRepository>
        <id>sf.net</id>
        <name>Lausdahl Snapshots</name>
        <releases>
          <enabled>true</enabled>
          <updatePolicy>daily</updatePolicy>
          <checksumPolicy>ignore</checksumPolicy>
        </releases>
        <snapshots>
          <enabled>true</enabled>
          <updatePolicy>never</updatePolicy>
          <checksumPolicy>ignore</checksumPolicy>
        </snapshots>
        <url>http://overturetraces.svn.sourceforge.net/viewvc/overturetraces</url>
        <layout>default</layout>
      </pluginRepository>
    </pluginRepositories>
  </profile>
 </profiles>
</settings>
```

2.5 Settings file for Ubuntu / Linux

```
<settings>
  <activeProfiles>
    <activeProfile>default</activeProfile>
  </activeProfiles>
  <profiles>
    <profile>
      <id>default</id>
      <activation>
        <activeByDefault>true</activeByDefault>
      </activation>
      <properties>
        <user.vdmtoolscmdpath>
          c:\Program Files\The VDM++ Toolbox v8.2b\bin\vppde.exe
        </user.vdmtoolscmdpath>
        <user.eclipseStartup>
          plugins/org.eclipse.equinox.common_3.4.0.v20080421-2006.jar
        </user.eclipseStartup>
      </properties>
      <repositories>
        <repository>
          <id>lausdahl.com</id>
          <name>Lausdahl Snapshots</name>
          <releases>
            <enabled>true</enabled>
            <updatePolicy>daily</updatePolicy>
            <checksumPolicy>ignore</checksumPolicy>
          </releases>
          <snapshots>
            <enabled>true</enabled>
            <updatePolicy>never</updatePolicy>
            <checksumPolicy>ignore</checksumPolicy>
          </snapshots>
          <url>http://maven2.lausdahl.com</url>
          <layout>default</layout>
        </repository>
        <repository>
          <id>sf.net</id>
          <name>Lausdahl Snapshots</name>
          <releases>
            <enabled>true</enabled>
```

```
            <updatePolicy>daily</updatePolicy>
            <checksumPolicy>ignore</checksumPolicy>
          </releases>
          <snapshots>
            <enabled>true</enabled>
            <updatePolicy>never</updatePolicy>
            <checksumPolicy>ignore</checksumPolicy>
          </snapshots>
          <url>http://overturetraces.svn.sourceforge.net/viewvc/overturetraces</url>
          <layout>default</layout>
        </repository>
      </repositories>
      <pluginRepositories>
        <pluginRepository>
          <id>sf.net</id>
          <name>Lausdahl Snapshots</name>
          <releases>
            <enabled>true</enabled>
            <updatePolicy>daily</updatePolicy>
            <checksumPolicy>ignore</checksumPolicy>
          </releases>
          <snapshots>
            <enabled>true</enabled>
            <updatePolicy>never</updatePolicy>
            <checksumPolicy>ignore</checksumPolicy>
          </snapshots>
          <url>http://overturetraces.svn.sourceforge.net/viewvc/overturetraces</url>
          <layout>default</layout>
        </pluginRepository>
      </pluginRepositories>
    </profile>
  </profiles>
</settings>
```

2.6 Screen dumps of Eclipse plug-in installation

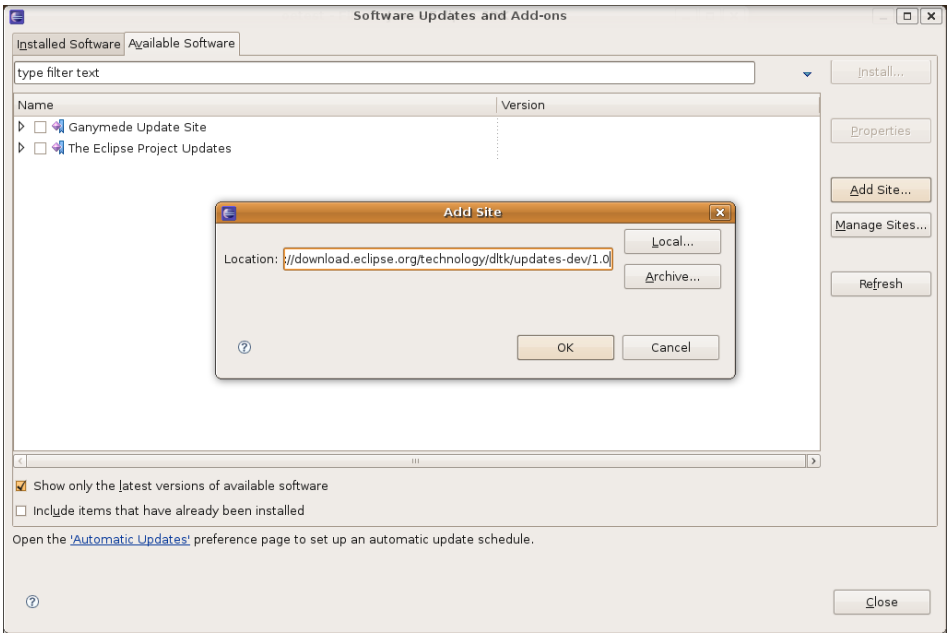

Figure 4: Add DLTK update site

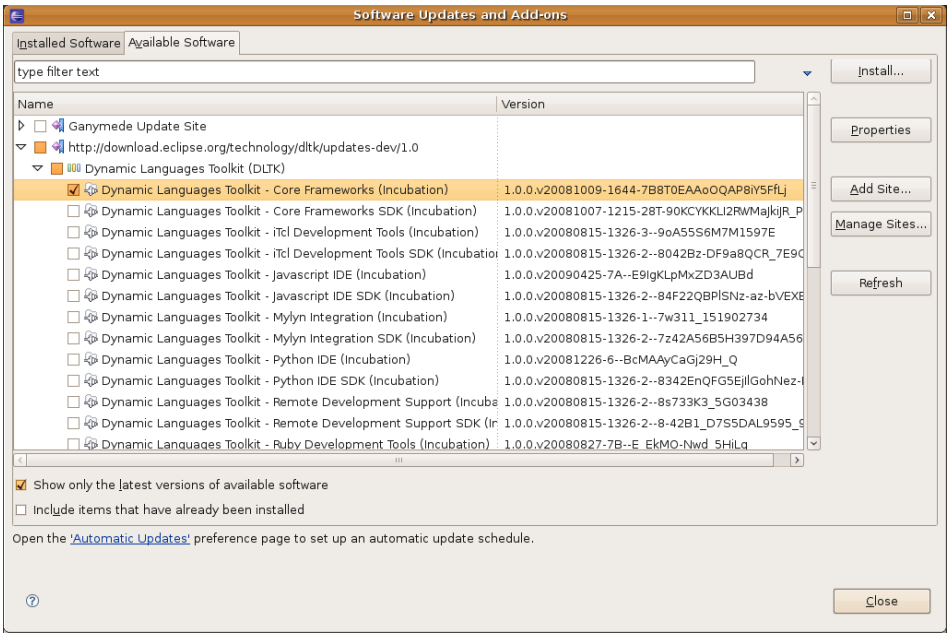

Figure 5: Select DLTK core

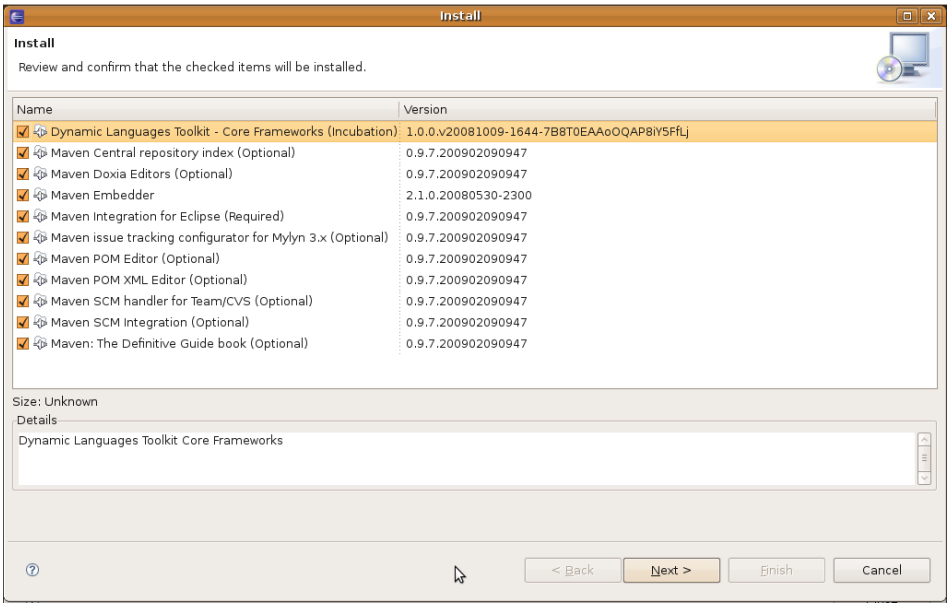

Figure 6: Confirm DLTK core and Maven

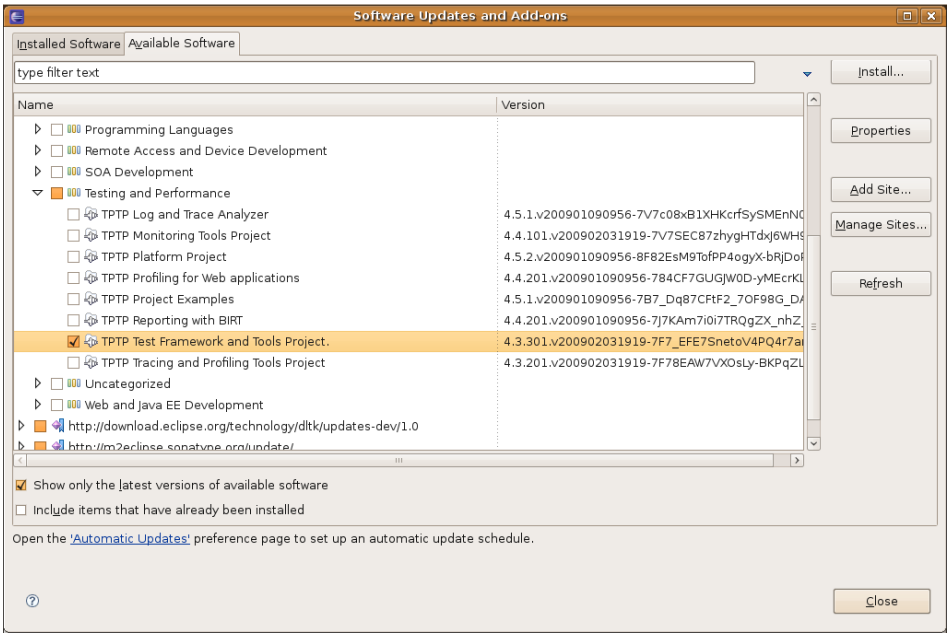

Figure 7: Select TPTP

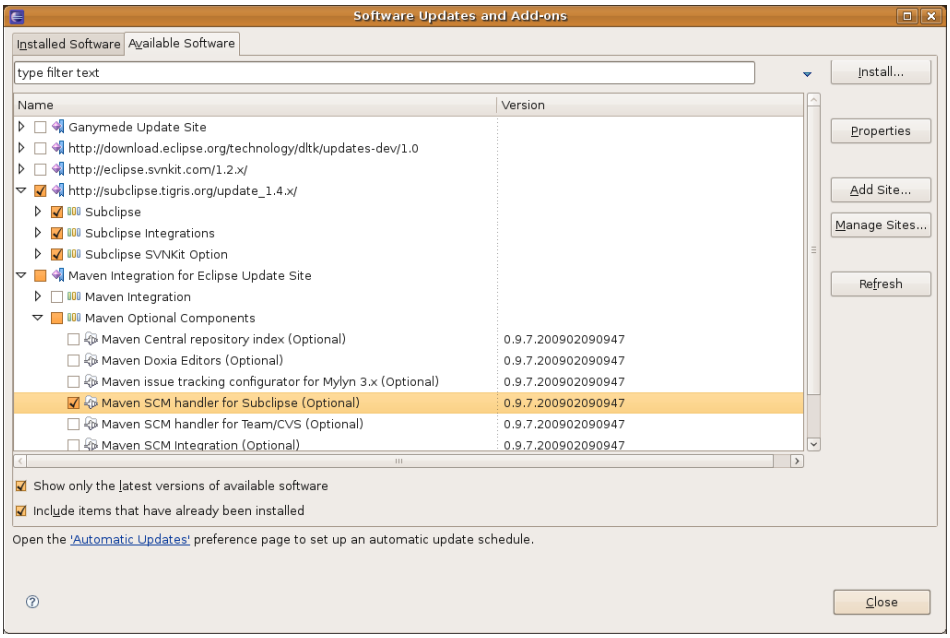

Figure 8: Select Subclipse and Maven for subclipse

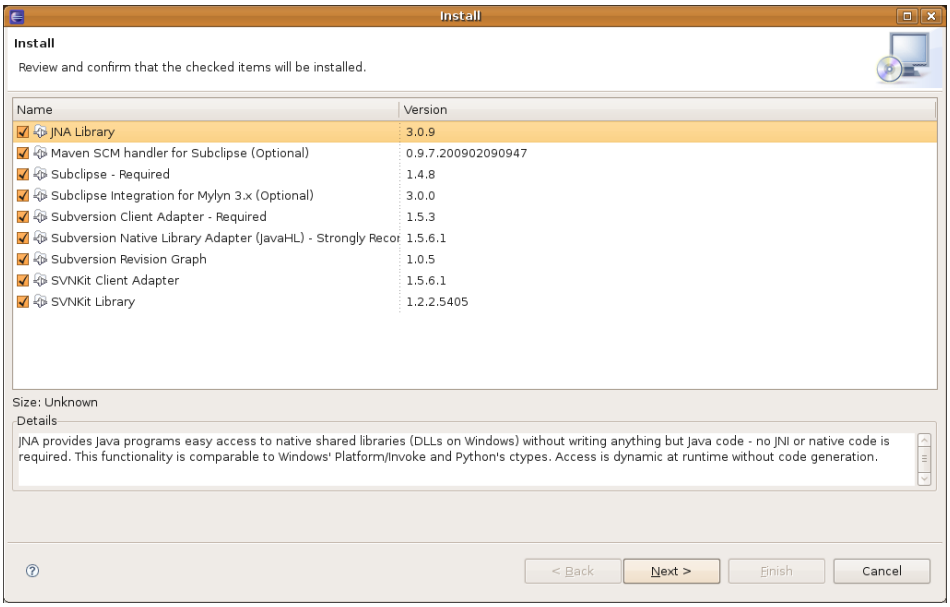

Figure 9: Confirm Subclipse and Maven for subclipse