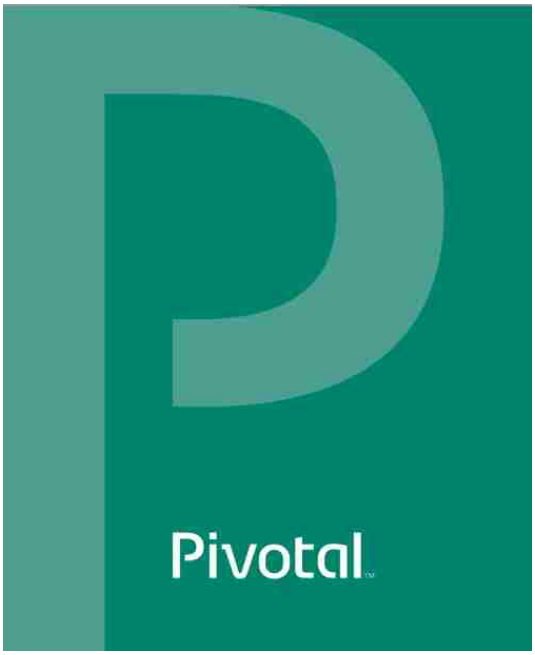

Manual Installation of the Lab Environment

Last review: November 2015

1 Introduction

Welcome to the course. We have tried to make the lab setup as simple as possible. There are some installers that you can use to install everything you need for the course on your file system.

However, the installers only cover the most commonly used platforms/operating systems. If you wish to use a different Operating System or are having problems with our courseware installers, you can also install your lab environment manually. If the approach described by section 4 fails for you, see section 6 for yet another option.

2 Installing SpringSource Tool Suite

Download the latest STS release from http://spring.io/tools (if you already have a recent version it will probably be fine).

Install STS on your computer

You *do not* need to launch it for the moment

3 Copying Resources to Local File System

As part of the resources you've been provided at the beginning of the course, there is a folder called *Manual Installation* or something similar. Copy its content to a local folder on your file system. You will find 3 items there:

1. `com.springsource.training.workspace.builder-1.x.x.jar`
   Custom Eclipse plugin developed by Spring. You should copy this file inside the STS install folder (under the `dropins` subfolder).

2. `<course-name>-XXX.RELEASE-all-platforms.zip`
   Contains all resources for the labs where `XXX` is the current version.

3. This document.θ

On your filesystem, unzip `<course-name>-XXX.RELEASE-all-platforms.zip`. You now can see the following folders:

1. `<course-name>-XXX.RELEASE` contains the projects that are used for the labs and the lab instructions (inside the doc subfolder).

2. `repository` contains all the libraries used in the course (jar files).

3. Any other resources, such as Apache Tomcat, needed by your course.

4 Starting up STS

At this stage, STS should be closed. If you have already launched it, you can restart it so the plugin you just installed will be loaded.

The following examples assume you are using an MS Windows machine, installing Core Spring V3.1.1 onto the `C:` drive. Your platform, course and/or version will be different but the procedure is identical.

This Eclipse plugin makes the installation process easier. Right after launching STS, you will be prompted for a workspace location (any place is fine). You will then see the following screen:

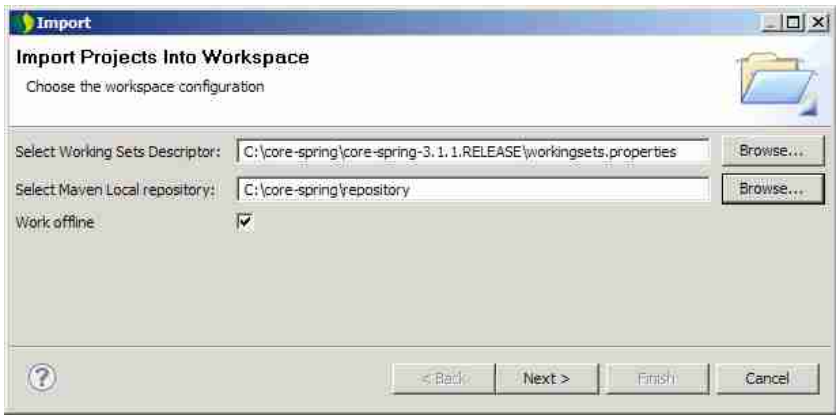

Figure 4.1: Import Projects #1

1. The screenshot assumes that you have extracted the content of `core-spring-XXX.RELEASE-all-platforms.zip` into `C:\core-spring` (this is not the default). You can replace this with your local path. *θ*

2. Click on `Next>`. You now can specify the home folder (wherever you unpacked the zip file) for all your Eclipse project as follows:

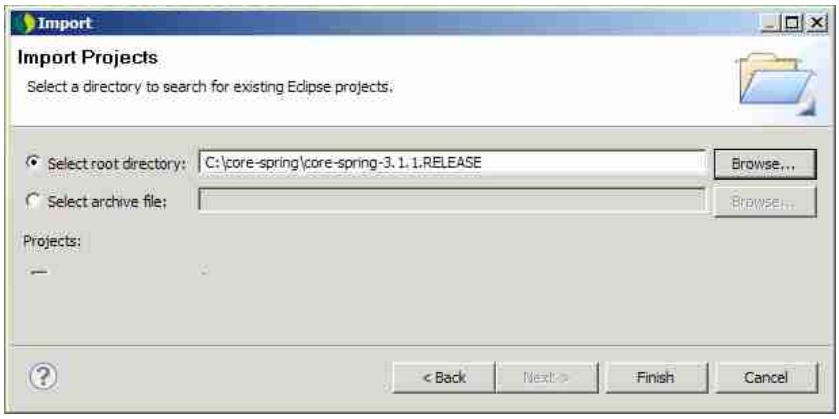

Figure 4.2: Import Projects #2

You can now click on `Finish`.

It takes several minutes to build everything. Eclipse displays "building workspace" at the bottom right corner until all projects have been built. *Please leave it alone* until is has finished. Your lab environment is ready. Congratulations!

If this option fails for you, consider the *maven* option described by section 6.

5 Tomcat or tc Server

Starting from the third day of the course, there are a few labs that require a web environment (in particular the MVC lab). Note that the labs based on Spring Boot use an embedded Tomcat and *do not* require the following setup.

By default, STS does not install with Tomcat, it used *tc Server* instead. Pivotal's *tc Server* (short for Tomcat Server) is built on Tomcat and the labs work well using it.

However, if you wish to work with pure Tomcat, you just need to install it. It was provided in the zip file.

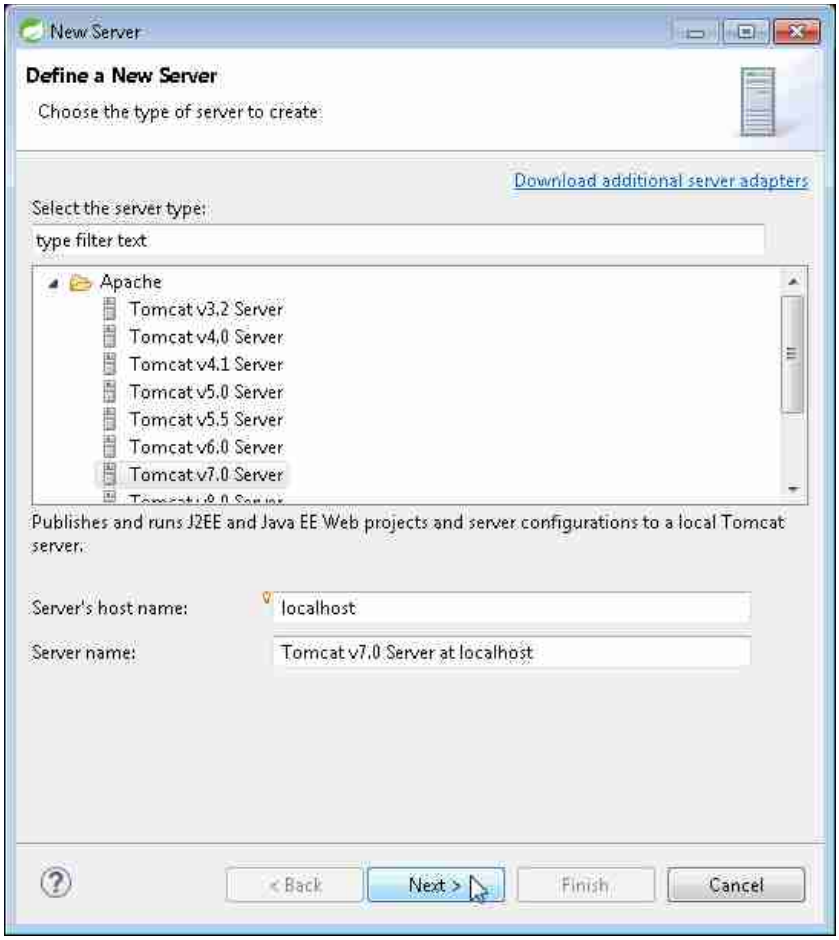

Figure 4.3: Pick the right Tomcat version

- If the *Servers* window is showing (usually it is the bottom-left view in STS) it will already contain a tc Server installation. Right-click in the white area of the view window and select `New -> Server`.

- Alternatively select `Windows -> Preferences -> Servers -> Runtime Environments` then click the `Add` button.

- Open the Apache folder (see figure 4.3) and select the version of Tomcat supplied in the zip file. At the time of writing (November 2015) that is Tomcat V7.

- Click `Next >`

- You will need to tell it where to find the Tomcat installation – wherever you unpacked the zip file. In the examples above, it would be `c:\core-spring.3.1.1.RELEASE\apache-tomcat-XXX` where `XXX` is the Tomcat version (see figure 4.4).

- Click Finish and you are done. The new server should appear in the *Servers* view.

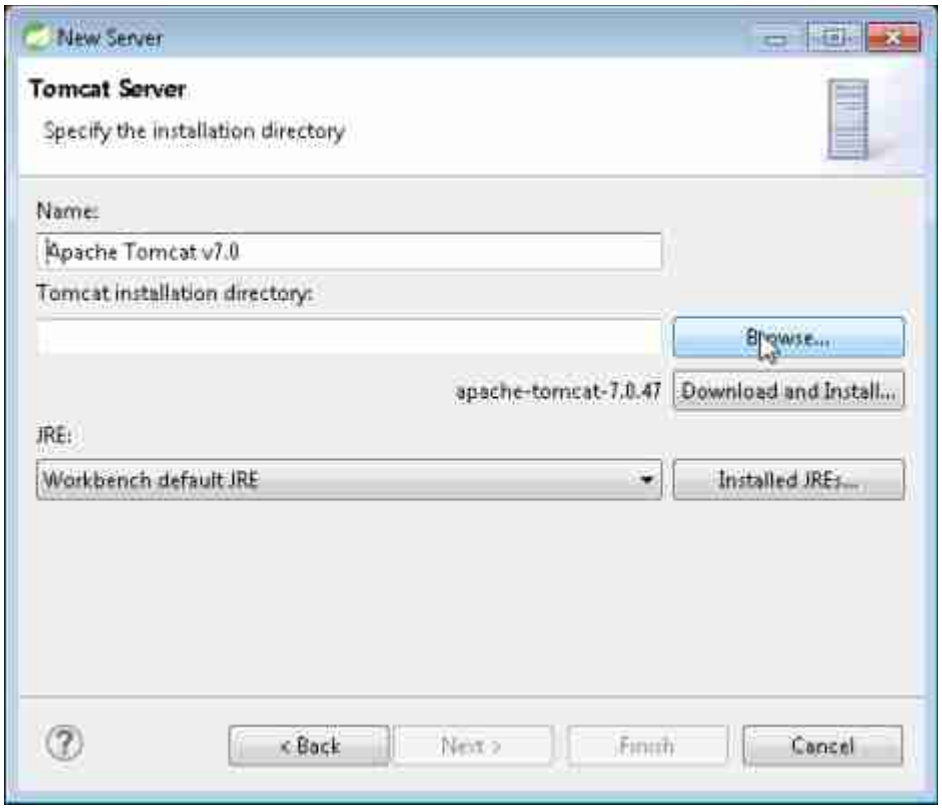

Figure 4.4: Browse for the Tomcat installation

6 Alternative Setup *without* the Plugin

The Eclipse plugin we provide sets up the labs as Eclipse projects in working sets – one working set per lab, numbered in the right order.

Not everyone is successful with this plugin, or you may just prefer to use the labs with Maven. The following is an alternative setup procedure.

6.1 Maven Setup

- If you previously copied the plugin jar (`com.springsource.training.workspace.builder-1.x.x.jar`) to the STS `dropins` folder , quit STS and *delete* the jar now.
- Restart STS. When it prompt for a workspace, select an *empty* directory. *Do not* reuse an existing workspace directory.
- Once STS is running there should be nothing in the left-side panel except possibly a Servers directory.
- Use `File -> Import … -> Maven -> Existing Maven project` and browse to the labs folder. Continuing the example above, this would be the directory:

`C:\core-spring\core-spring-XXX.RELEASE`

- Make sure everything is selected, including the `pom.xml` at the root. Click OK.
- Let all the projects import and build.

6.2 Internet Access

- Maven requires Internet access – make sure you don't have STS in offline mode (check `Windows -> Preferences -> Maven` and ensure the `Offline` check-box is *not* selected).
  - If you use a proxy, you will need to configure STS to use it (select `Windows -> Preferences` again and type Proxy in the search box at the top or go to `General -> Network Connections`).
- If you cannot access the Internet, copy the entire repository directory from the zip file to your `.m2` directory.
  - On MS Windows you may need to open a `CMD` window and create the `.m2` directory manually using `mkdir c:\Users\<user>\.m2` where `<user>` is your login name. Then, using Windows Explorer, drag and drop `C:\core-spring\repository` to `c:\Users\<user>\.m2`
  - On Mac OSX platforms:
    `mkdir ~/.m2` (may exist already)
    `cp -R /Applications/core-spring-XXX.RELEASE/repository ~/.m2`
  - On Linux platforms:
    `mkdir ~/.m2` (may exist already)
    `cp -R ~/core-spring-XXX.RELEASE/repository ~/.m2`

6.3 Notes and Warnings

Using this approach means that working sets are *not* created. Labs will be listed in alphabetical order which does not match the order of the course. Please be careful to do the right lab each time.

Note that labs come in pairs – one to work on and a solution. For example the MVC section of the Core Spring course is represented by two labs: `mvc` and `mvc-solution`.

If you need to also setup a server – see previous section.

7 Using IntelliJ

IntelliJ users can use a similar procedure to import the labs as Maven projects into the IntelliJ IDE.