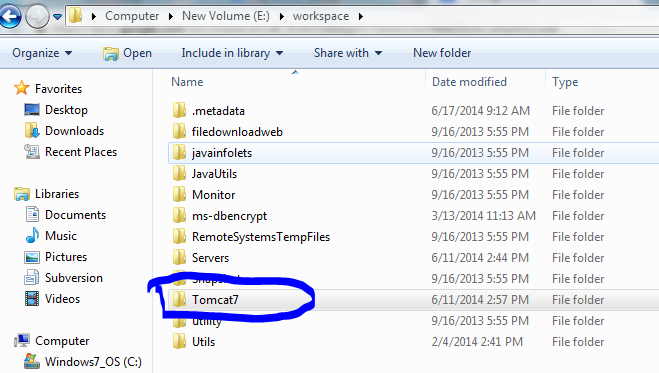
Tired building jar and restart tomcat service on every line of code change?

We used to build the jars (with ant script) and copy into application deployment folder and start the service to reflect the code changes for testing. It is needed even for single line character of code change. For some reasons, restart tomcat would take few minutes and test developer’s patience. To avoid building jars and restart tomcat steps, We need to hot deploy the code changes into running tomcat so that the code changes reflects immediately for testing.

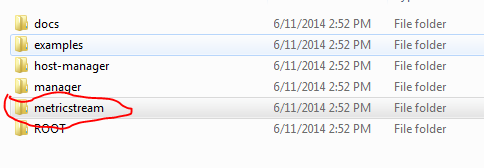
I think it would definitely improve productivity of the developer or at least developer can see the code change results immediately without waiting

Here you go for the steps

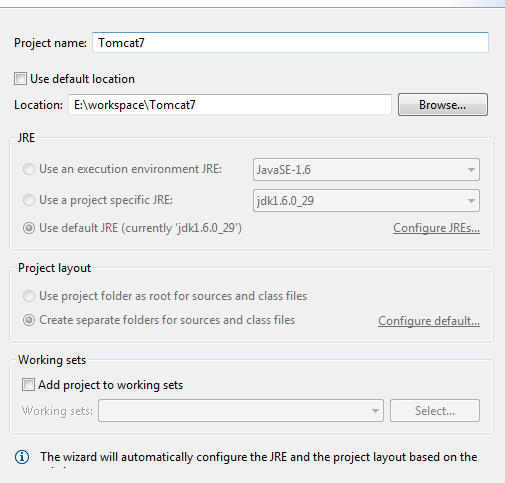
1. Download / copy zipped version of tomcat (I use Tomcat 7.0) and extract to your desired location



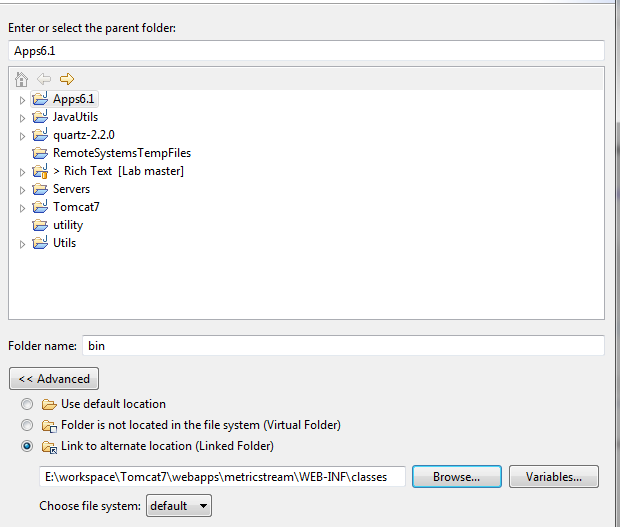
2. Copy “metricstream” application folder (existing) into Tomcat/webapps folder



3. Now create a new Java project in Eclipse and name it of your choice (I named it Tomcat7). But give the project location as Tomcat home (previously created folder)



4. Now we need to have automatically copy build class files(either manually press build all button in Menu bar or set automatically build option) to metricstream/web-inf/classes folder. On metricstream code project which is already created as a project in Eclipse workspace. My trunk base java project name is “Apps6.1”. Right click on the project → New → Folder. Name it as of your choice (I named it as “bin”), select advanced in the wizard, select third option “Link to alternative location” and give the path to Tomcat/webapps/metricstream/web-inf/classes



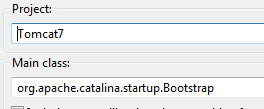
5. Now the tomcat is to be started with in Eclipse so that the compiled classes will be hot deployed by Eclipse. From menu bar, select Run → Debug Configurations. Under configurations, select java application → right click select new and fill following

**Main tab:**

Name: Tomcat7 (your choice)

Project: Tomcat7 (browse your tomcat project previously added

Main class: “org.apache.catalina.startup.Bootstrap”



**Arguments tab:**

VM arguments:

-Xmx1500m

-XX:MaxPermSize=128m

**CLASSPATH tab:**

Add all jars from tomcat/lib and tomcat/bin folders

**Source tab:**

Add your code base java project

Apply the changes and press “Debug”. If everything configured correctly, tomcat will start in eclipse and you should be able to login into application.

**Note: Make sure that all classes(atleast core modules) are compiled in code base project so that class files will be generated to application classes folder**