

Setting Up An Eclipse Development Environment For Shindig (Java)

This tutorial shows you how to setup an Eclipse development environment to develop, build, and test Shindig.

1. Download and unzip the Eclipse IDE for Java EE Developers from the [Eclipse downloads page](#). At the time that this tutorial was written Eclipse Indigo is the current release.
2. Download and unzip Apache Maven from the [Maven downloads page](#). At the time that this tutorial was written Maven 3.0.3 is the current release.
3. Download and unzip Apache Tomcat from the [Tomcat downloads page](#). At the time that this tutorial was written Tomcat 7.0.23 is the current release.
4. Download and unzip the Eclipse Subversive update site from the [Eclipse Subversive downloads page](#). The version you download depends on the version of Eclipse you have installed. Make sure you download the correct version.
5. Start Eclipse.
6. Select a workspace you would like to use for your development.
7. Once Eclipse has started close the Welcome page.
8. Go to Help -> Install New Software
9. In the Install dialogs "Work With" field select "All Available Sites" from the drop down. It may take a bit to get all the update sites.
10. In the filter field type Maven.
11. Select m2e - Maven Integration For Eclipse. There may be more than one entry for this plugin, just select one and click Next.
12. Select Next on the Install Details dialog.
13. Accept the license agreements and click Finish.
14. After the installation finishes, you will be prompted to restart Eclipse, select Not Now.
15. Go to Help -> Install New Software again.
16. Select Add to add a new update site.
17. Select Local on the Add Repository dialog.
18. Navigate to the Eclipse subversive update site you downloaded and unzipped earlier in step 4 and select Open.
19. Give the update site and name, such as Subversive, and select OK to the Add Repository dialog.
20. Select all the categories in the update site and click Finish. At the time that this tutorial was written there was 3 categories Subversive Integration Plug-ins, Subversive Sources, Subversive SVN Team Provider Plugin.
21. Select Finish on the Install Details dialog.
22. After the install finishes you will be prompted to Restart Eclipse, select Restart Now.
23. After Eclipse restarts you will be prompted to install and SVN Connector. Select the latest SVN Kit Connector and select Finish on the install dialog. At the time that this tutorial was written the latest connector was SVN Kit 1.3.5.
24. At the Install dialog make sure all the plugins are selected and select Next.
25. Select Next to the Install Details dialog.
26. Accept the licenses, and click Finish.
27. If you get any security warnings, click OK.
28. Once the installation finishes you will be prompted to restart Eclipse, click Restart Now.
29. Once Eclipse restarts close the Welcome view.
30. Open Eclipse Preferences (On Mac Eclipse -> Preferences, on Windows Help -> Preferences)
31. Go to Maven -> Installations and select Add.
32. Browse to the Apache Maven installation you unzipped in step 2 and select Open.
33. This should add an external Maven installation, make sure it is selected and press Apply.
34. In the preferences dialog navigate to Maven -> Discovery and select Open Catalog.
35. In the m2e Marketplace dialog's Find field type "Subversive", and select m2e - subversive SCM Handler and click Finish.
36. On the Install dialog make sure Maven SCM Handler for Subversive is selected and click Next.
37. On the Install Details dialog select Next.
38. Accept the license agreements and select Finish.
39. If you get any security warnings, click OK.
40. The installation will probably happen in the background, but after it is done a restart dialog will pop up, select Yes to restart Eclipse.
41. Once Eclipse restarts go to Window -> Open Perspective -> Other... -> SVN Repository Exploring and select OK.
42. In the SVN Repositories View select the New Repository Location icon.
43. In the New Repository Location dialog enter the URL to the Shindig SVN repository. You can find the URL [here](http://svn.apache.org/repos/asf/shindig). At the time that this tutorial was written the URL is <http://svn.apache.org/repos/asf/shindig>. (Use https if you are a Shindig committer.)
44. Once the repository is added you should see it listed in the SVN Repositories view. Expand the Shindig SVN Repository and right click on trunk. Select Checkout as Maven Project.
45. In the Checkout as Maven project from SCM dialog select Finish.
46. Trunk will be checked out in the background. While that is happening go to Window -> Open Perspective -> Java.
47. After trunk is checked out an Import Maven Projects dialog should popup. Make sure all the projects are selected and select Next
48. On the Setup Maven plugin connectors dialog select Finish. (There may be some errors on this dialog, like yuicompressor, this is OK)
49. An Install dialog will popup to install more m2e Connectors, select all the plugins in the dialog and select Next.
50. On the Install Details dialog select Next.
51. Accept the licenses and select Finish
52. If you get a Security Warning dialog, select OK.
53. After the installation finishes you will be prompted to restart Eclipse, click Yes.
54. After the clean finishes right click on shindig-project and select Run As -> Maven Build...
55. On the resulting screen type "clean install" into the *Goals* field and "all,reporting" into the *Profiles* field. This will mimic the builds that happen [via Jenkins](#) on builds.apache.org. For more information on building with Maven, please see the [Maven documentation](#).
56. After the build finishes there will be a web app in <workspace dir>/shindig-project/java/server/target/shindig-server-2.5.0-SNAPSHOT
57. In Eclipse go to Window -> Show View -> Other... -> Server -> Servers.
58. Right click in the Servers view and select New -> Server.
59. Expand Apache and select Tomcat v7.0 Server. Select Next.
60. Click Browser and navigate to the directory you extracted the Apache Tomcat zip to in step 3 and select Open. Click Finish.
61. Double click on the Tomcat server in the servers view and select the Modules tab.
62. Click the Add External Web Module... button.
63. In the Add Web Module dialog select Browse and navigate to the web app generated from the build in, <workspace dir>/shindig-project/java/server/target/shindig-server-2.5.0-SNAPSHOT and select Open.

64. Click OK to the Add Web Module dialog and save the Web Module changes.
65. Click the Tomcat server in the server view and select the Debug icon.
66. Open a browser and navigate to <http://localhost:8080/samplecontainer/examples/commoncontainer/index.html> and render some gadgets!