

Running Jetty with jconsole

 These instructions apply to Jetty 6. For instructions for Jetty 7, see [the Jetty 7 wiki](#).

How to Run Jetty with jconsole

To run Jetty, start jconsole and then start Jetty with a special system property:

Jetty Standalone

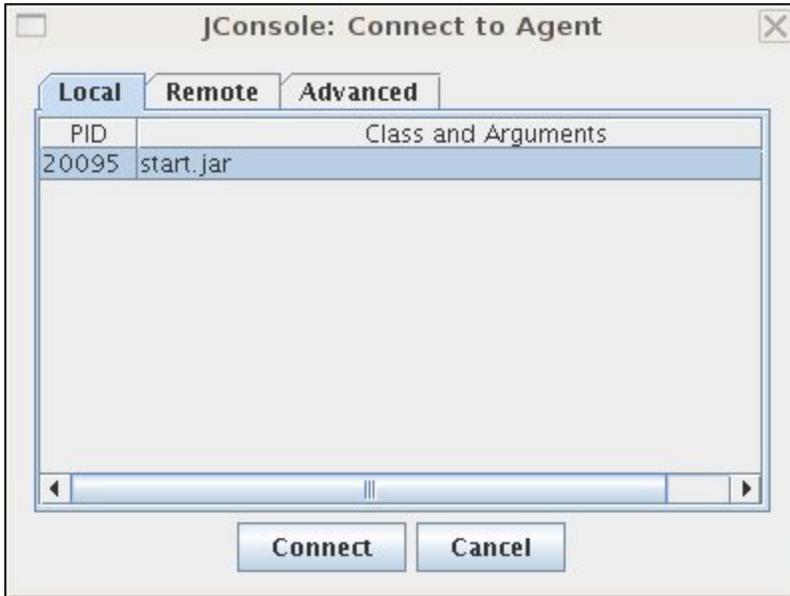
```
jconsole &  
java -Dcom.sun.management.jmxremote -jar  
start.jar [config files]
```

Jetty Maven Plugin

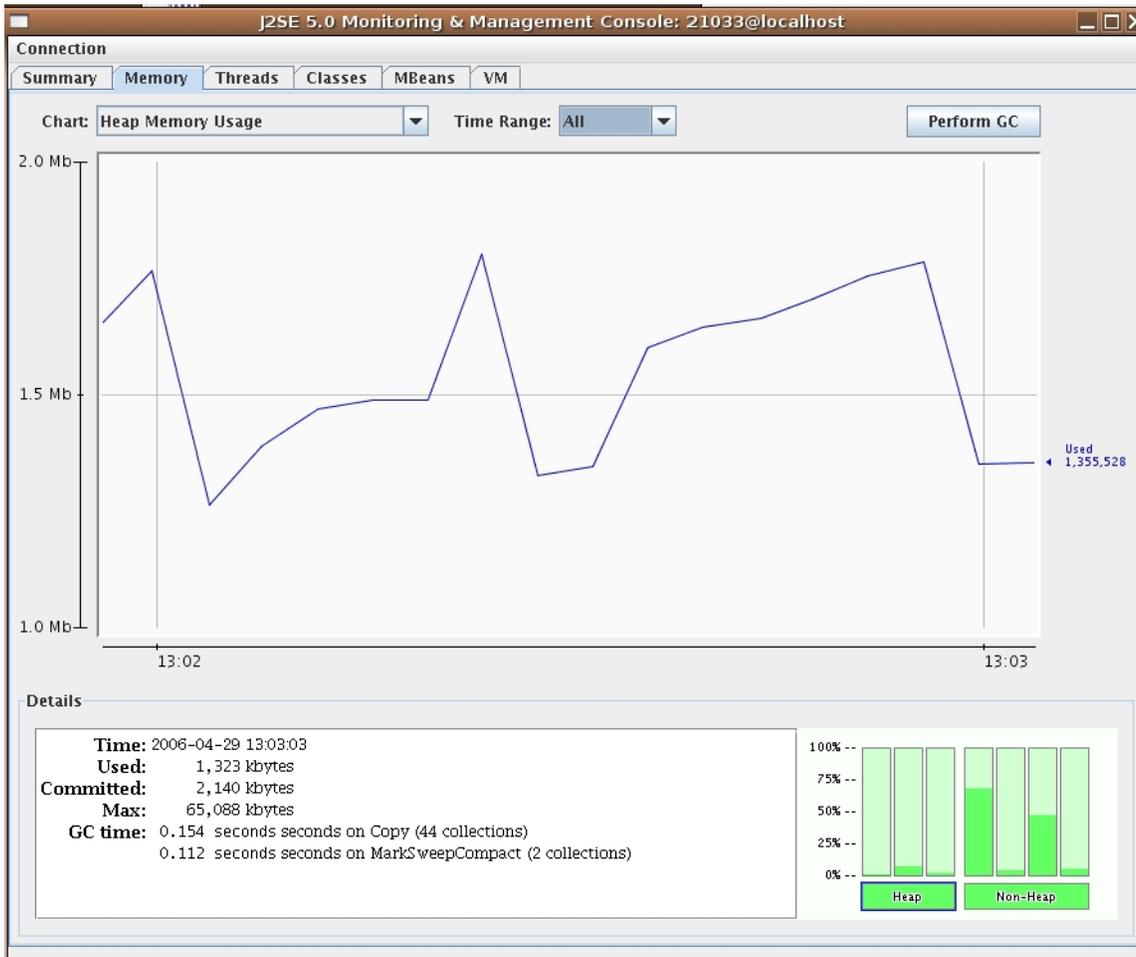
If you are running the [jetty maven plugin](#), you need to tell Maven to set the system property `com.sun.management.jmxremote` before running the plugin. The way to do this is to set the environment variable `MAVEN_OPTS`. For example, on UNIX systems:

```
export  
MAVEN_OPTS=-Dcom.sun.management.jmxremote  
mvn jetty:run
```

You should see a dialog like so (**Note** that with the jetty maven plugin, the entry will say "org.codehaus.classworlds.Launcher "jetty:run"" instead):



Select the start.jar entry and click on the Connect button. A new jconsole window will open:



From this window you will be able to monitor memory usage, thread usage, classloading and vm statistics. You'll also be able to perform operations such as doing a manual garbage collect. Jconsole is an extremely powerful and useful tool.

Using the Jetty MBeans with jconsole

The MBean tab of jconsole allows access to managed objects within the java application, including MBeans provided by the JVM.

If you also want to be able to interact with the [Jetty JMX](#) implementation via jconsole, then you need to start [Jetty JMX](#) in a form that jconsole can access.

Jetty Standalone

```
java -Dcom.sun.management.jmxremote -jar
start.jar etc/jetty-jmx.xml etc/jetty.xml
[config files]
jconsole &
```

See [JMX](#) for more details.

Jetty Maven Plugin

The easiest way to do this is to supply the etc/jetty-jmx.xml file in the plugin [configuration](#) . Do this like so:

```
<plugin>
  <groupId>org.mortbay.jetty</groupId>

  <artifactId>maven-jetty-plugin</artifactId>
  <version>6.1.18</version>
  <configuration>
    ...

  <jettyConfig>../../etc/jetty-jmx.xml</jettyC
onfig>
    ...
  </configuration>
</plugin>
```



If you already have a jetty configuration file that you use with the plugin, you can merge the contents of `etc/jetty-jmx.xml` into it