OpenCDS Developer Workstation Configuration Instructions for OpenCDS 1.1 Release

Last updated April 1, 2014

- Document Version History
  - 8/28/2011
    - Created by Kensaku Kawamoto based on configuration instructions for OpenCDS alpha release
    - Changes include:
      - Use of Drools and Guvnor 5.2 instead of 5.1.1
      - More information on 64-bit OS configuration
      - Use of newest releases of software components
  - 11/8/2011
    - Updated by Kensaku Kawamoto, with verification on 32-bit and 64-bit Windows OSs
    - Use of Drools 5.3.0 instead of 5.2.0
  - 11/9/2011
    - Updated by Kensaku Kawamoto to include use of SilkSVN for 32-bit OSs as well
  - 11/10/2011
    - Updated by Kensaku Kawamoto to note need to use latest Tortoise SVN version
  - 11/16/2011
    - Updated by Kensaku Kawamoto to include preliminary Guvnor/Designer installation instructions and to make additional note regarding Tortoise SVN
  - 11/29/2011
    - Updated by David Shields to include solution to failed installation of JBoss Application Server in a path including "Program Files (x86)".
  - 12/16/2011
    - Updated by David Shields to include alternate installation paths, 2.0 version of JBoss Designer, alternate ports for JBoss, and availability of OpenCDS 1.0 binaries.
  - 12/17/2011
    - Updated by David Shields to include information on extended default timeout for JBoss AS 7.0.2
  - 11/14/2012
    - Major update by David Shields to reflect latest versions of all components, including Drools 5.4, JBPM Designer 5.3, and JBoss 7.1.x
  - 1/11/2013
    - Updated to reflect Drools 5.5, Designer 5.4
  - 5/24/2013
    - Updated to move Guvnor installation instructions to separate document

Table of Contents

Note: These instructions have been verified via installations onto a Windows XP 32-bit machine and two Windows 7 64-bit machines. If you have Windows User Access Control turned on, you may need to disable it in order to complete the installation properly.

We also have reports that OpenCDS has been successfully installed on Linux and Macintosh machines, by users who are experts in those environments.

Also note: these instructions do not encompass setting up an Apelon DTS terminology service instance. Setting up such an instance is not required for use of OpenCDS. Documentation on setting up an Apelon DTS instance can be found at http://apelon-dts.sourceforge.net/.

NOTE: we have created a common, publicly accessible Apelon DTS instance for maintaining a central set of concepts for use by the OpenCDS community. Software to access this is included in the OpenCDS source distribution.

I. Installation and Setup of Infrastructure Components

A. Install Java

1) **Install Java SE SDK version 6** (or 7 if you are comfortable updating the POM files for Maven)
   a) Download appropriate 32-bit or 64-bit Java SE SDK from http://www.oracle.com/technetwork/java/javase/downloads/index.html.
      i) Current Maven POM configuration files expect Java 1.6. Experts may wish to use Java 1.7 and make appropriate changes to the POMs.
   b) Run executable, install using default options
      i) You can use a non-default installation directory if desired
   c) Control Panel → System → Advanced → Environment Variables → System Variables
      i) Set JAVA_HOME to Java SE SDK location
         (1) E.g., C:\Program Files\Java\jdk1.6.0_26
      ii) Add %JAVA_HOME%\bin to Path
         (1) In this example, was added to beginning of Path

B. Install Maven

1) **Download .zip file of latest stable Maven** build from http://maven.apache.org/download.html
   a) These instructions have been tested for both Maven 3.0.3 at http://www.apache.org/dyn/closer.cgi/maven/binaries/apache-maven-3.0.3-bin.zip and Maven version 2.2.1 at http://www.apache.org/dyn/closer.cgi/maven/binaries/apache-maven-2.2.1-bin.zip
   b) Unzip archive
   c) Follow instructions in README.txt, provided below
   d) Directory specifications below are not mandatory, but recommended for consistency purposes
      i) Place unzipped folder (apache-maven-3.0.3) in following directory: C:\Program Files\Apache Software Foundation
   e) Control Panel → System → Advanced → Environment Variables → System Variables
      i) Set MAVEN_HOME to "C:\Program Files\Apache Software Foundation\apache-maven-3.0.3"
      ii) Add %MAVEN_HOME%\bin to Path
         (1) In this example, was added to beginning of Path

C. Install Apache CXF

1) **Download latest binary version of Apache CXF** from http://cxf.apache.org/download.html
   a) Unzip and move the apache-cxf-2.X.X folder to be under C:\Program Files\Apache Software Foundation

D. Install Tortoise SVN

1) **Download TortoiseSVN** from http://tortoisesvn.net/downloads.html

a) Note: if you already have TortoiseSVN installed, get the latest version if you plan to update Subclipse (the Eclipse Subversion plugin) to the latest version. You may run into some problems when using TortoiseSVN to check out the OpenCDS project if TortoiseSVN uses a Subversion version that is older than the Subversion version used by Eclipse.

b) Install using default options, then restart machine

E. Install Tomcat

1) **Download latest Tomcat** using "32-bit/64-bit Windows Service Installer" from relevant download page at http://tomcat.apache.org/
   a) OpenCDS has been tested using both Apache Tomcat versions 6 and 7, you can use either
   b) Accept defaults
   c) Installation directory may be changed if desired
   d) Set Tomcat Administrator Login to something secure
      i. You may even prefer to remove the administrator login if you have physical access to the server.
2) Control Panel → System → Advanced → Environment Variables → System Variables
   a) Set CATALINA_HOME to, e.g., "`C:\Program Files\Apache Software Foundation\Tomcat 7.0`"
   b) Add %CATALINA_HOME%\bin to Path
      i. In this example, was added to beginning of Path

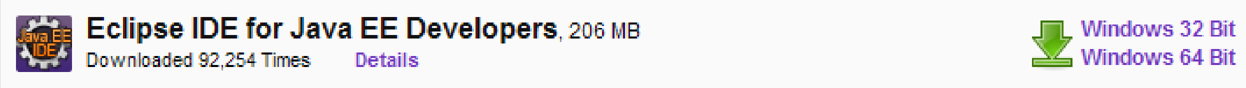
F. Install Latest Stable Version of Eclipse

1) **Download Eclipse** from http://www.eclipse.org/downloads/

a) Compatibility note: as of 11/8/2011, the JBoss Drools Eclipse Workbench is not available as a stable release for Eclipse version 3.7.X (Eclipse Indigo). Therefore, it is recommended that Eclipse Helios (version 3.6.X) or newer Eclipse is used instead.

i) **Update** as of 11/14/2012: *the above issue is no longer a problem*

b) In this example, installed Eclipse Helios version 3.6.2 from http://www.eclipse.org/downloads/packages/release/helios/sr2



Eclipse IDE for Java EE Developers, 206 MB
Downloaded 92,254 Times

c) Select 32 or 64-bit version, whichever is appropriate for your machine/OS.

2) **Create an Eclipse instance** by unzipping the zip archive into a directory of choice

a) **NOTE:** if you already have eclipse installed (e.g., at `C:\Program Files\eclipse`) for other projects you are working on, we **strongly** recommend that you install eclipse in a different directory (e.g., `C:\Program Files\eclipseOpenCDS`). This way, you will not impact any existing code or Eclipse settings (e.g., with regard to plugins). Furthermore, in the event that you need to uninstall and reinstall OpenCDS, you will be able to delete the OpenCDS eclipse folder without impacting any other projects.

b) The default directory used for these instructions is `C:\Program Files\eclipse`, which is where the Eclipse instance will be created if you unzip the zip archive directly into `C:\Program Files`

3) Notes

a) You do not need to modify your path or add environment variables to use eclipse

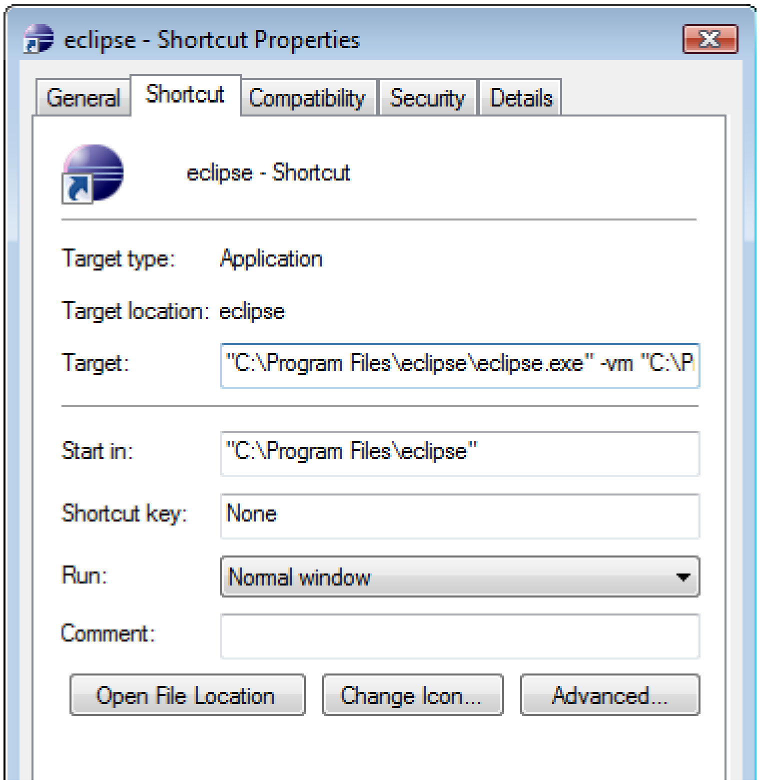
b) Read the release notes (especially if you have problems) found at `C:\Program Files\eclipse\readme\readme_eclipse.html`

c) Remember that you can often fix weird problems in eclipse from a command prompt by running "`eclipse -clean`" which will start eclipse with a clean slate (and make it forget all the in-process state of files you were currently working on)

4) **Create a shortcut for Eclipse that passes in Java Virtual Machine parameters** that a) ensures that the JDK is used rather than the JRE and b) ensures sufficient memory is allocated. Then, use the shortcut for launching Eclipse. Different shortcuts can be created if different Eclipse configurations are desired.

a) E.g., edit target of shortcut to:

```
"C:\Program Files\eclipse\eclipse.exe" -vm "C:\Program Files\Java\jdk1.6.0_26\bin" -
vmargs -Xmx900m
```



eclipse - Shortcut

Target type: Application

Target location: eclipse

Target: "C:\Program Files\eclipse\eclipse.exe" -vm "C:\P

Start in: "C:\Program Files\eclipse"

Shortcut key: None

Run: Normal window

Comment:

- b) Alternatively, the `C:\Program Files\eclipse\eclipse.ini` can be edited.
  - i) For more information, see http://wiki.eclipse.org/FAQ_How_do_I_increase_the_heap_size_available_to_Eclipse%3F, http://wiki.eclipse.org/Eclipse.ini.

5) When first starting Eclipse, **set workspace to appropriate directory**
   - a) Suggested location is "`C:\OpenCDS`"
   - b) An alternate suggested location if installing multiple versions of OpenCDS on the same machine is "`C:\OpenCDS_[version]`"
     - i) e.g., "`C:\OpenCDS_v1.0_preview`", "`C:\OpenCDS_alpha`", "`C:\OpenCDS_v1.1.2`", etc.

G. Install Eclipse plugins

Eclipse has built-in tools to locate and install plug-ins, and to update them when necessary. Use the built-in tools whenever possible.

There are two ways to do this. The new method is called Eclipse Marketplace. Use it whenever you can by going to

Help → Eclipse Marketplace

The traditional method is more tedious, and will need to be used if you can't find the software you are looking for in the Eclipse Marketplace. It is also used for some plugins that don't install all the required options when run from the Marketplace menu. This method is used by going to

Help → Install New Software

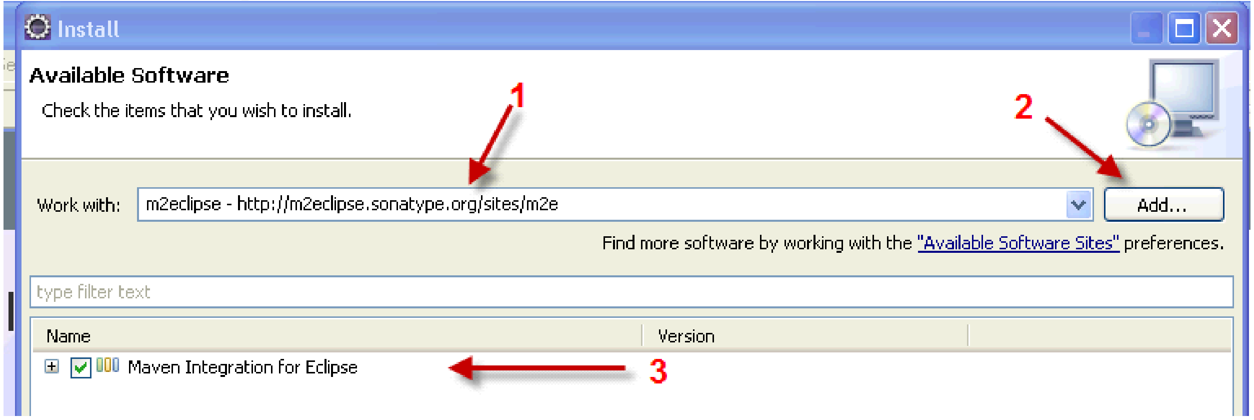
a) Install M2Eclipse

i) Before you can use Maven from Eclipse, you will need to make sure that you are running Eclipse from the JRE that is part of a JDK virtual machine. One safe way to do this is to create a shortcut for running Eclipse, and adding the path to the JDK in the shortcut target, as described above under the Eclipse installation instructions.

ii) From **Eclipse Marketplace** – DO NOT DO THIS! You will need to do the initial install the traditional way, as described next (the Eclipse Marketplace does not currently allow you to do a base install of m2eclipse)…

iii) From **Install New Software**

(1) Help → Install New Software


Install

Available Software

Check the items that you wish to install.

Work with: m2eclipse - http://m2eclipse.sonatype.org/sites/m2e

Find more software by working with the "Available Software Sites" preferences.

Name	Version
☑ Maven Integration for Eclipse	

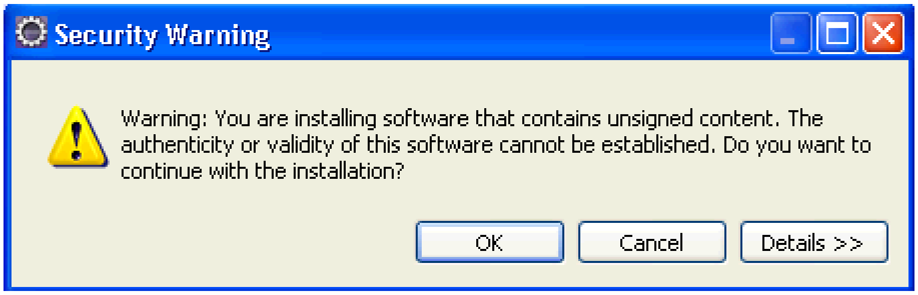
Type just the URL (http://m2eclipse.sonatype.org/sites/m2e) (1), click the Add button (2), type in "`m2eclipse`" for the name, and hit "okay" for adding the repository. You should then click in the box beside "Maven Integration for Eclipse (3), and click the Next button at the bottom of the Install window.

It will verify your selection (making sure dependencies are present), and then give you a chance to go back to add dependencies, or continue by clicking Next

Accept the license and click Finish.

Restart Eclipse if prompted.

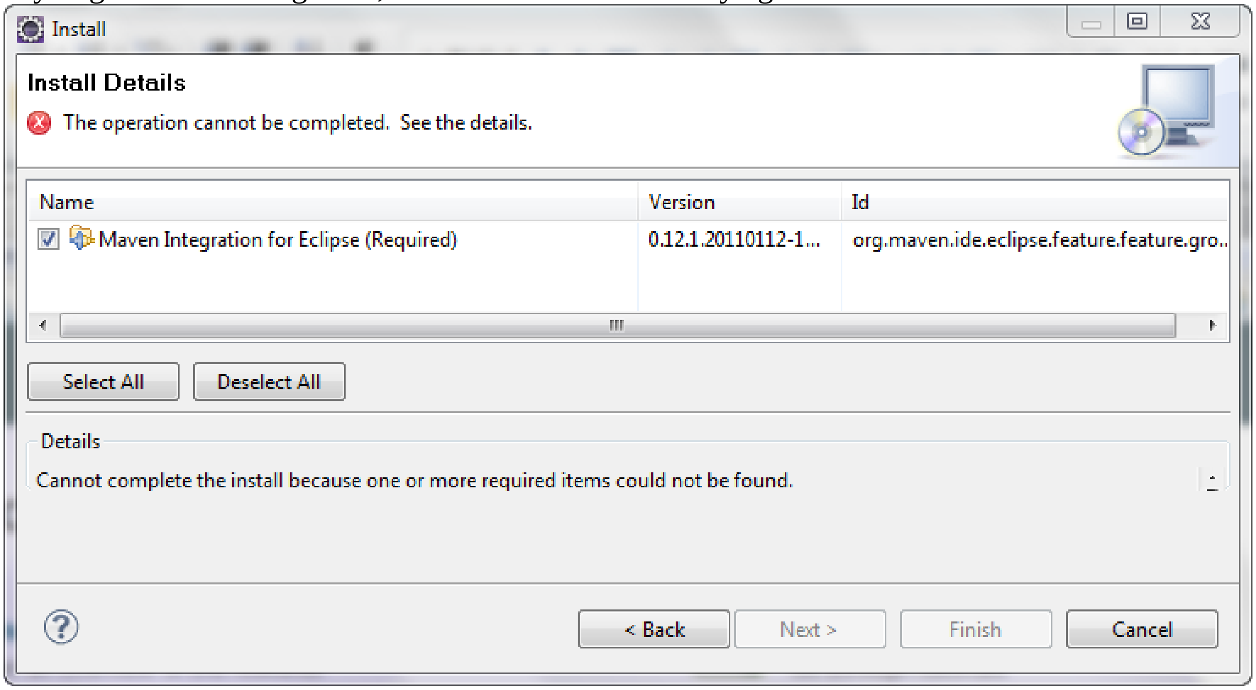
If you get the following warning, click OK to ignore.



Security Warning

Warning: You are installing software that contains unsigned content. The authenticity or validity of this software cannot be established. Do you want to continue with the installation?

If you get the following error, restart the machine and try again.



Install

Install Details

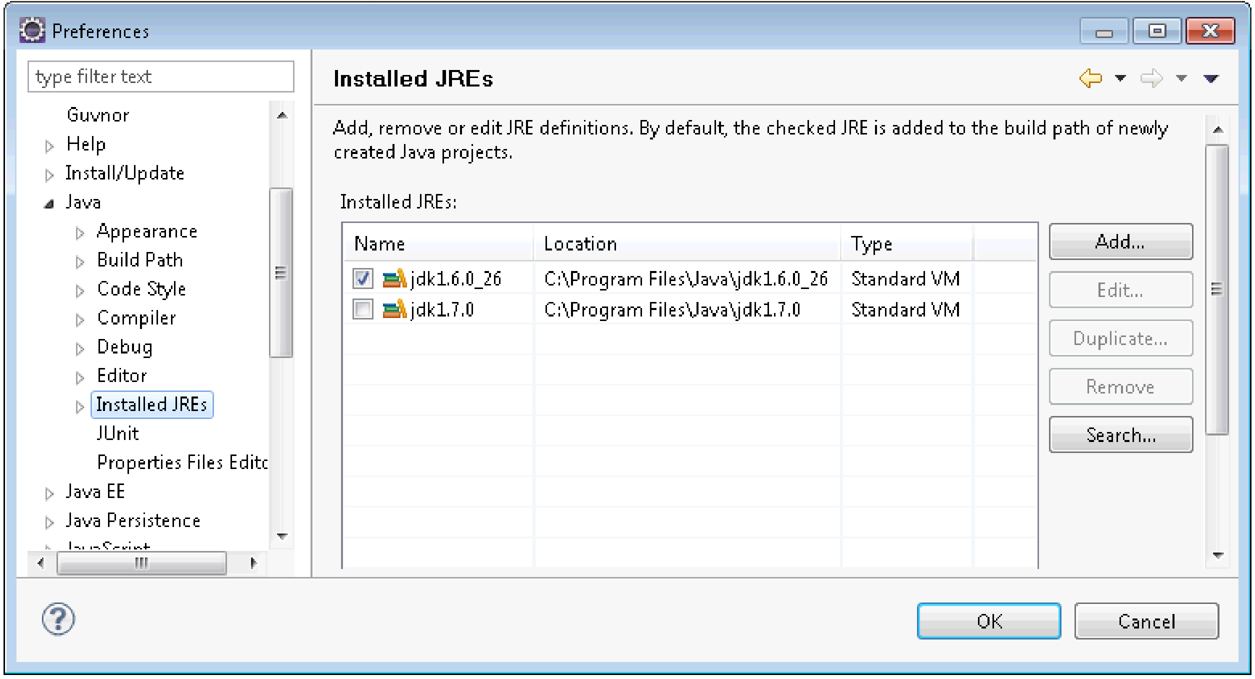
The operation cannot be completed. See the details.

Name	Version	Id
☑ Maven Integration for Eclipse (Required)	0.12.1.20110112-1...	org.maven.ide.eclipse.feature.feature.gro...

Details

Cannot complete the install because one or more required items could not be found.

b) **Make sure Eclipse is pointed to a JDK** rather than a JRE in Eclipse Preferences (Windows → Preferences → Java → Installed JREs and Windows → Preferences → Java → Installed JREs → Execution Environments)

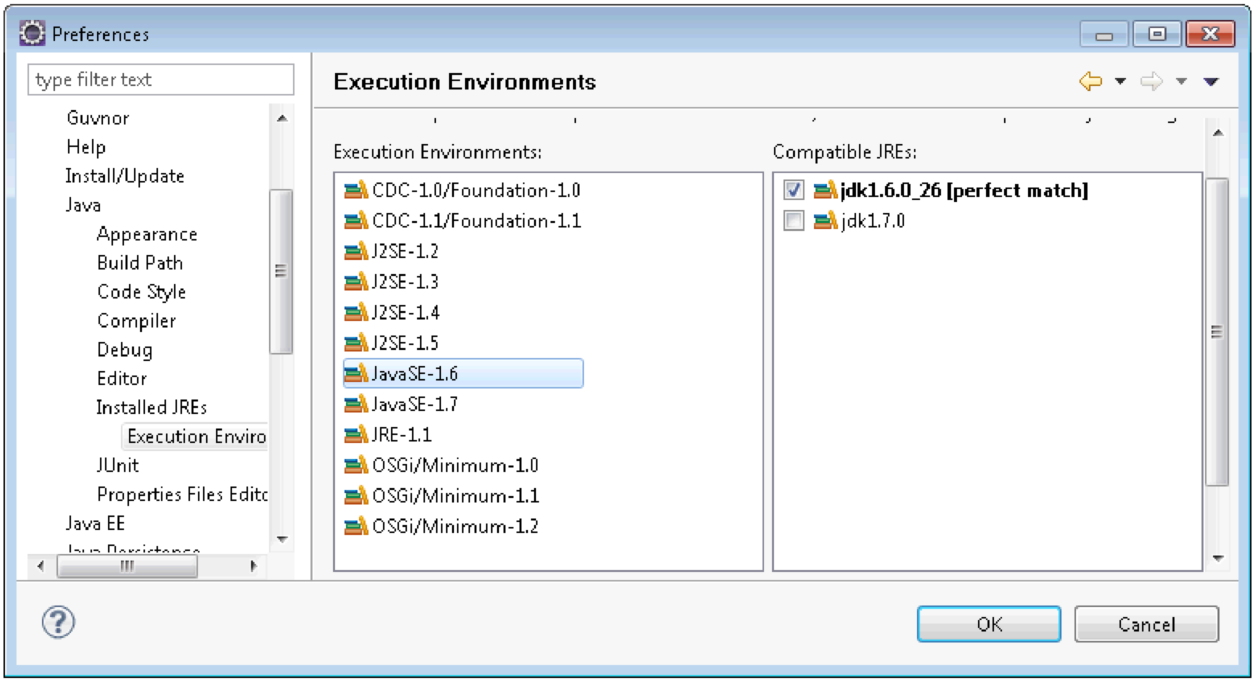


Installed JREs

Add, remove or edit JRE definitions. By default, the checked JRE is added to the build path of newly created Java projects.

Installed JREs:

Name	Location	Type
☑ jdk1.6.0_26	C:\Program Files\Java\jdk1.6.0_26	Standard VM
□ jdk1.7.0	C:\Program Files\Java\jdk1.7.0	Standard VM



Execution Environments

Execution Environments:

- CDC-1.0/Foundation-1.0
- CDC-1.1/Foundation-1.1
- J2SE-1.2
- J2SE-1.3
- J2SE-1.4
- J2SE-1.5
- JavaSE-1.6
- JavaSE-1.7
- JRE-1.1
- OSGi/Minimum-1.0
- OSGi/Minimum-1.1
- OSGi/Minimum-1.2

Compatible JREs:

- ☑ **jdk1.6.0_26 [perfect match]**
- □ jdk1.7.0

c) **Install Subclipse**

i) From **Eclipse Marketplace**: DO NOT DO THIS! It installs correctly, but is missing some important functionality.

   (1) UPDATE: The install from Eclipse Marketplace has apparently been fixed.

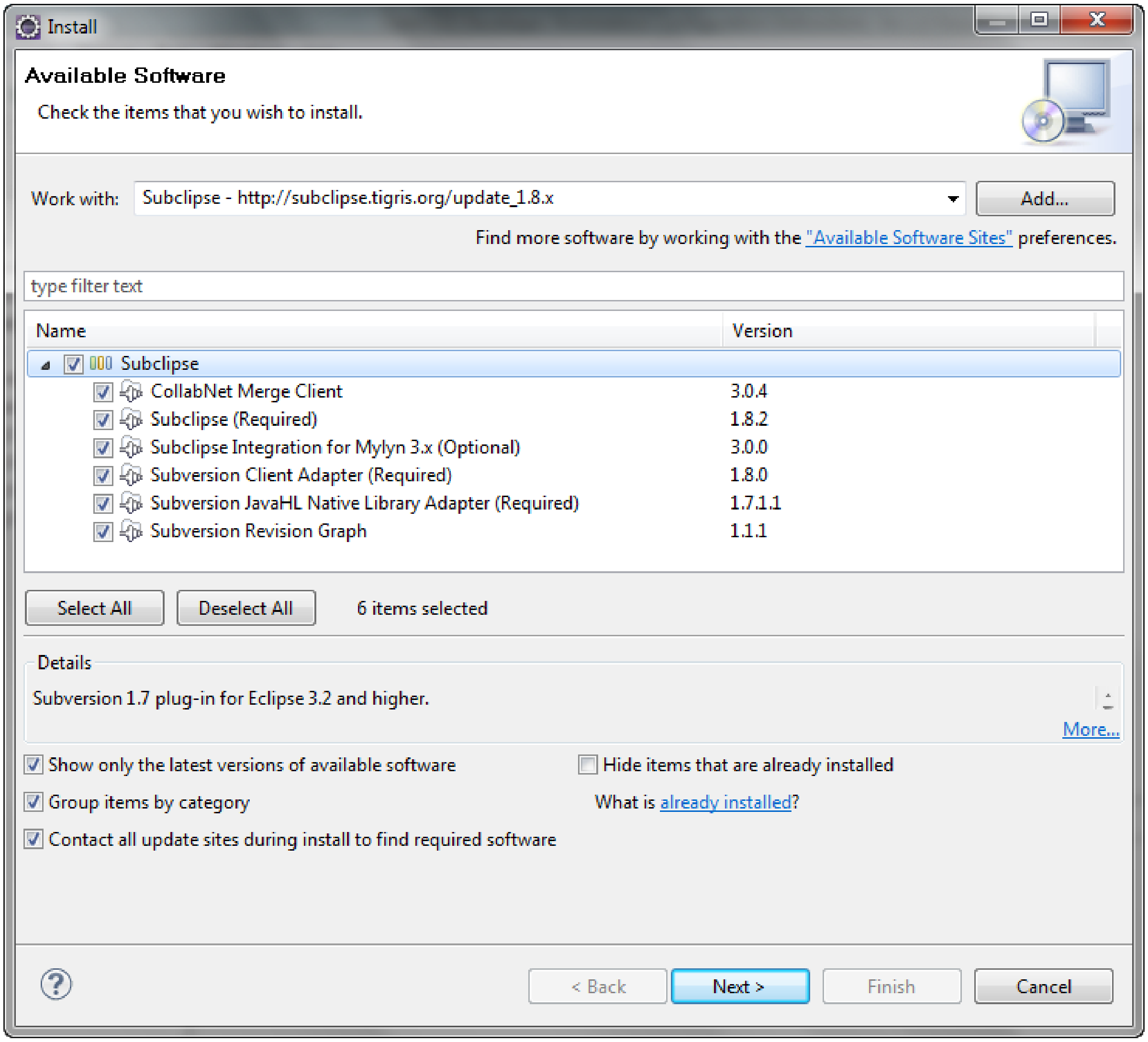
ii) From **Install New Software**

   (1) Select all options from the menu after you enter / select the path as shown below

      (a) URL = `http://subclipse.tigris.org/update_1.8.x`

      (b) Name = anything; suggest "Subclipse"

   (2)



Available Software

Check the items that you wish to install.

Work with: Subclipse - http://subclipse.tigris.org/update_1.8.x

Find more software by working with the "Available Software Sites" preferences.

Name	Version
☑ Subclipse	
☑ CollabNet Merge Client	3.0.4
☑ Subclipse (Required)	1.8.2
☑ Subclipse Integration for Mylyn 3.x (Optional)	3.0.0
☑ Subversion Client Adapter (Required)	1.8.0
☑ Subversion JavaHL Native Library Adapter (Required)	1.7.1.1
☑ Subversion Revision Graph	1.1.1

6 items selected

Details

Subversion 1.7 plug-in for Eclipse 3.2 and higher.

☑ Show only the latest versions of available software

☐ Hide items that are already installed

☑ Group items by category

What is already installed?

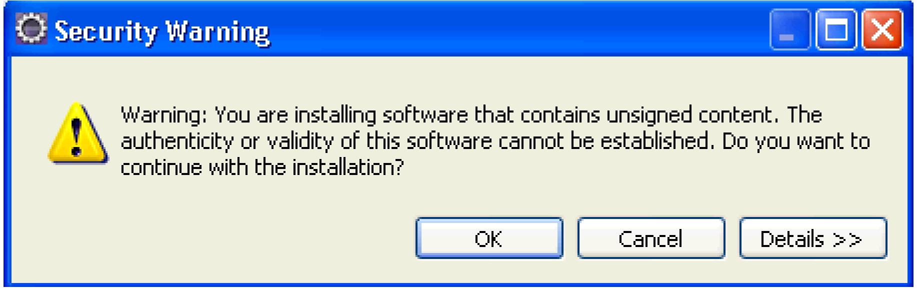
☑ Contact all update sites during install to find required software

   (3) Click on Next

      (a) It will verify your selection (making sure dependencies are present), and then give you a chance to go back to add dependencies, or continue by clicking Next.

   (4) Accept the license and click Finish

(5) If you get the following warning, click OK to ignore.



Warning: You are installing software that contains unsigned content. The authenticity or validity of this software cannot be established. Do you want to continue with the installation?

(6) Restart Eclipse if prompted

d) Set Eclipse Preferences for Subversion

i) Window → Preferences → Team → SVN

ii) Select a client for the SVN Interface

(1) If no clients are listed, you probably installed Subclipse from the Marketplace, and didn't get the client. Go back to the step about installing Subclipse and get the client.

(2) For this example, JavaHL is selected

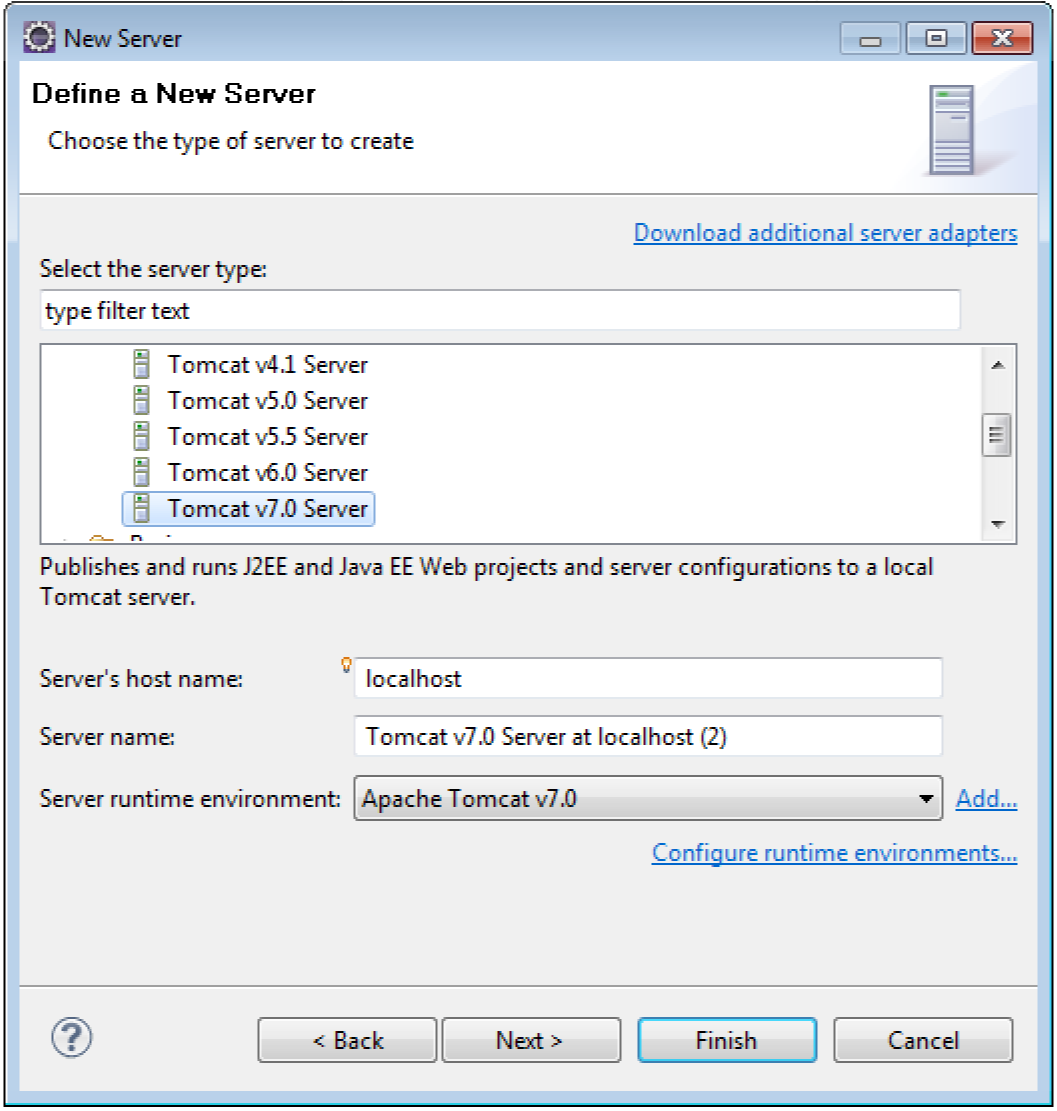
iii) Window → Preferences → Team → SVN → Menu icons

(1) Select Tortoise SVN

e) Set up Tomcat Server in Eclipse

i) File → New → Other → Server → Server

(1) Select Apache Tomcat (e.g., Tomcat 7.0) and use default settings



New Server

Define a New Server

Choose the type of server to create

Select the server type:

type filter text

Tomcat v4.1 Server

Tomcat v5.0 Server

Tomcat v5.5 Server

Tomcat v6.0 Server

Tomcat v7.0 Server

Publishes and runs J2EE and Java EE Web projects and server configurations to a local Tomcat server.

Server's host name: localhost

Server name: Tomcat v7.0 Server at localhost (2)

Server runtime environment: Apache Tomcat v7.0

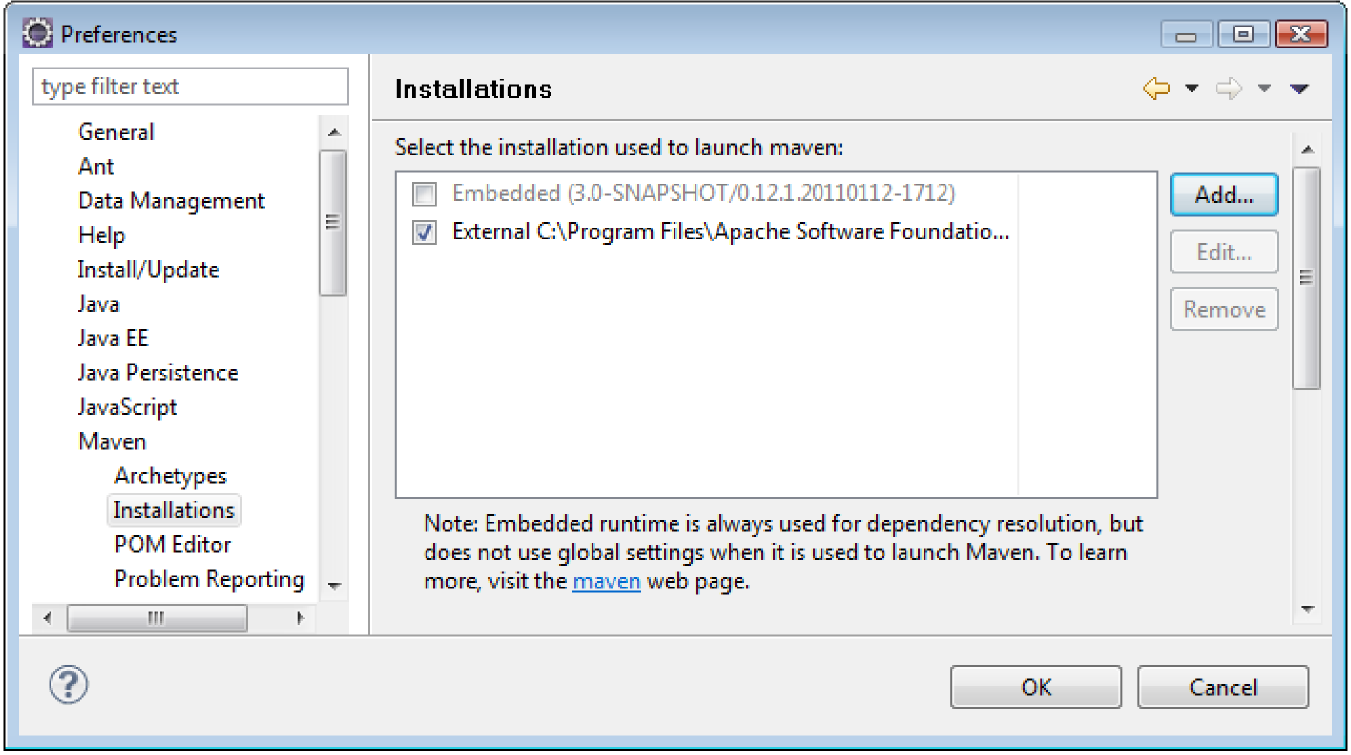
Download additional server adapters
Add...
Configure runtime environments...
< Back
Next >
Finish
Cancel

(2)

ii) Select installation directory, Finish

f) Set Eclipse Preferences for Maven

i) Window → Preferences → Maven → Installations → Add → specify Maven root folder (e.g., `C:\Program Files\Apache Software Foundation\apache-maven-3.0.3`) → OK



Installations

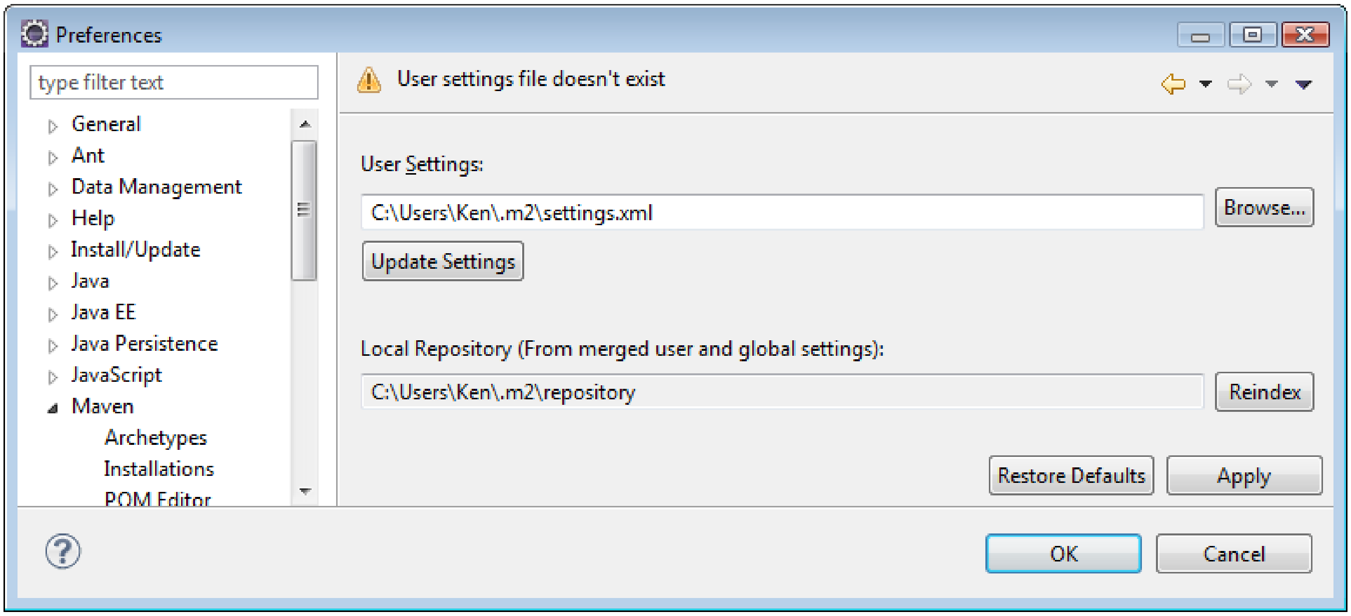
Select the installation used to launch maven:

☐ Embedded (3.0-SNAPSHOT/0.12.1.20110112-1712)

☑ External C:\Program Files\Apache Software Foundatio...

Note: Embedded runtime is always used for dependency resolution, but does not use global settings when it is used to launch Maven. To learn more, visit the maven web page.

ii) Create a settings.xml file in location specified in Eclipse under Window → Preferences → Maven → User Settings



User settings file doesn't exist

User Settings: C:\Users\Ken\.m2\settings.xml

Local Repository (From merged user and global settings): C:\Users\Ken\.m2\repository

iii) Sample XML (used for this example):

```
<?xml version="1.0" encoding="UTF-8"?>
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0
http://maven.apache.org/xsd/settings-1.0.0.xsd">
        <pluginGroups>
        </pluginGroups>
        <proxies>
        </proxies>
        <servers>
        </servers>
        <mirrors>
        </mirrors>
        <profiles>
        </profiles>
</settings>
```

iv) Restart Eclipse

v) Update central Maven repository index, as follows. Note that if you don't do it here, it will automatically take place later in the installation process. We recommend updating the index here, so that if there is an error, it is clear where the error occurred.

(1) Windows → Show View → Other → Maven → Maven Repositories

(a) Click Global Repositories → right click "central", select Update Index

(b) This may take a while; you can track progress by clicking on the progress icon on the bottom right of the screen ( )

(c) Sometimes, it takes multiple attempts before this completes. You may need to restart Eclipse in the process.

(d) If you see an error message of "Unable to update index for central," it may mean you need to turn off a Proxy connection (see http://www.ehow.com/how_6079770_using-proxy-server.html) or configure Proxy-based Maven connections (see http://maven.apache.org/guides/mini/guide-proxies.html). Once you have made the changes, verify that you do not get this error by repeating the steps above.

g) **Install Jboss Drools Eclipse Workbench** (*allows writing Drools rules and working with Drools Flow, and testing / debugging directly from Eclipse; part of JBoss Tools*).

i) **Automatic installation** from Eclipse

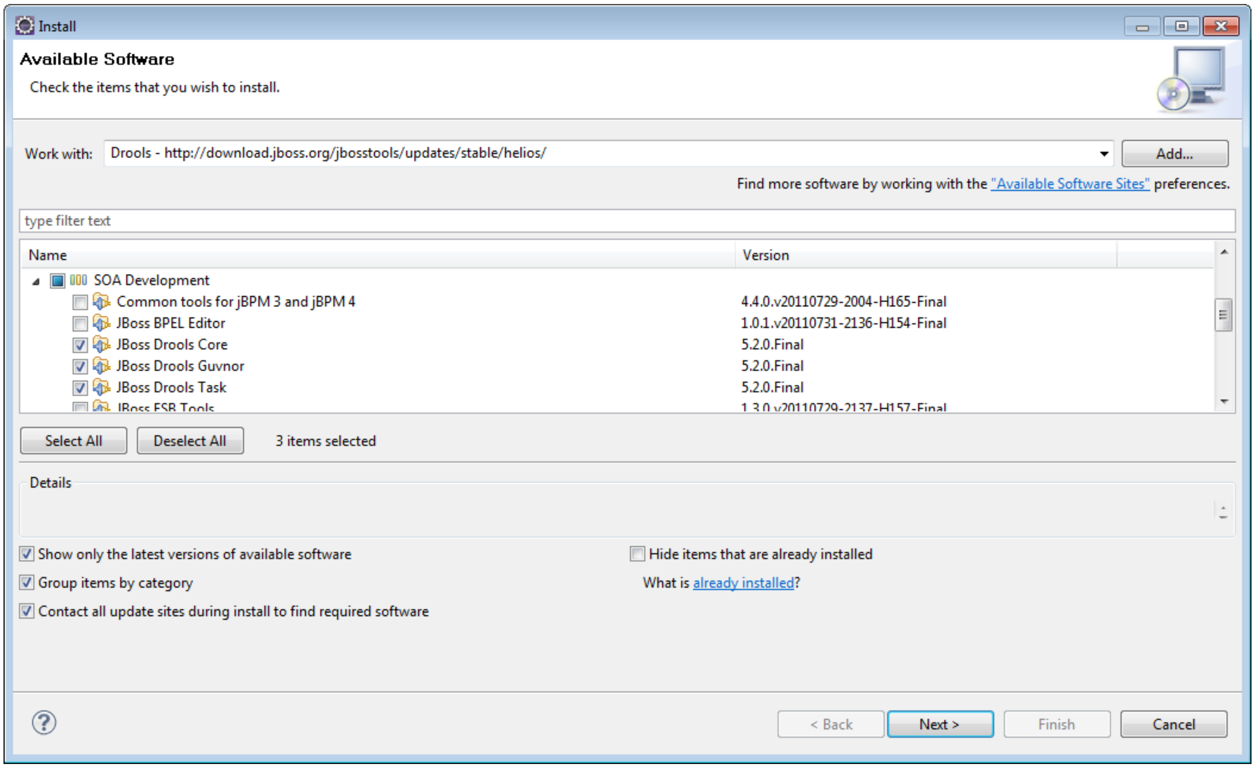
(1) Download Drools plugin for Eclipse

(a) Help → Install New Software

(b) Use appropriate URL found at http://www.jboss.org/tools/download.html. In this case, use http://download.jboss.org/jbosstools/updates/stable/helios/

(c) Name used in this example for repository: Drools

(d) Install the three Drools components indicated below under the SOA Development category



Available Software

Check the items that you wish to install.

Work with: Drools - http://download.jboss.org/jbosstools/updates/stable/helios/

Find more software by working with the "Available Software Sites" preferences.

Name	Version
■ SOA Development	
□ Common tools for jBPM 3 and jBPM 4	4.4.0.v20110729-2004-H165-Final
□ JBoss BPEL Editor	1.0.1.v20110731-2136-H154-Final
☑ JBoss Drools Core	5.2.0.Final
☑ JBoss Drools Guvnor	5.2.0.Final
☑ JBoss Drools Task	5.2.0.Final
□ JBoss ESB Tools	1.3.0.v20110729-2137-H157-Final

3 items selected

☑ Show only the latest versions of available software

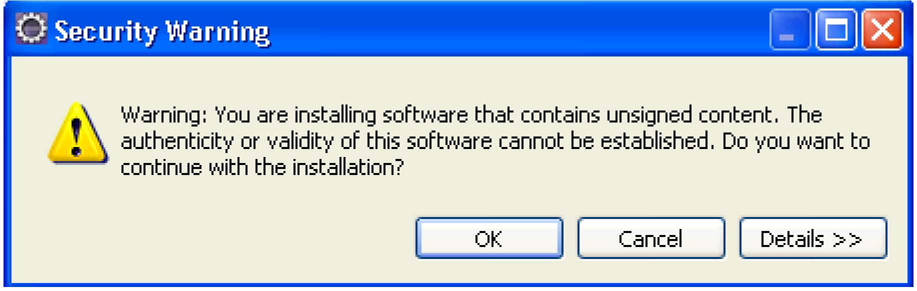
☑ Group items by category

☑ Contact all update sites during install to find required software

□ Hide items that are already installed

What is already installed?

(e) If you get the following warning, click OK to ignore.



Warning: You are installing software that contains unsigned content. The authenticity or validity of this software cannot be established. Do you want to continue with the installation?

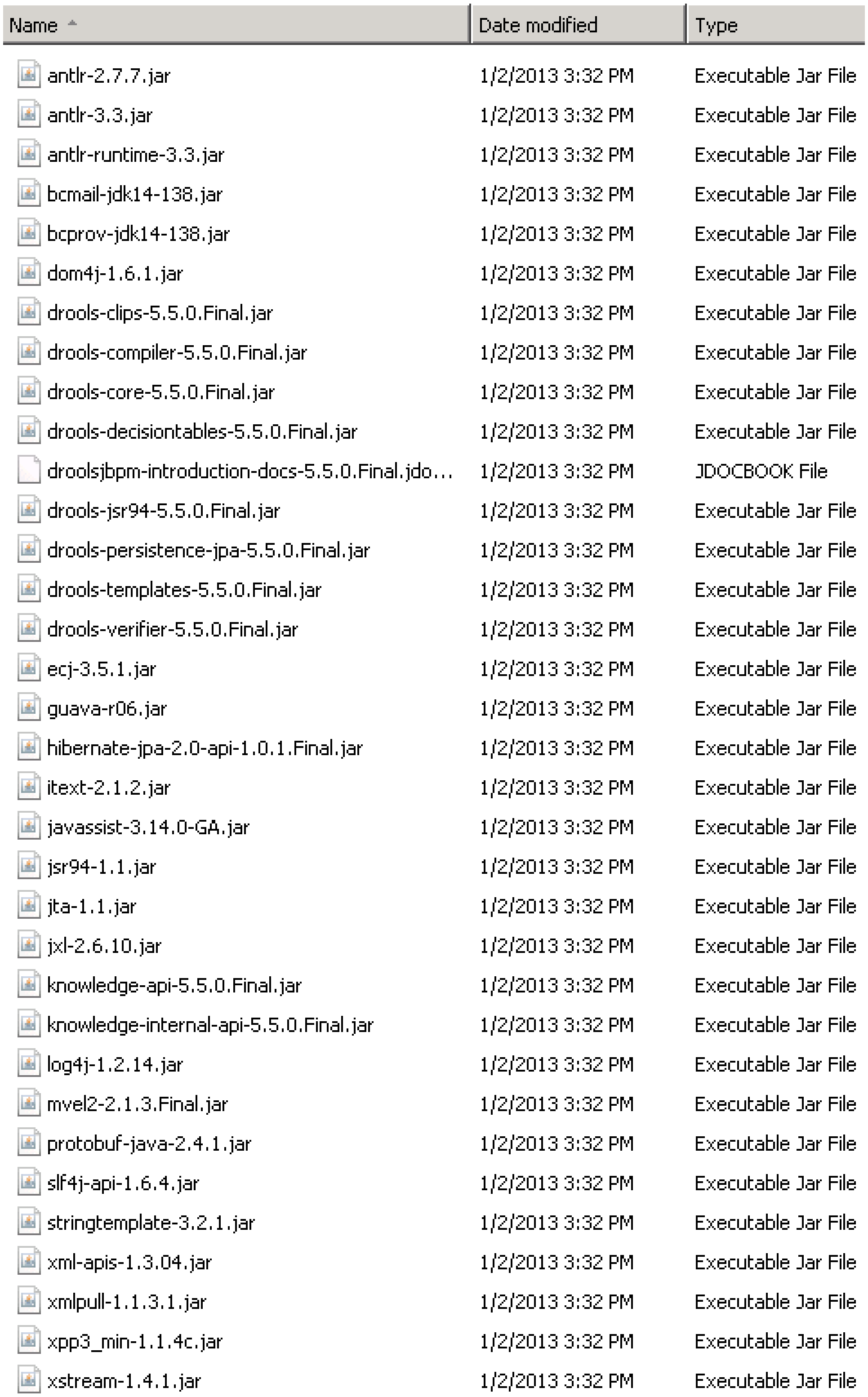
(f) Restart Eclipse if prompted

ii) **Download the Drools Runtime** version 5.5.0.Final

(1) Download file named **Drools Binaries** from http://www.jboss.org/drools/downloads.html, using link at http://download.jboss.org/drools/release/5.5.0.Final/drools-distribution-5.5.0.Final.zip

(2) Place contents of the **binaries** folder in the zip archive into a directory (`C:\Data\Drools\Drools5.5.0.Final` is used in this guide)

(3) E.g., this directory should look like the following:

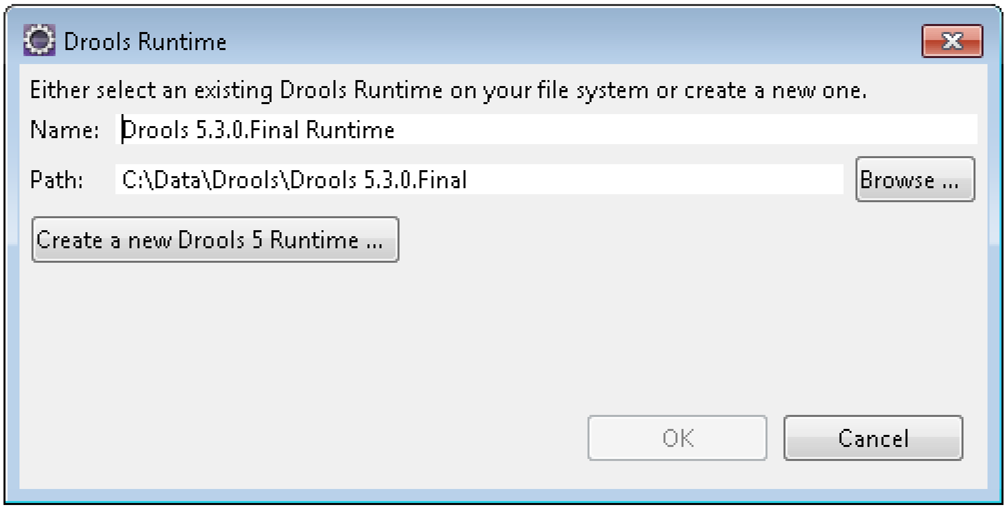


Name	Date modified	Type
antlr-2.7.7.jar	1/2/2013 3:32 PM	Executable Jar File
antlr-3.3.jar	1/2/2013 3:32 PM	Executable Jar File
antlr-runtime-3.3.jar	1/2/2013 3:32 PM	Executable Jar File
bcmail-jdk14-138.jar	1/2/2013 3:32 PM	Executable Jar File
bcprov-jdk14-138.jar	1/2/2013 3:32 PM	Executable Jar File
dom4j-1.6.1.jar	1/2/2013 3:32 PM	Executable Jar File
drools-clips-5.5.0.Final.jar	1/2/2013 3:32 PM	Executable Jar File
drools-compiler-5.5.0.Final.jar	1/2/2013 3:32 PM	Executable Jar File
drools-core-5.5.0.Final.jar	1/2/2013 3:32 PM	Executable Jar File
drools-decisiontables-5.5.0.Final.jar	1/2/2013 3:32 PM	Executable Jar File
droolsjbpm-introduction-docs-5.5.0.Final.jdo...	1/2/2013 3:32 PM	JDOCBOOK File
drools-jsr94-5.5.0.Final.jar	1/2/2013 3:32 PM	Executable Jar File
drools-persistence-jpa-5.5.0.Final.jar	1/2/2013 3:32 PM	Executable Jar File
drools-templates-5.5.0.Final.jar	1/2/2013 3:32 PM	Executable Jar File
drools-verifier-5.5.0.Final.jar	1/2/2013 3:32 PM	Executable Jar File
ecj-3.5.1.jar	1/2/2013 3:32 PM	Executable Jar File
guava-r06.jar	1/2/2013 3:32 PM	Executable Jar File
hibernate-jpa-2.0-api-1.0.1.Final.jar	1/2/2013 3:32 PM	Executable Jar File
itext-2.1.2.jar	1/2/2013 3:32 PM	Executable Jar File
javassist-3.14.0-GA.jar	1/2/2013 3:32 PM	Executable Jar File
jsr94-1.1.jar	1/2/2013 3:32 PM	Executable Jar File
jta-1.1.jar	1/2/2013 3:32 PM	Executable Jar File
jxl-2.6.10.jar	1/2/2013 3:32 PM	Executable Jar File
knowledge-api-5.5.0.Final.jar	1/2/2013 3:32 PM	Executable Jar File
knowledge-internal-api-5.5.0.Final.jar	1/2/2013 3:32 PM	Executable Jar File
log4j-1.2.14.jar	1/2/2013 3:32 PM	Executable Jar File
mvel2-2.1.3.Final.jar	1/2/2013 3:32 PM	Executable Jar File
protobuf-java-2.4.1.jar	1/2/2013 3:32 PM	Executable Jar File
slf4j-api-1.6.4.jar	1/2/2013 3:32 PM	Executable Jar File
stringtemplate-3.2.1.jar	1/2/2013 3:32 PM	Executable Jar File
xml-apis-1.3.04.jar	1/2/2013 3:32 PM	Executable Jar File
xmlpull-1.1.3.1.jar	1/2/2013 3:32 PM	Executable Jar File
xpp3_min-1.1.4c.jar	1/2/2013 3:32 PM	Executable Jar File
xstream-1.4.1.jar	1/2/2013 3:32 PM	Executable Jar File

iii) **Configure** Drools Eclipse Workbench

Windows → Preferences → Drools → Installed Drools Runtime → Add

(a) Add the version of Drools (since you are allowed to have more than one installed) that you want to add to the build path of newly created Drools projects (the example below shows an older version, you should use the current version of Drools that you have just installed).

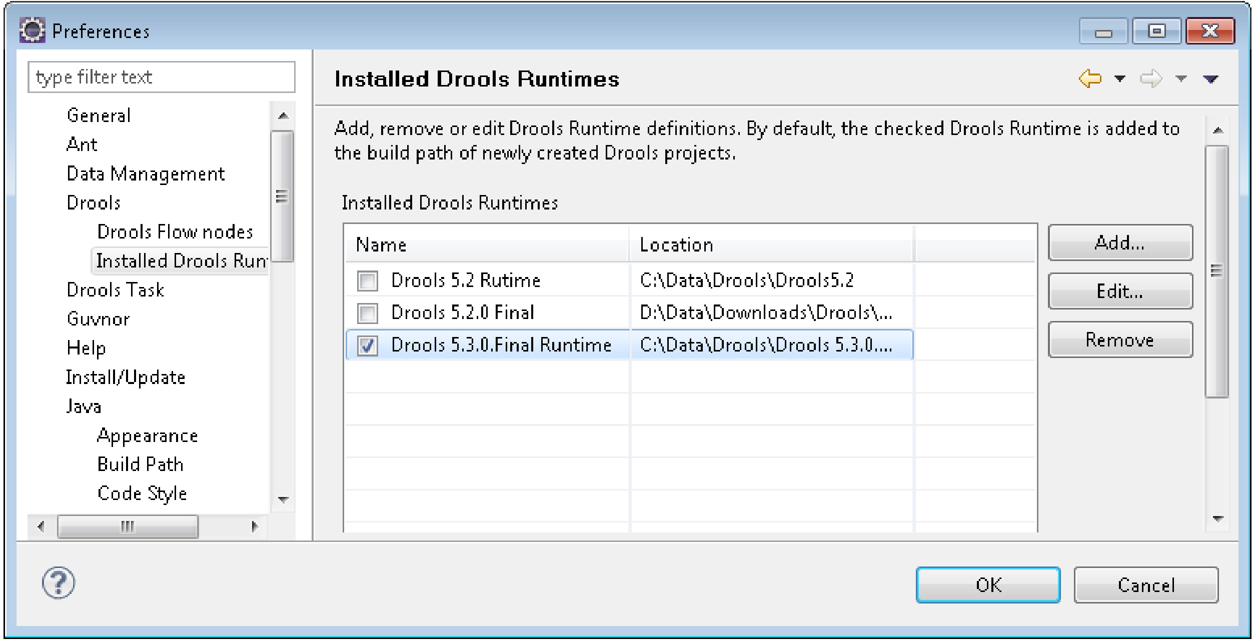


Either select an existing Drools Runtime on your file system or create a new one.

Name: Drools 5.3.0.Final Runtime

Path: C:\Data\Drools\Drools 5.3.0.Final

(2) Select the Drools runtime for use (Use the current version, not 5.3 as shown below)



Installed Drools Runtimes

Add, remove or edit Drools Runtime definitions. By default, the checked Drools Runtime is added to the build path of newly created Drools projects.

Installed Drools Runtimes

Name	Location
□ Drools 5.2 Rutime	C:\Data\Drools\Drools5.2
□ Drools 5.2.0 Final	D:\Data\Downloads\Drools\...
☑ Drools 5.3.0.Final Runtime	C:\Data\Drools\Drools 5.3.0....

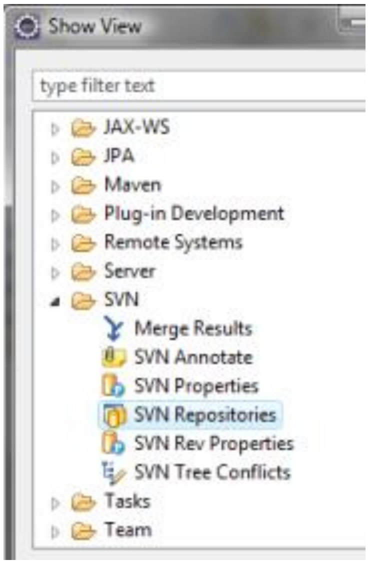
(3) Restart Eclipse

II. OpenCDS Configuration in Eclipse

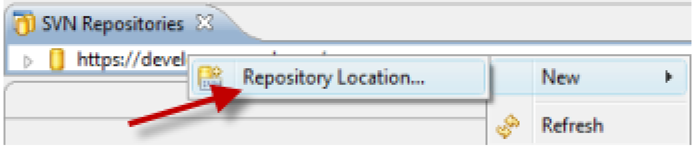
All OpenCDS sub-projects are contained within a parent project called "**opencds-parent**". An overview of these sub-projects will be made available in a separate document in the near future.

A. Import Projects into Eclipse

1) **Add the OpenCDS repository** to the list of repositories available to Eclipse.
   a) Window → Show View → Other…→ SVN → SVN Repositories



   b) You will see a list of the configured Subversion repositories (your initial list may be empty). Right-click on the list and choose New → Repository Location from the pop-up dialog



   c) Add the repository for the OpenCDS Trunk Project. The relevant URLs are as follows. Most individuals should initially connect to the 1.x release repository.
      i) E.g., enter http://develop.opencds.org/svn_v1.0/trunk/opencds-parent, and click on Finish

Repository	URL and Description	Access
Main development trunk	http://develop.opencds.org/svn_active/trunk/opencds-parent Locus of the active development on 1.x releases.	Currently restricted to active code contributors. Access may be requested from David Shields at david.shields@opencds.org.
1.x release	http://develop.opencds.org/svn_v1.0/trunk/opencds-parent Stable snapshots from the main development trunk.	Public read-only access. ID and password not required.

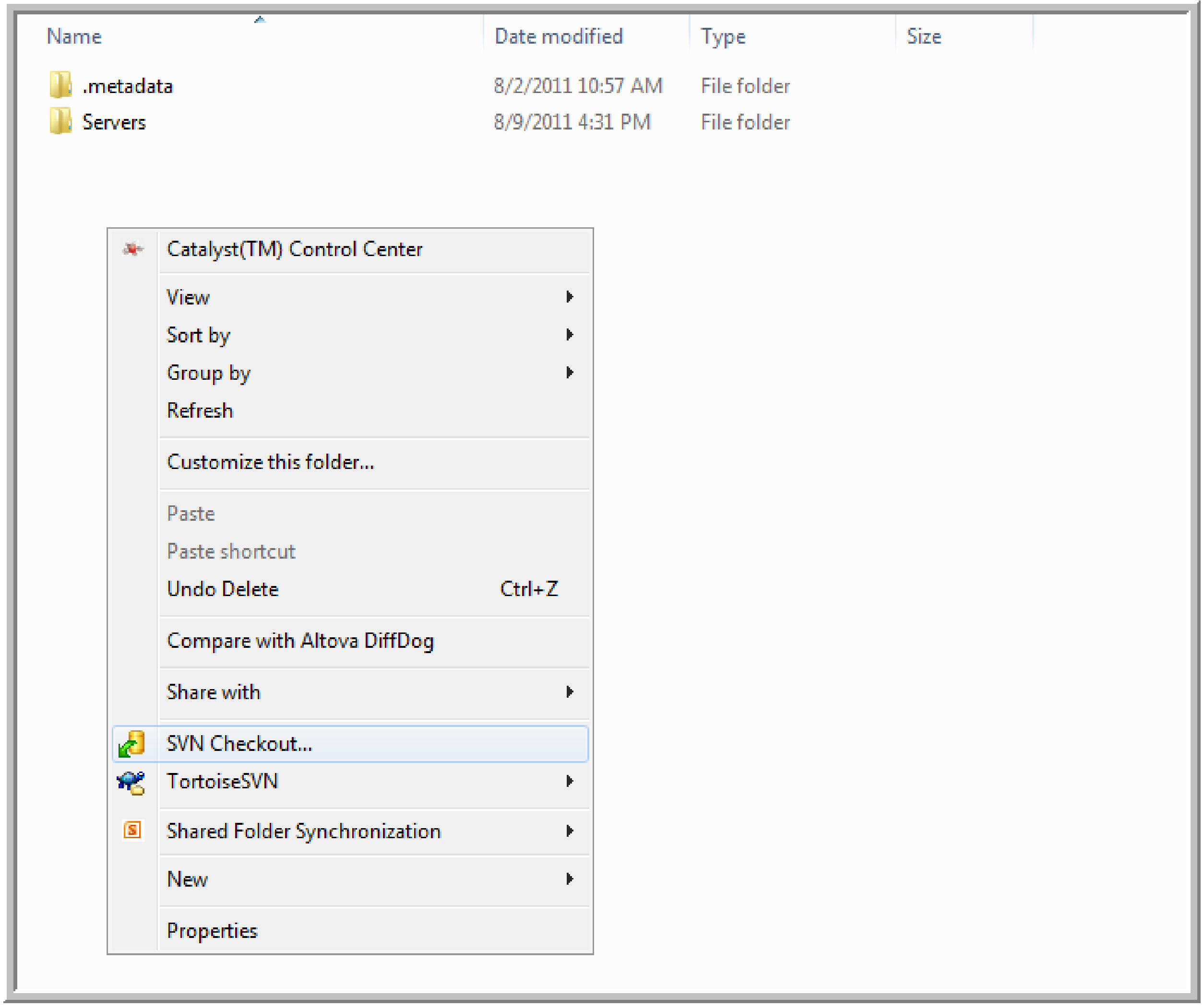
2) **Import the OpenCDS Project Step 1: Establish Maven Projects**

   a) Close Eclipse if open

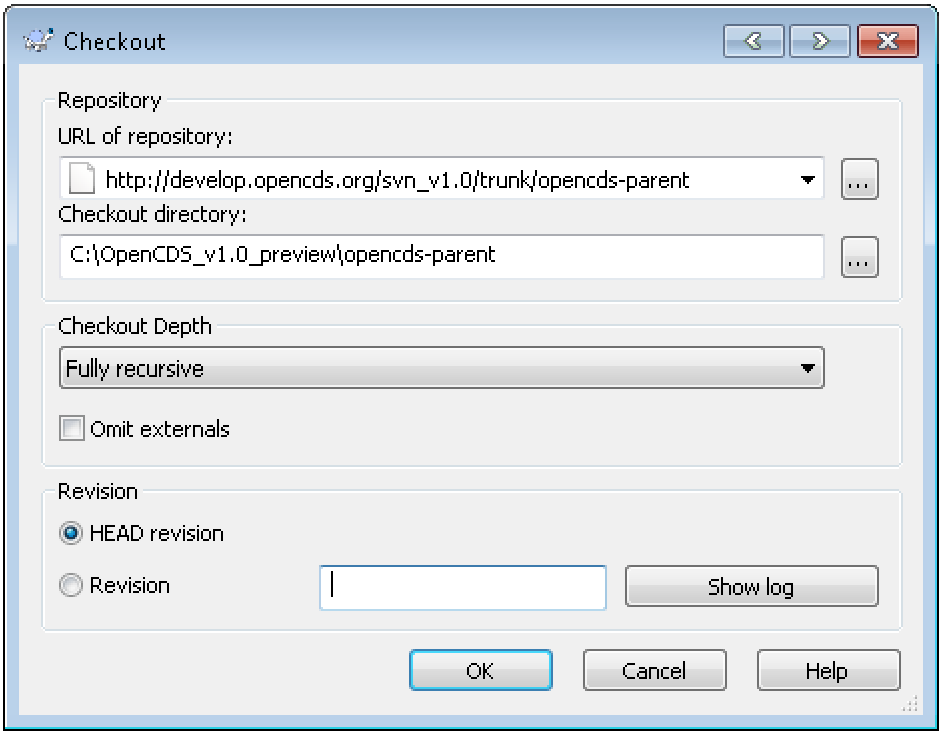
   b) Check out OpenCDS parent project into the Eclipse OpenCDS workspace as follows:

      i) Go to workspace location (e.g., `C:\OpenCDS`)

      ii) Right click and select SVN Checkout from context menu



      iii) Enter URL of repository and proceed, as noted below. Note that the URL of the repository should be the appropriate one noted in the table above. Also note that you must add "`opencds-parent`" to the end of the Checkout directory.



Repository

URL of repository: http://develop.opencds.org/svn_v1.0/trunk/opencds-parent

Checkout directory: C:\OpenCDS_v1.0_preview\opencds-parent

Checkout Depth: Fully recursive

□ Omit externals

Revision

● HEAD revision

○ Revision

iv) Now, build the project with Maven from the command line

(1) From the command line, go to the opencds-parent folder in the workpace (e.g., `C:\OpenCDS\opencds-parent`)

(2) Run the command

```
(a) mvn clean
```

(3) Run the command

```
(a) mvn install
```

(4) Run the command

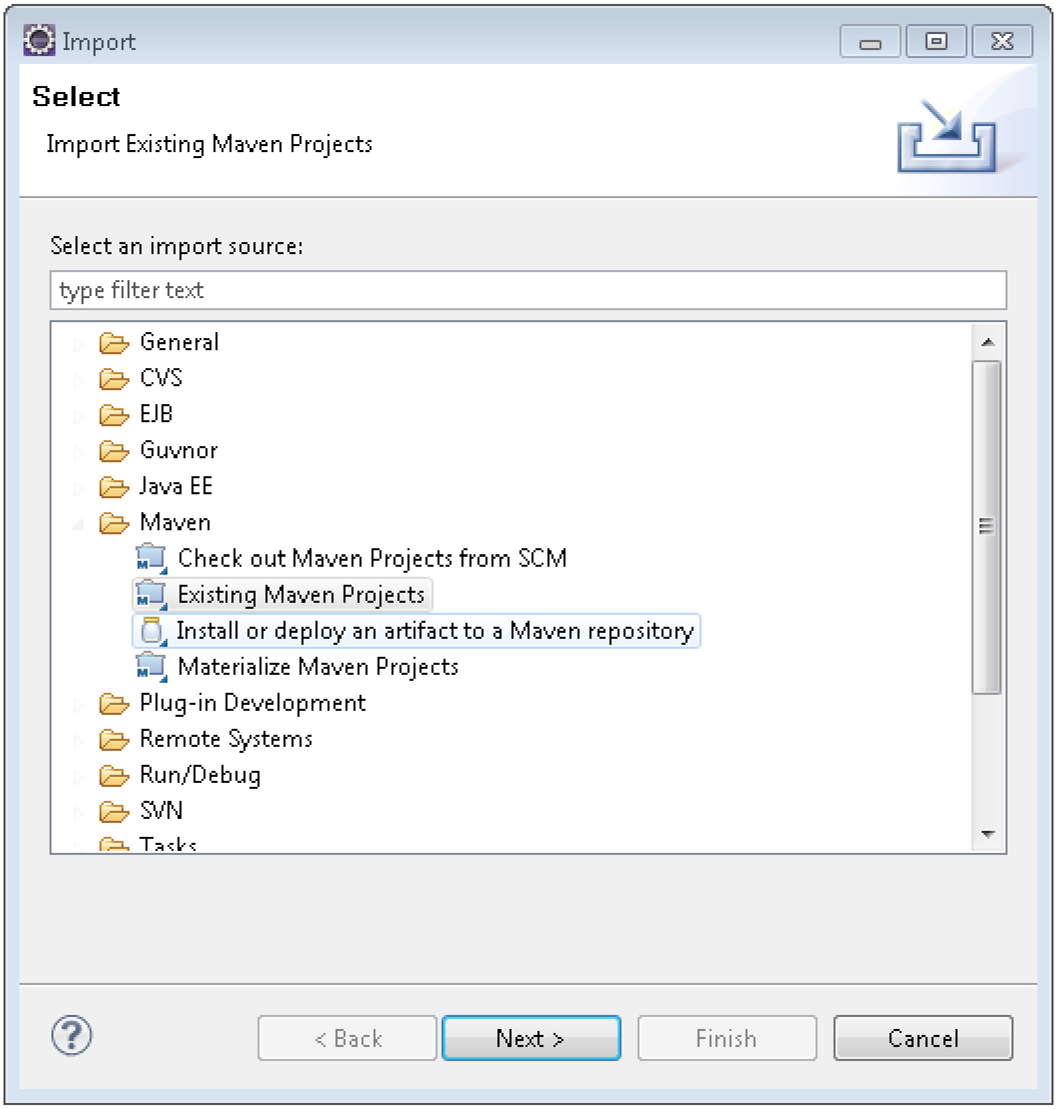
```
(a) mvn eclipse:eclipse
(b) running this command creates all the Eclipse-specific project settings for
    OpenCDS so that you can work with it in Eclipse.
```

3) **Import the OpenCDS Project Step 2: Import Maven Projects**

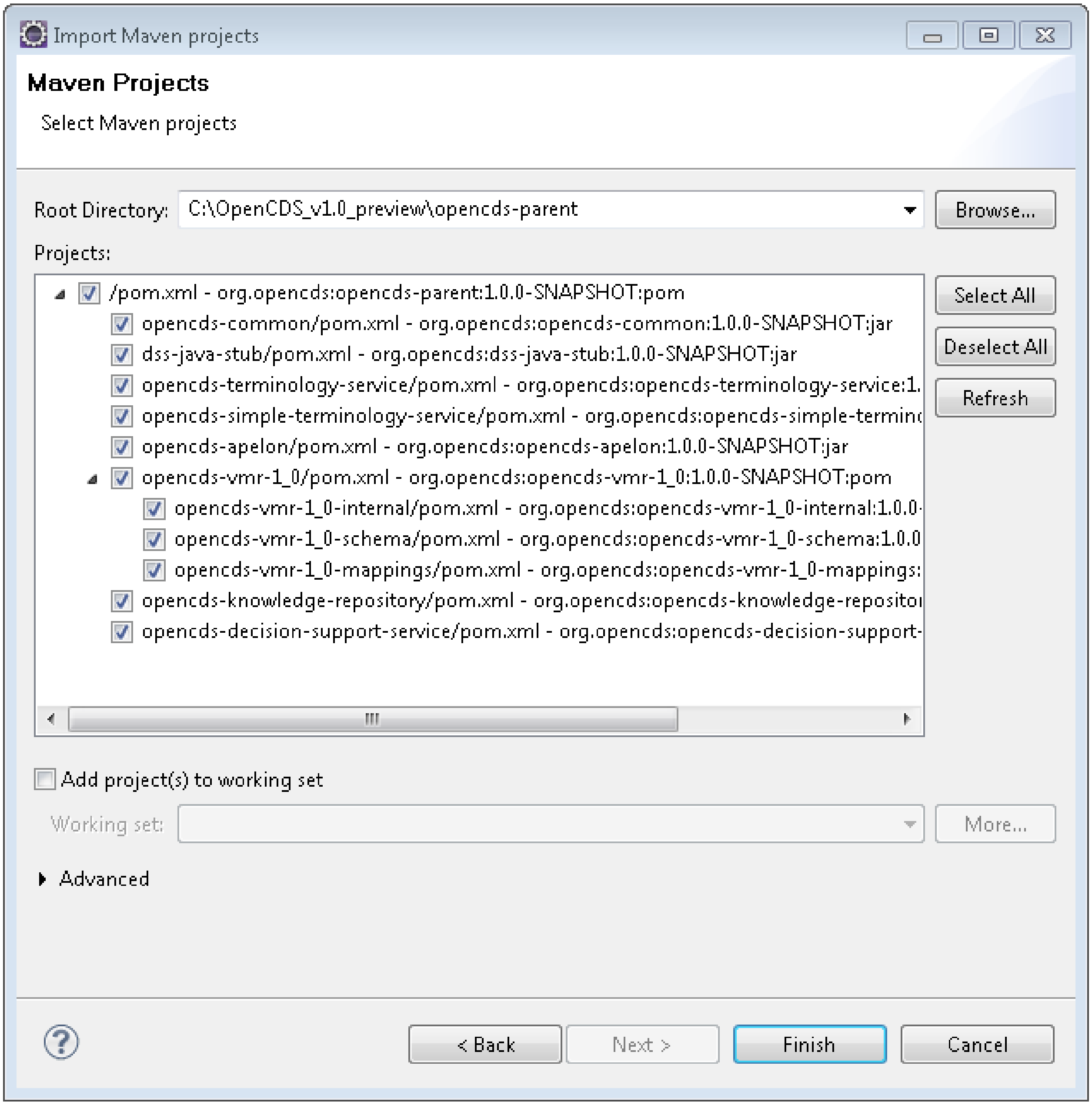
a) Open Eclipse

b) Import the OpenCDS parent project into Eclipse OpenCDS workspace as a Maven project, as follows:

i) File → Import → Maven → Existing Maven Projects → Next



ii) Root Directory → Browser → opencds-parent location (e.g., `C:\OpenCDS\opencds-parent`). Select all Projects, Finish (the latest version of OpenCDS may have slightly different modules than those shown in the example below)



Maven Projects

Select Maven projects

Root Directory: C:\OpenCDS_v1.0_preview\opencds-parent

Projects:

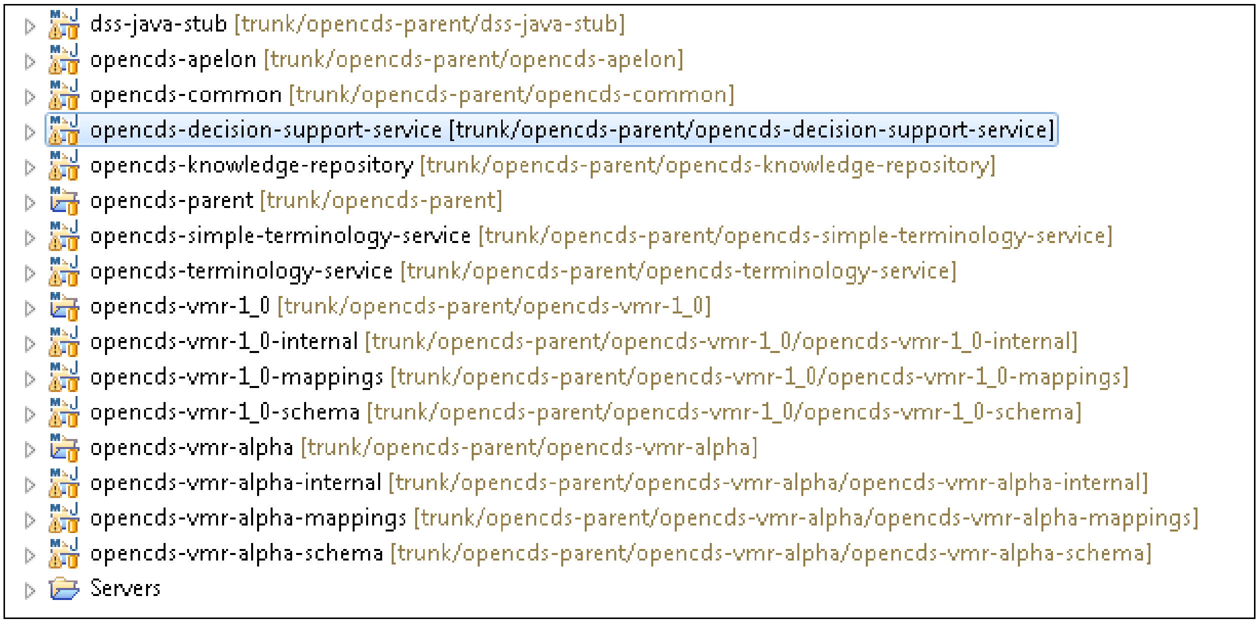
- ☑ /pom.xml - org.opencds:opencds-parent:1.0.0-SNAPSHOT:pom
  - ☑ opencds-common/pom.xml - org.opencds:opencds-common:1.0.0-SNAPSHOT:jar
  - ☑ dss-java-stub/pom.xml - org.opencds:dss-java-stub:1.0.0-SNAPSHOT:jar
  - ☑ opencds-terminology-service/pom.xml - org.opencds:opencds-terminology-service:1.
  - ☑ opencds-simple-terminology-service/pom.xml - org.opencds:opencds-simple-terminc
  - ☑ opencds-apelon/pom.xml - org.opencds:opencds-apelon:1.0.0-SNAPSHOT:jar
  - ☑ opencds-vmr-1_0/pom.xml - org.opencds:opencds-vmr-1_0:1.0.0-SNAPSHOT:pom
    - ☑ opencds-vmr-1_0-internal/pom.xml - org.opencds:opencds-vmr-1_0-internal:1.0.0-
    - ☑ opencds-vmr-1_0-schema/pom.xml - org.opencds:opencds-vmr-1_0-schema:1.0.0
    - ☑ opencds-vmr-1_0-mappings/pom.xml - org.opencds:opencds-vmr-1_0-mappings:
  - ☑ opencds-knowledge-repository/pom.xml - org.opencds:opencds-knowledge-reposito
  - ☑ opencds-decision-support-service/pom.xml - org.opencds:opencds-decision-support-

☐ Add project(s) to working set

Working set:

B. Build OpenCDS Eclipse Project

1) Following the process above, you should see something similar to the following in Eclipse as built projects (there are more modules as time goes on, and some get renamed as well):

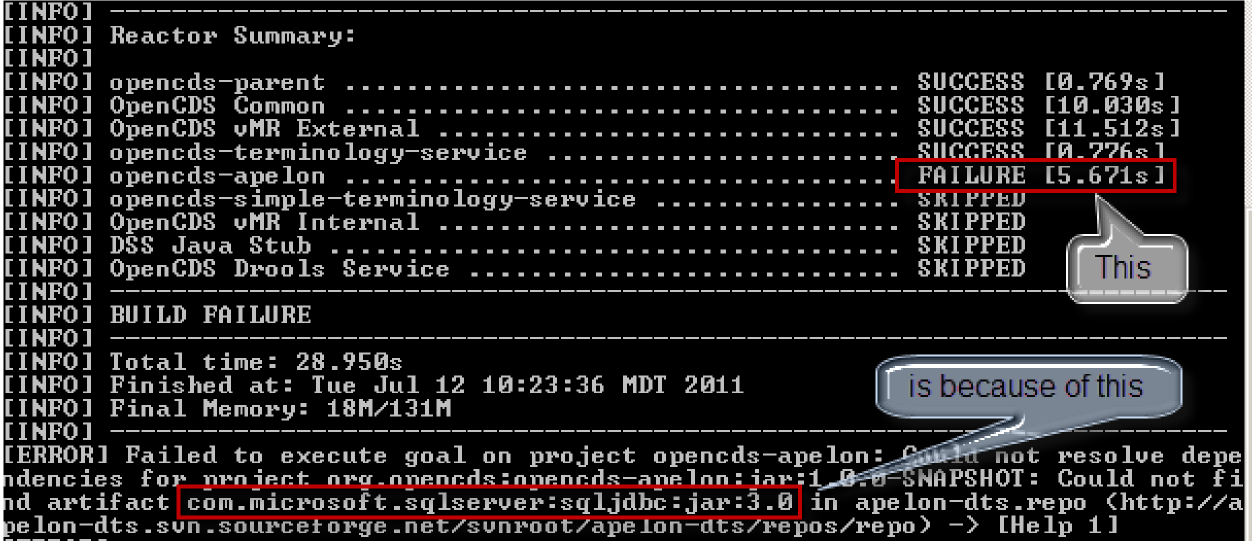


```
dss-java-stub [trunk/opencds-parent/dss-java-stub]
opencds-apelon [trunk/opencds-parent/opencds-apelon]
opencds-common [trunk/opencds-parent/opencds-common]
opencds-decision-support-service [trunk/opencds-parent/opencds-decision-support-service]
opencds-knowledge-repository [trunk/opencds-parent/opencds-knowledge-repository]
opencds-parent [trunk/opencds-parent]
opencds-simple-terminology-service [trunk/opencds-parent/opencds-simple-terminology-service]
opencds-terminology-service [trunk/opencds-parent/opencds-terminology-service]
opencds-vmr-1_0 [trunk/opencds-parent/opencds-vmr-1_0]
opencds-vmr-1_0-internal [trunk/opencds-parent/opencds-vmr-1_0/opencds-vmr-1_0-internal]
opencds-vmr-1_0-mappings [trunk/opencds-parent/opencds-vmr-1_0/opencds-vmr-1_0-mappings]
opencds-vmr-1_0-schema [trunk/opencds-parent/opencds-vmr-1_0/opencds-vmr-1_0-schema]
opencds-vmr-alpha [trunk/opencds-parent/opencds-vmr-alpha]
opencds-vmr-alpha-internal [trunk/opencds-parent/opencds-vmr-alpha/opencds-vmr-alpha-internal]
opencds-vmr-alpha-mappings [trunk/opencds-parent/opencds-vmr-alpha/opencds-vmr-alpha-mappings]
opencds-vmr-alpha-schema [trunk/opencds-parent/opencds-vmr-alpha/opencds-vmr-alpha-schema]
Servers
```

   a. A build has already taken place from the command line as a part of the setup process. For future builds, you can do through the following procedure:
      i. Right click on **opencds-parent** → Run as → Maven Clean
      ii. Right click on **opencds-parent** → Run as → Maven Install

C. Troubleshooting

Maven is great when things are working. However, it can be challenging to get things to work when Maven is not working. The most common problem will look something like this:



```
[INFO] ------------------------------------------------------------------------
[INFO] Reactor Summary:
[INFO]
[INFO] opencds-parent .................................... SUCCESS [0.769s]
[INFO] OpenCDS Common .................................... SUCCESS [10.030s]
[INFO] OpenCDS vMR External .............................. SUCCESS [11.512s]
[INFO] opencds-terminology-service ....................... SUCCESS [0.776s]
[INFO] opencds-apelon .................................... FAILURE [5.671s]
[INFO] opencds-simple-terminology-service ................ SKIPPED
[INFO] OpenCDS vMR Internal .............................. SKIPPED
[INFO] DSS Java Stub ..................................... SKIPPED
[INFO] OpenCDS Drools Service ............................ SKIPPED
[INFO] ------------------------------------------------------------------------
[INFO] BUILD FAILURE
[INFO] ------------------------------------------------------------------------
[INFO] Total time: 28.950s
[INFO] Finished at: Tue Jul 12 10:23:36 MDT 2011
[INFO] Final Memory: 18M/131M
[INFO] ------------------------------------------------------------------------
[ERROR] Failed to execute goal on project opencds-apelon: Could not resolve depe
ndencies for project org.opencds:opencds-apelon:jar:1.0.0-SNAPSHOT: Could not fi
nd artifact com.microsoft.sqlserver:sqljdbc:jar:3.0 in apelon-dts.repo (http://a
pelon-dts.svn.sourceforge.net/svnroot/apelon-dts/repos/repo) -> [Help 1]
```

Here are a couple suggestions to fix the problem:

1. Check the results of the Maven Installs

Maven will sometimes report a successful installation when the specified file (the "-Dfile=…" parameter) does not exist. This will then cause downstream failures of Maven builds. Verify that the local Maven repository contains the jar files you installed by looking in your local Maven repository (found as a folder named ".m2" in your "User" directory). For example, this path for the SpringSource jar will be:

```
.m2\repository\org\apache\jasper\springsource\com.springsource.org.apache.jasper.org.eclipse.jdt.springsource\6.0.20.S2-r5956\com.springsource.org.apache.jasper.org.eclipse.jdt.springsource-6.0.20.S2-r5956.jar
```

If you find that one of the jars that you installed with a "`mvn install …`" command is missing, then rerun the installation, making sure that you have used the correct path for the source file, and spelled everything correctly. The most common error is to download the files to the default download path on your machine (eg, "C:\yourUserName\Downloads"), but to install it using the sample command that we furnished which has a different path. Another common problem is that the downloaded package might have a "`.zip`" extension, and needs to be changed to a "`.jar`" extension before use.

2. The nuclear option

If you cannot get things to work using the above approach (usually due to a Maven issue), you can reset your system to the state just prior to the Eclipse installation (section I.F) via the following:

- Delete Eclipse folder (e.g., `C:\Program Files\eclipse`)
- Delete OpenCDS workspace (e.g., `C:\OpenCDS`)
- Delete Maven repository (e.g., `C:\Users\Ken\.m2`)

You can then re-attempt the installation and configuration process.