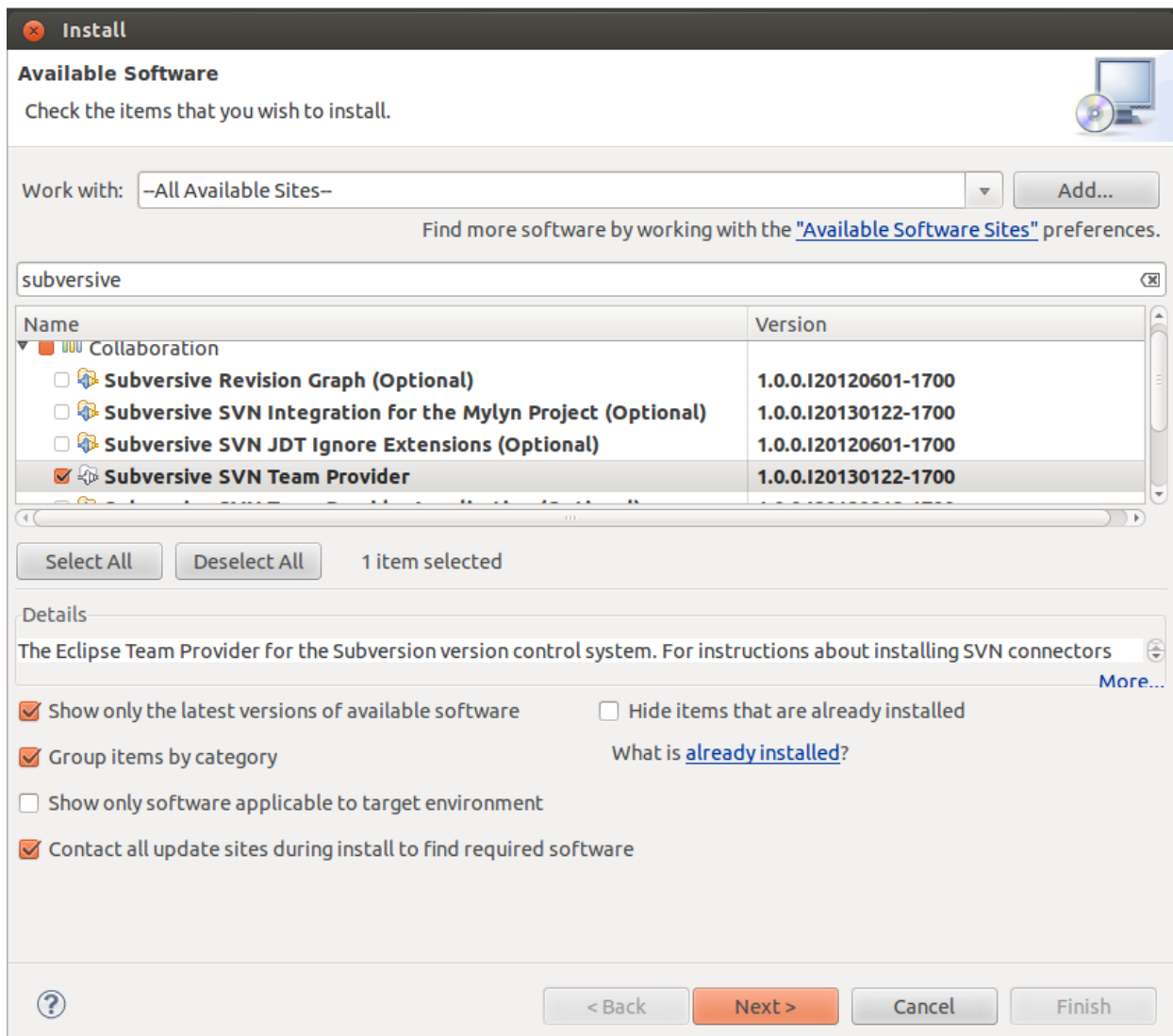


LSim in Eclipse

1. First of all, you need to have the Subversion plugin for Eclipse. If you don't have it: Go to “Help” → ”Install new software”. Then, search “Subversive” and check “Subversive SVN Team Provider”. Click on “Next”, accept the terms and click on “Finish”. You will need to restart Eclipse.



2. Create a new repository location with the URL `svn://eimtserv.uoc.edu/LSim/trunk` and your user and password.

New Repository Location

Enter Repository Location Information

Define the SVN repository location information. You can specify additional settings for proxy and svn+ssh, https connections.

General | Advanced | SSH Settings | SSL Settings

URL: Browse...

Label

Use the repository URL as the label

Use a custom label:

Authentication

User:

Password:

Save authentication (could trigger secure storage login)

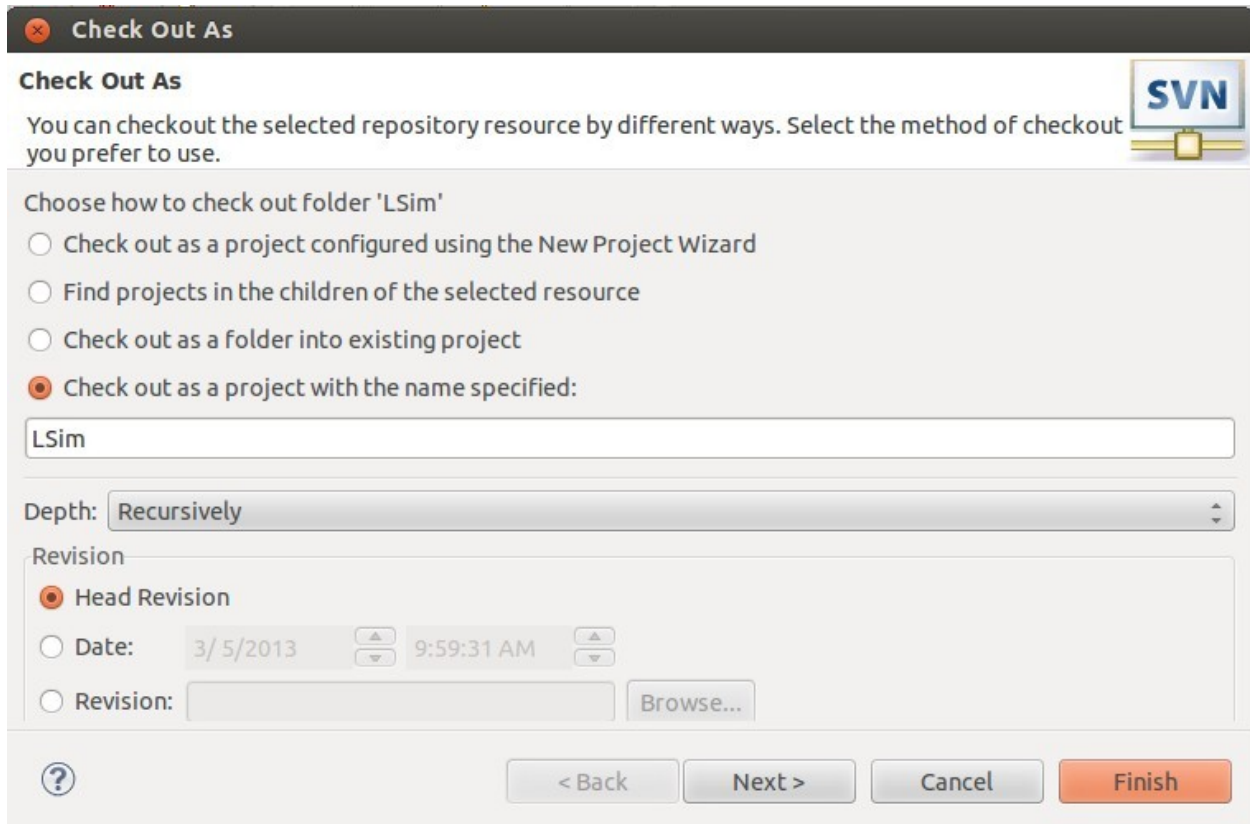
To manage your security data, please see ["Secure Storage"](#)

Show Credentials For:

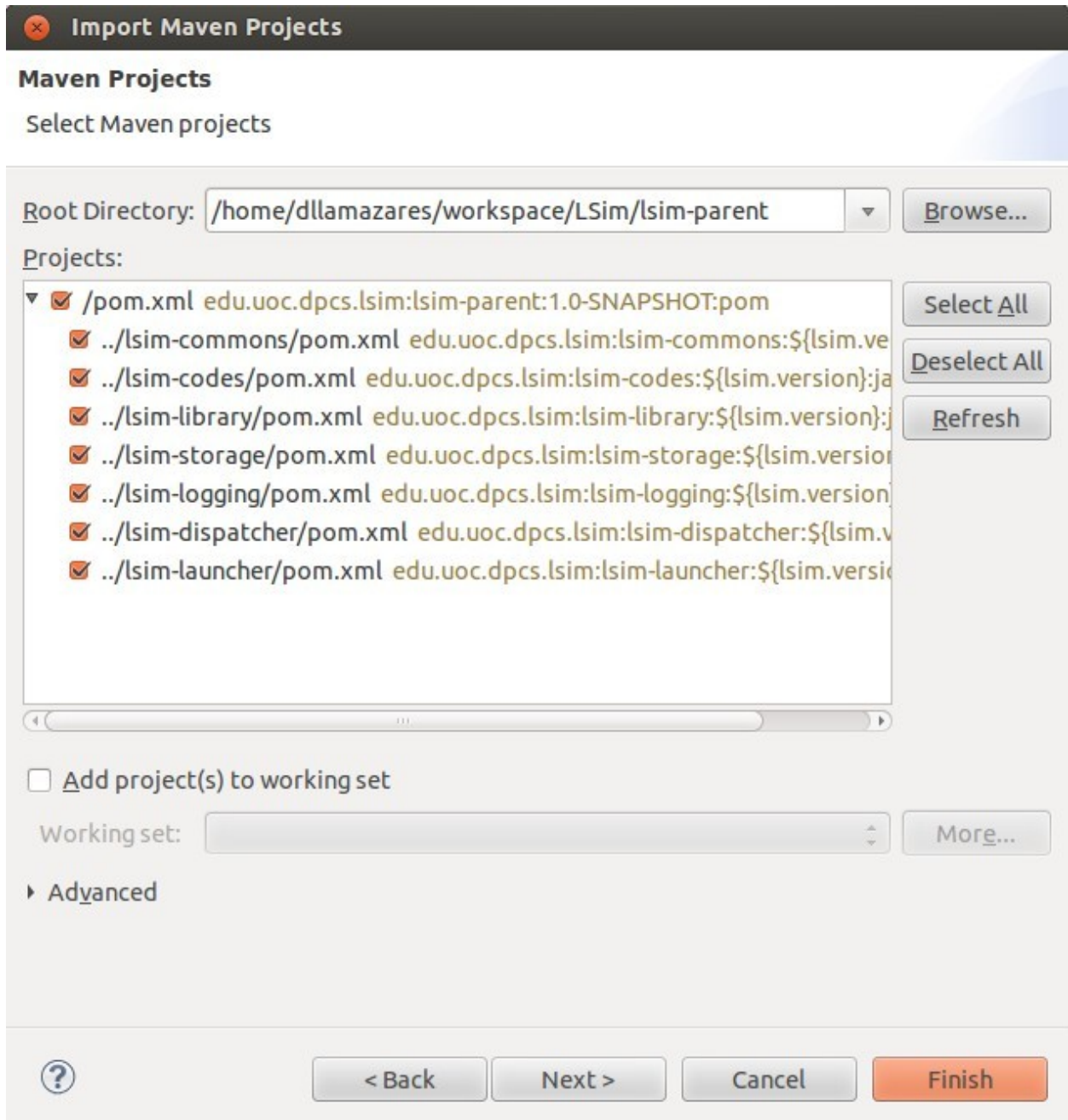
Validate Repository Location on finish Reset Changes

? Cancel Finish

3. Right click on the repository created and do “Find/Check Out As...”. Select “Check out as a project with the name specified:” and click on “Finish”.



4. After doing the checkout, it's time to import the maven projects. Go to “File” → ”Import” → ”Existing Maven Projects”. Then, browse the directory of “lsim-parent” in “LSim” folder. Now, select all projects and click “Finish”.



5. To compile all projects with maven: Right click on “lsim-parent” → ”Run As” → ”Maven install”. It is also possible to compile with the Console, doing “mvn install” in the “lsim-parent” directory. Anyway, you should get something like this:



```
<terminated> /usr/lib/jvm/jdk1.7.0_07/bin/java (05/03/2013 10:23:37)
[INFO] lsim-parent ..... SUCCESS [0.200s]
[INFO] lsim-launcher ..... SUCCESS [0.251s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 4.487s
[INFO] Finished at: Tue Mar 05 10:23:43 CET 2013
[INFO] Final Memory: 11M/103M
[INFO] -----
```

Maven

LSim projects have maven dependencies and sometimes it is needed to update one of the libraries used. For example, imagine that we have made some changes in the source code corresponding to the library `runenvironment-1.0.jar` and we have generated a new version of this library (a new JAR), called `runenvironment-2.0.jar`.

How to incorporate this new library to the project LSim? It's easy: First, the new library has to be uploaded to the maven repository, following the maven structure. Then, the `pom.xml` file of every project with dependencies to the library has to be edited indicating the desired version of it.

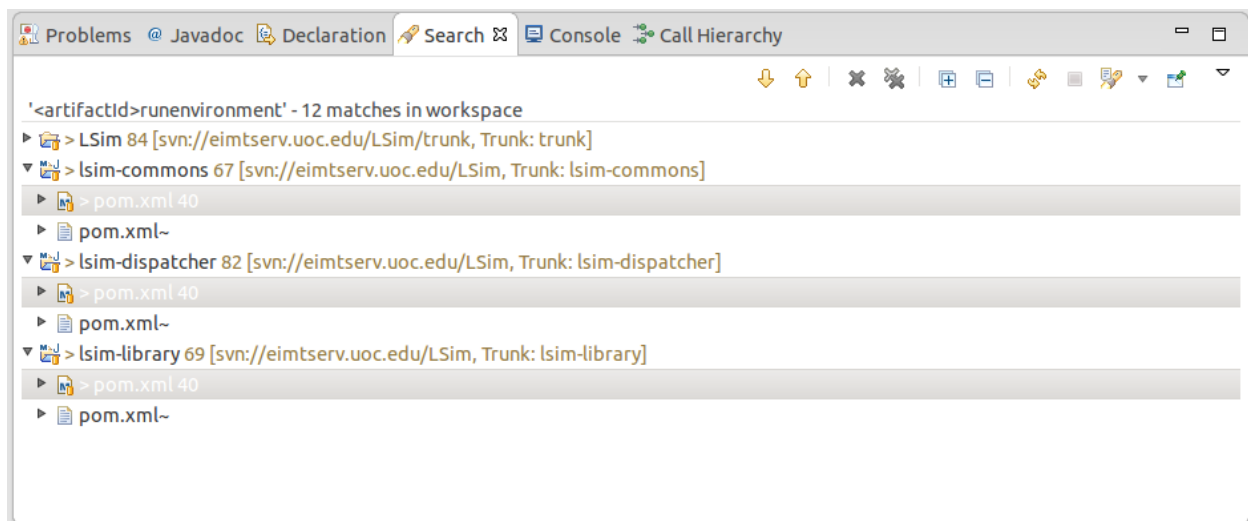
The maven repository is located at `dpcscodes.uoc.edu`, in the directory `"/var/www/maven"`. So, it is only needed to upload the new files to the server. For example:

```
scp -P 55055 ./runenvironment-2.0.jar dpcscodes.uoc.edu:/home/dllamazares
```

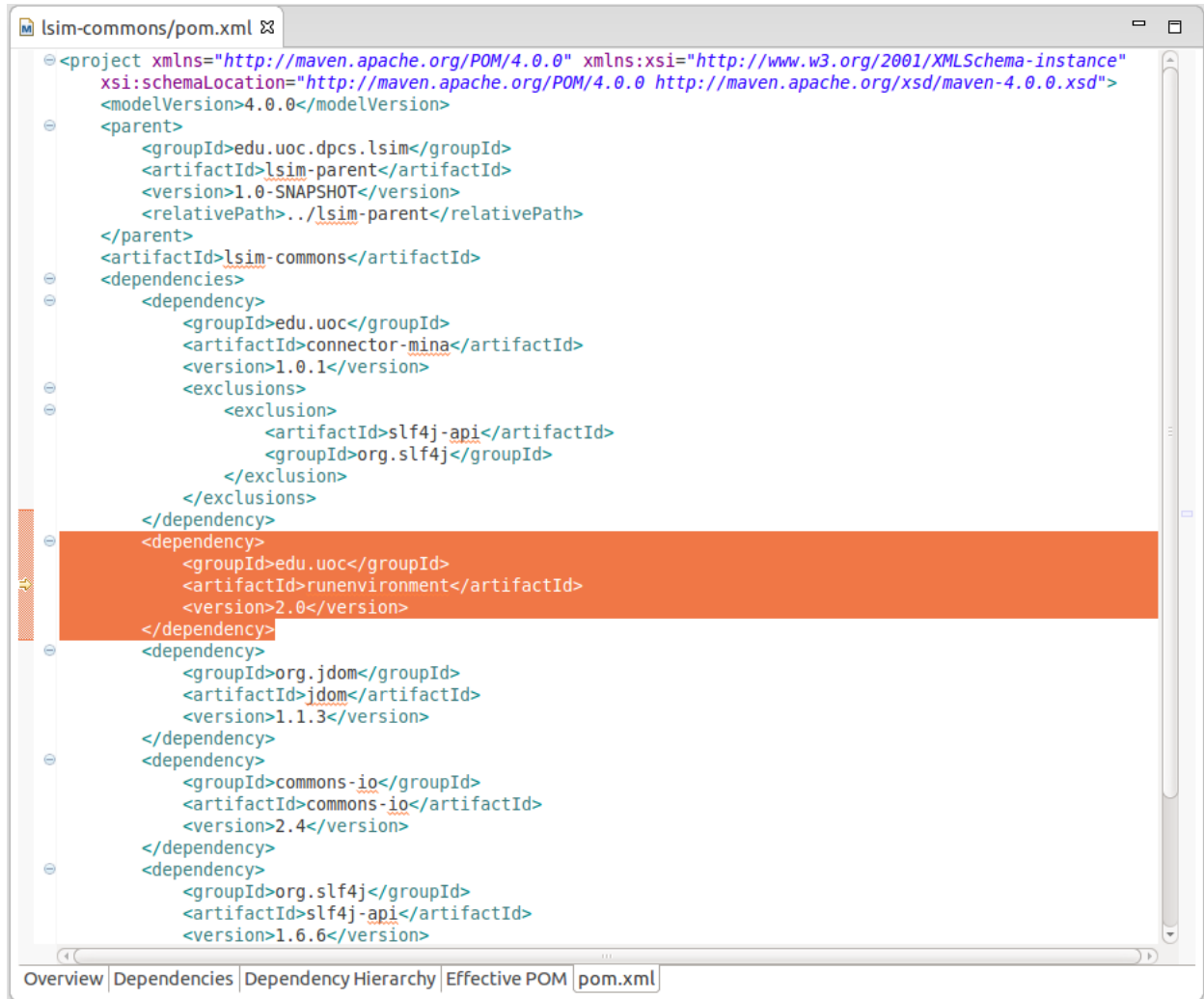
Then, move the files to the right directory. Continuing with our example:

```
ssh -p 55055 dpcscodes.uoc.edu
sudo cp runenvironment-2.0.jar /var/www/maven/edu/uoc/runenvironment/2.0/
```

Once the repository have the new library, it's time to edit the `pom.xml` files. A good way to find these projects is going to the Eclipse top menu and clicking “Search” → “Search”. Then select the left tab (“File Search”) and search “`<artifactId>runenvironment`”. The `pom.xml` files to be edited will be shown. In our example:



For every `pom.xml`, open it and change the “Overview” tab to the “pom.xml” tab. Then, look for the dependency that has to be edited and change the version of it. In our example, we have changed the version from 1.0 to 2.0:



```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>edu.uoc.dpcs.lsim</groupId>
    <artifactId>lsim-parent</artifactId>
    <version>1.0-SNAPSHOT</version>
    <relativePath>../lsim-parent</relativePath>
  </parent>
  <artifactId>lsim-commons</artifactId>
  <dependencies>
    <dependency>
      <groupId>edu.uoc</groupId>
      <artifactId>connector-mina</artifactId>
      <version>1.0.1</version>
      <exclusions>
        <exclusion>
          <artifactId>slf4j-api</artifactId>
          <groupId>org.slf4j</groupId>
        </exclusion>
      </exclusions>
    </dependency>
    <dependency>
      <groupId>edu.uoc</groupId>
      <artifactId>runenvironment</artifactId>
      <version>2.0</version>
    </dependency>
    <dependency>
      <groupId>org.jdom</groupId>
      <artifactId>jdom</artifactId>
      <version>1.1.3</version>
    </dependency>
    <dependency>
      <groupId>commons-io</groupId>
      <artifactId>commons-io</artifactId>
      <version>2.4</version>
    </dependency>
    <dependency>
      <groupId>org.slf4j</groupId>
      <artifactId>slf4j-api</artifactId>
      <version>1.6.6</version>
    </dependency>
  </dependencies>
</project>
```

That's all. For more information about maven, you can check this tutorial:
<http://maven.apache.org/guides/getting-started/>