**Remote Debug a JVM with Eclipse  
Including Tomcat**

**Part A: JVM flag**

Start the JVM with the following flag:

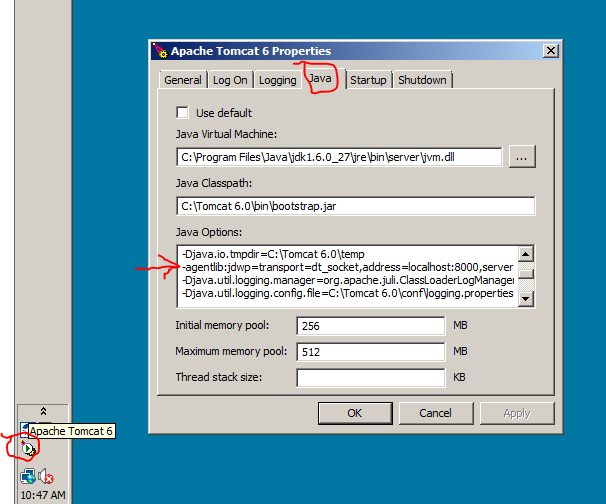
-agentlib:jdwp=transport=dt\_socket,address=localhost:8000,server=y,suspend=n

**Part B: Tomcat JVM flag**

These steps are for Tomcat on Windows, to launch the Tomcat config GUI. Other setups use this [link](http://wiki.apache.org/tomcat/FAQ/Developing).

1. Find Tomcat in the task tray (or launch the “w” executable like Tomcat9w.exe in <tomcat\_home>/bin)
2. Java tab
3. Add flag to Java Options on a new line
4. OK
5. Restart Tomcat

Follow the 2 red circles and the red arrow.

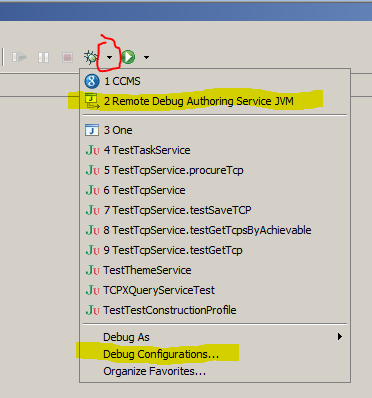


**Part C: Eclipse**

Create a new Debug Configuration which you will launch and use in addition to any other JVMs started in eclipse.

When done, you will launch the remote debugging like this, using a named, favorite configuration. But first open Debug Configurations.

In the picture, the second yellow highlight launches the Debug Configurations window. A completed configuration marked as a Favorite appears like the first yellow highlight.

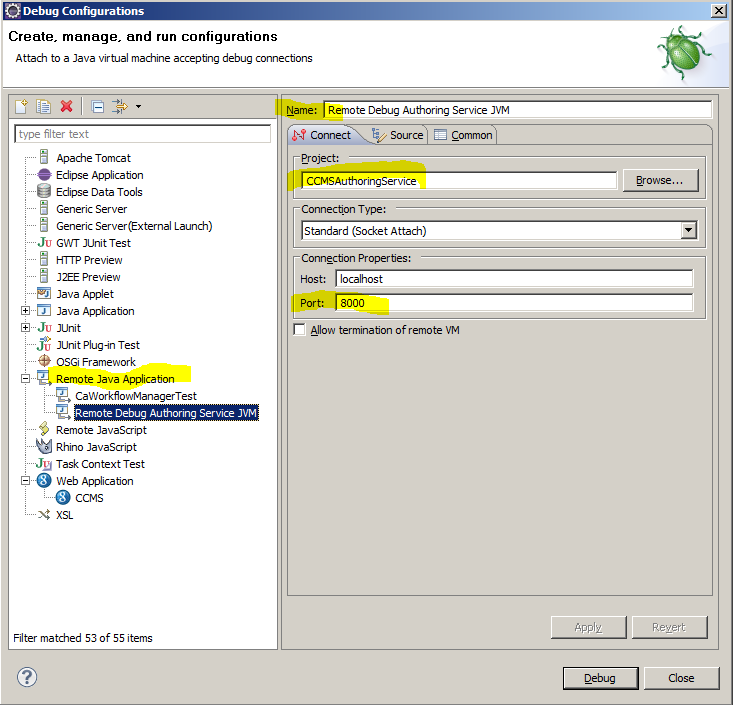


Double-click Remote Java Application to create a new configuration, Give it a good name, and Select the project to watch. The project has the code you want to debug.

Make sure the port listed here matches the port in the JVM entry you added to Tomcat (default is 8000).

Next, on the Debug drop-down, pick Organize Favorites to add the new config to the drop down list.

Now you are setup to debug a remote JVM! Launch the configuration you just created to contact the remote JVM. The debug perspective lists any paused threads that have hit breakpoints.



A running debug configuration called “Debug Docato Tomcat” reveals the threads on the remote JVM.

