Appendix L Importing SOBA onto STS

Some readers are interested in building SOBA on the Spring Tool Suite (STS) IDE instead of a plain Eclipse IDE. This appendix describes the steps to make this happen.

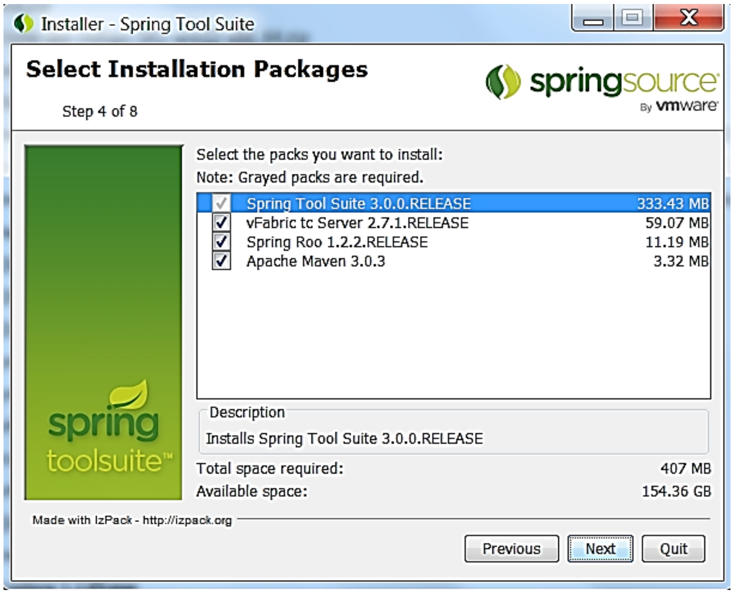
L1. Installing STS

STS can be installed either standalone or as plugins in Eclipse. I prefer a standalone install as the two IDEs (STS standalone and Eclipse with STS plugins) are identical, and I was afraid of messing up my existing Eclipse environment that hosts many of my projects.

To install STS standalone, download the proper version by starting from the link http://www.springsource.org/spring-tool-suite-download, which requires submitting a form with your personal information. I downloaded the installer.exe file for spring-tool-suite-3.0.0.RELEASE for my 64-bit Windows 7 PC. Clicking on this installer.exe would start the installation process as follows:

- Welcome: Click Next
- License: Click Accept and Next
- Target Path: I selected c:\mspc\app\springsource and you can choose your own
- Select Installation Packages (I selected all as shown by default in Figure L.1)
- JDK_PATH: I selected C:\jdk1.6.0_31_64
- … Finish

As is seen from Figure L.1, in addition to STS, the *vFabric tc Server*, *Spring Roo*, and *Maven 3* are installed as well. However, you may not see the `settings.xml` file in the *C:\Users\<login-name>\.m2* folder for Maven 3. Also, the tc Server may include a tomcat 6 and a tomcat 7 folder, but they are not the same as the standalone tomcat 6/7 installs. Here, I assume that you have installed tomcat and Maven 3 standalone, and we are only interested in the STS IDE.


Installer - Spring Tool Suite
springsource
By vmware
spring
toolsuite™
Made with IzPack - http://izpack.org
Previous
Next
Quit

Select Installation Packages

Step 4 of 8

Select the packs you want to install:
Note: Grayed packs are required.

☑	Spring Tool Suite 3.0.0.RELEASE	333.43 MB
☑	vFabric tc Server 2.7.1.RELEASE	59.07 MB
☑	Spring Roo 1.2.2.RELEASE	11.19 MB
☑	Apache Maven 3.0.3	3.32 MB

Description
Installs Spring Tool Suite 3.0.0.RELEASE

Total space required: 407 MB
Available space: 154.36 GB

Figure L.1 STS packs.

L.2 STS VERSUS ECLIPSE IDE

An STS IDE and an Eclipse IDE look similar to each other except that you would see a lot of Spring specific project templates and features as expected. The differences between an STS IDE and an Eclipse IDE are shown in Figures L.2 and L.3. For now, we ignore these differences, as the procedure to import SOBA into either IDE is the same.

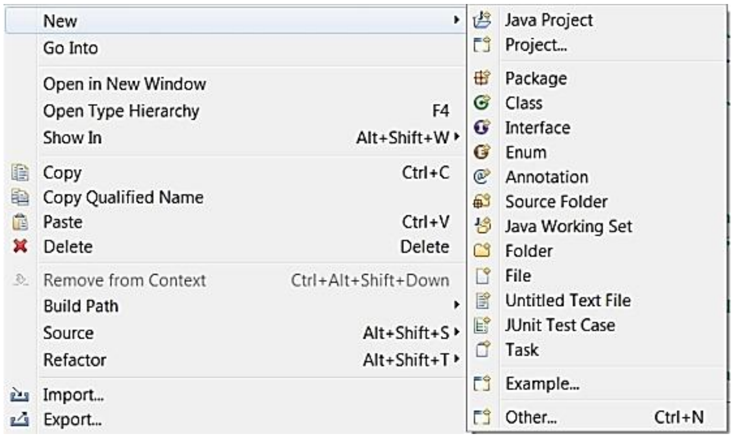


Figure L2. A plain Eclipse IDE.

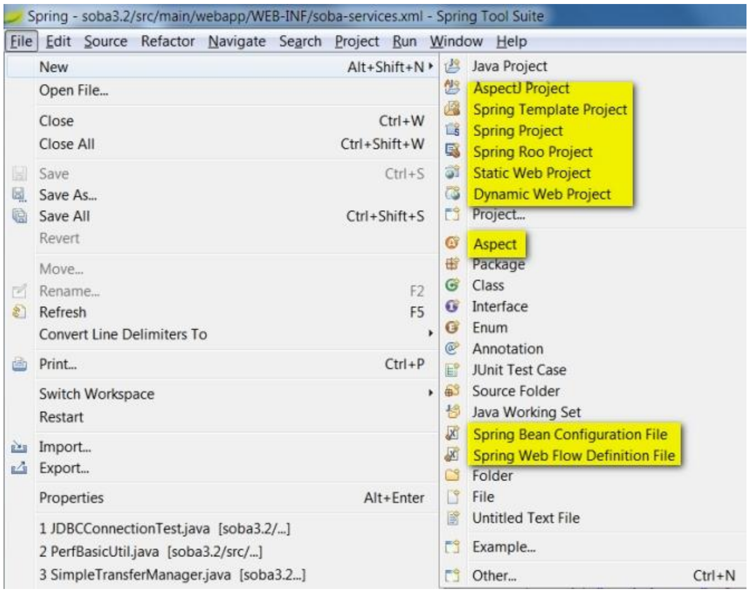


Figure L.3 An STS IDE.

L3. IMPORTING SOBA INTO STS

There really is no difference between importing SOBA into an STS IDE and an Eclipse IDE. So, please perform steps of E.2, E.4, and E.5 in Appendix E to complete importing SOBA into your STS IDE. Then, follow the steps in Appendix I to deploy SOBA straight from Maven to Tomcat 7.

Depending on the version of SOBA you download, you might encounter a few errors after importing SOBA into your STS IDE. These errors appear in `soba-services.xml` if any, complaining "*no setter found for property* …" associated with a few classes. Although these errors do not affect building SOBA successfully with Maven, you can fix them easily by applying the following changes:

- `SimpleAclTxManager.java` in the `service` package: change `setaclBankingTxDao` to `setAclBankingTxDao`
- `SimpleTransferManager.java` in the `service` package: change `setbankingTxDao` to `setBankingTxDao`
- `SimpleTxManager.java` in the `service` package: change `setbankingTxDao` to `setBankingTxDao`
- `PerfBasicUtil.java` in the `util` package
    - change `private static String profilingStatus ;` to `private static String profilingStatus = "disabled";`
    - delete the corresponding `<property>` element in `soba-services.xml` file.
- Do a project "*clean*" and those errors should disappear.

From this point on, everything else is the same whether SOBA is hosted on STS or Eclipse.