

Written by Chris Velado
September 28, 2011

MaintainJ with a J2EE Application

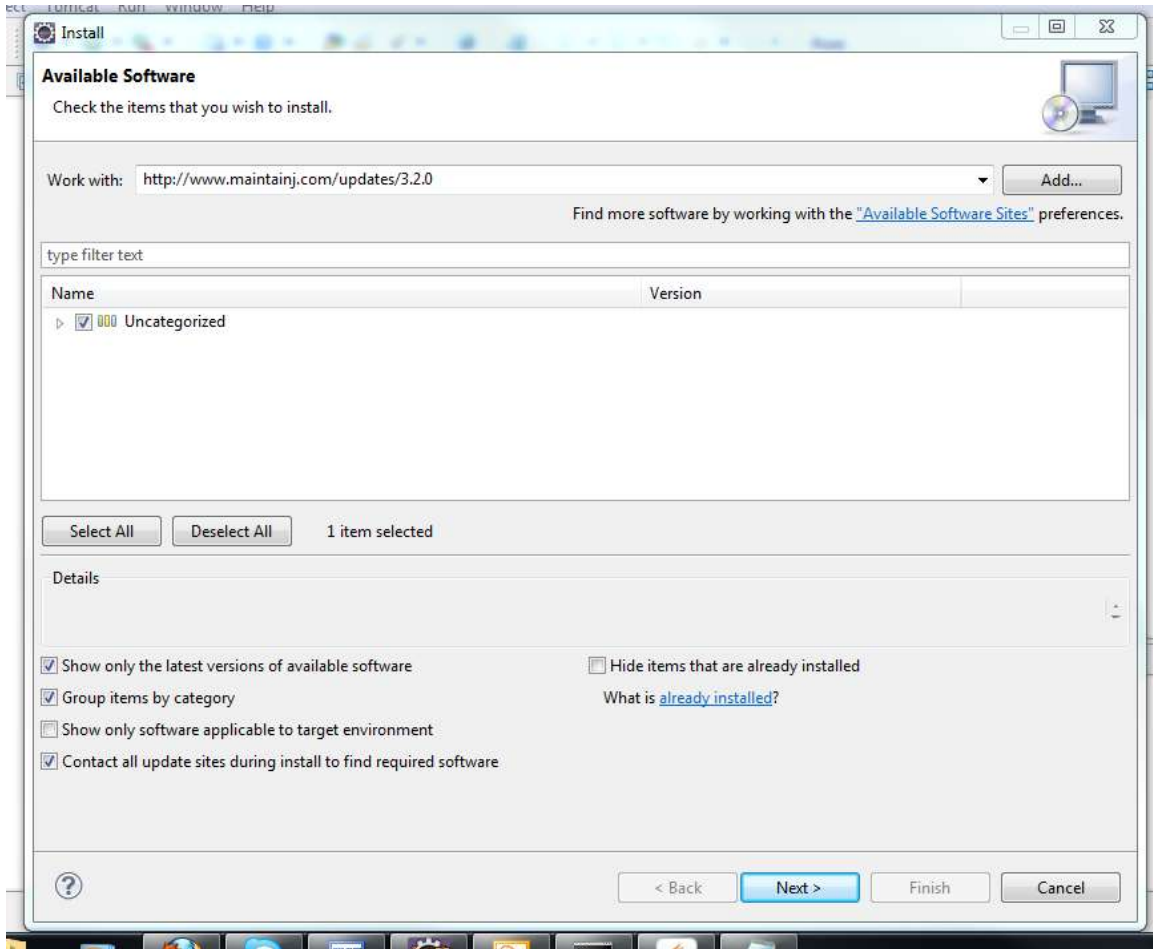
MaintainJ is a Java Application that allows you to view all calls within a J2SE or J2EE application. By allowing their application to perform reverse engineering, MaintainJ collects all the data from the stack trace and generates a .ser file; which then can be rendered in Eclipse and viewed as a sequence diagram.

For ease of setting up your environment, this tutorial will help show you how to integrate MaintainJ with a standard setup of a J2EE Application running on Tomcat 7, thanks to our new friend Choudary who has updated MaintainJ to work with the newest version of Tomcat.

Retrieving and Installing MaintainJ plug-ins (drop-ins) and graphics library

There are two ways of installing the plug-ins into Eclipse, all depending if you are running a standard version of Eclipse or Eclipse Yoxos. If you have a standard copy of Eclipse, you can simply navigate in Eclipse to Help - > Install New Software. There is one repository URL's that you will need to download from.

MaintainJ Features: <http://www.maintainj.com/updates/3.2.0>

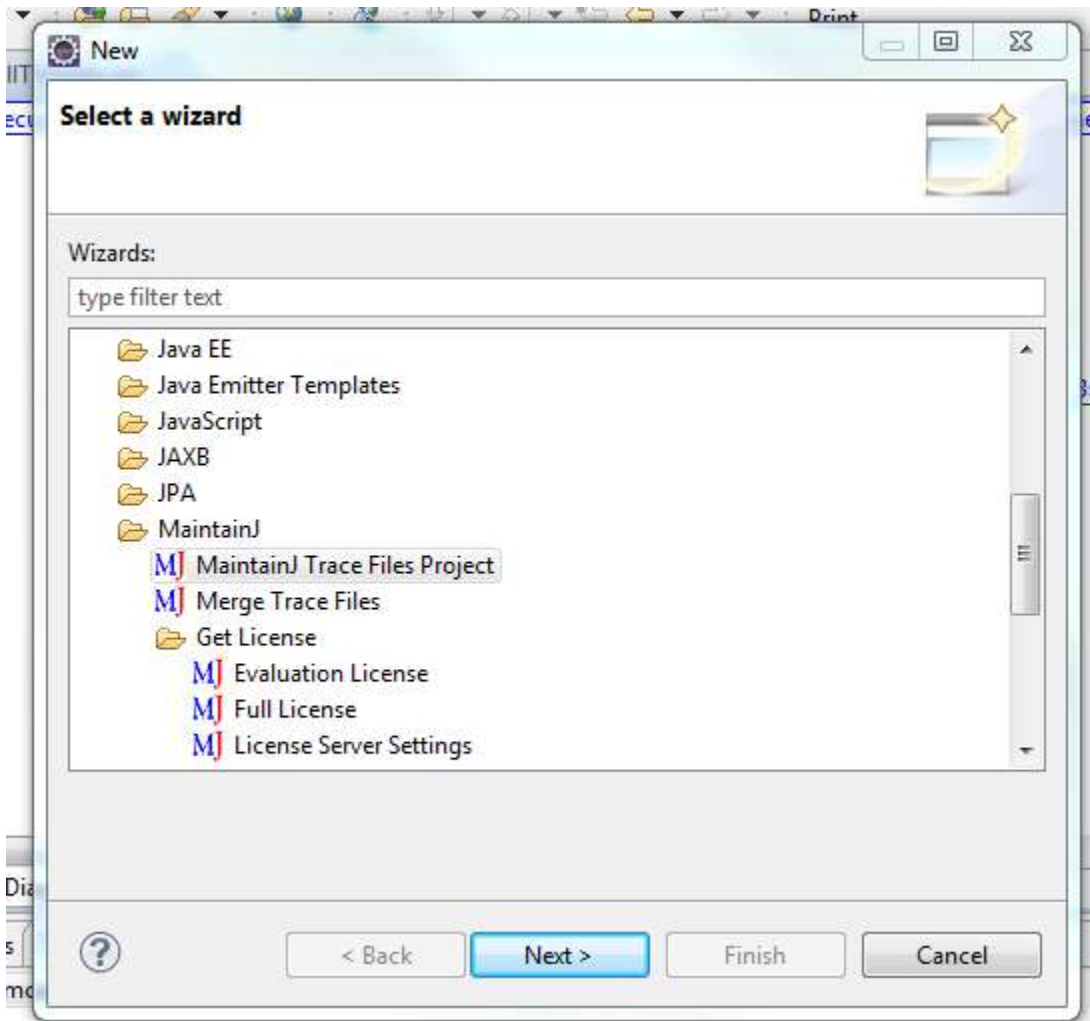


This plug-in should auto detect and install the GEF graphics library for your IDE. For users who are using Eclipse Yokos, click on the link to download the most recent version of the MaintainJ_3_2_Features.zip

<http://maintainj.com/userGuide.jsp?param=install>

Once the file is downloaded and unzipped; copy the content into your Eclipse's plug-in directory (drop-in directory if you are running Eclipse Indigo). Make sure Eclipse is closed and from the command line, run Eclipse -clean to allow your IDE to pick up the new software.

Once Eclipse is up and running, you will want to create a MaintainJ Trace Project. To do so, go to File -> New -> Other and select the MaintainJ Trace Files Project to create the new project. Now that your project is in your workspace, go ahead and jot down the path to the directory of that project; we will need this for the last step.



Running MaintainJ

Now it is time to download and run the MaintainJ application within the same Tomcat 7 server that your J2EE application will be located. Go back to the MaintainJ website and download the MaintainJ.war file and place it under `apache-tomcat-7.0.19\webapps`. Then simply startup Tomcat, assuming that your J2EE application is already configured with this instance. During startup, you should see that the .war file was extracted and placed under `apache-tomcat-7.0.19\work\Catalina\localhost\MaintainJ`.

So the server should now be up and running. Go ahead and check to make sure your J2EE application instance is properly responding. Then open up a new tab in your browser and type the following:

<http://localhost:8080/MaintainJ/>

Install and Configure MaintainJ

The following steps install MaintainJ on your server and instrument your application to generate the call trace files.

Step 1: [Install](#)

This step performs the following tasks.

- a) Copies MaintainJ and AspectJ related jar files to a folder under your server home.
- b) Generates a wrapper start script to your server's start script. Server classpath and JVM settings are updated in this script.

Step 2: [Configure Instrumentation Scope](#)

Specify the package(s) whose classes and sub-packages should be instrumented. Repeat this step whenever you need to restart the server every time after changing the instrumentation scope.

Step 3: Restart Server with the wrapper script generated in Step 1.

Step 4: [Verify Installation and Configuration](#)

Step 5: [Capture Call Trace](#)

You should see the MaintainJ Install and Configure page. Go ahead and navigate through each step. Step 1 will automatically place the needed .jar files for MaintainJ to fully operate. For step 2, go ahead and type the package name(s) within your application. These packages are usually located in your WEB-INF/classes/com directory within the application.

Toward the bottom of the page, you have options as to what type of calls you would like MaintainJ to trace. Once done, click Generate Aspect. Now you should be on Step 3. Without closing the browser tab, go ahead and shutdown the server. You should see a newly created catalina_with_mnj.bat file in the apache-tomcat-7.0.19\bin directory. Go ahead and open to edit this file. Currently while running MaintainJ with your J2EE application, the startup will hit a max perm and memory overload error. To fix this, we will have to specify that max amount of memory (-Xmx2G) and perm size arguments (-XX:MaxPermSize=128M). After editing the 6th line in the catalina_with_mnj.bat file, your copy should now be similar to the following, excluding line 4.

```
echo "***Starting Tomcat with your J2EE Application ***"  
SETLOCAL
```

```
set MJ_LIB=C:\Tomcats\J2EEApp\apache-tomcat-7.0.19\lib  
set AGENTJAR=%MJ_LIB%\aspectjweaver.jar
```

```
set JAVA_OPTS="-javaagent:%AGENTJAR%" -Daj.weaving.verbose=true
%JAVA_OPTS% "-Xmx2G" "-XX:MaxPermSize=128M"
call "catalina.bat" %*
```

ENDLOCAL

Next, go back to the command line and run the following to start up the Tomcat instance again.

```
catalina_with_mnj.bat start
```

After everything has loaded, go back to the MaintainJ tab and click on the Step 4 verification link. If all goes well, you should now be able to capture the call traces. Simply click on Start Tracing. Now take the directory path that we jotted down earlier to the MaintainJ Trace Project, and copy it along with the preferred file name with no extension into the Trace File input box. Next, click Stop Trace and restart by clicking on the Start Tracing button.

Viewing Sequence Diagram.

MaintainJ is now fully running and recording the calls for your J2EE application. Go ahead and do a simple test like logging into the application. Then go back and click Stop Trace to retrieve the .ser files. Next, go back to Eclipse and refresh your Trace Project (F5). You should now see the newly created .ser files in the project. Finally, just double click on the file and you should now be able to view and interact with the state diagram. Woo-Hoo, you should now be fully set up with MaintainJ!

References:

MaintainJ Homepage

<http://www.maintainj.com/>

Installing and Configuring MaintainJ

<http://maintainj.com/userGuide.jsp?param=install>

Support Forum

<http://www.maintainj.com/smf/>