

# Maven2 Plugin Tips

## Tips

- Starting multiple containers conditionally
- Starting Tomcat in security mode
- How to get Tomcat 5 working with the Java 5 XML Parsers
- How to use the ZIP or TAR.GZ distributions provided by servers on a Maven2 repository

## Starting multiple containers conditionally

Maven 2 supports the notion of [profiles](#) which can be used with Cargo to decide for example when to run tests on a specific container. Here's how you could use the Cargo m2 plugin to that effect:

```
<project>
[... ]
  <profiles>
    <profile>
      <id>tomcat5x</id>
      <build>
        <plugins>
          <plugin>
            <groupId>org.codehaus.cargo</groupId>
            <artifactId>cargo-maven2-plugin</artifactId>
            <configuration>
              <container>
                <containerId>tomcat5x</containerId>
                <zipUrlInstaller>

<url>http://www.apache.org/dist/jakarta/tomcat-5/v5.0.30/bin/jakarta-tomcat-5.0.30.zip
</url>

                <downloadDir>${project.build.directory}/downloads</downloadDir>
                <extractDir>${project.build.directory}/extracts</extractDir>
              </zipUrlInstaller>
            </container>
          </configuration>
          <home>${project.build.directory}/tomcat5x/container</home>
        </configuration>
      </configuration>
      <executions>
        <execution>
          <id>start-container</id>
          <phase>pre-integration-test</phase>
          <goals>
            <goal>start</goal>
          </goals>
        </execution>
        <execution>
          <id>stop-container</id>
          <phase>post-integration-test</phase>
          <goals>
            <goal>stop</goal>
          </goals>
        </execution>
      </executions>
    </profile>
  </profiles>
</project>
```

```
    </plugin>
  </plugins>
</build>
</profile>
<profile>
  <id>orion2x</id>
  <build>
    <plugins>
      <plugin>
        <groupId>org.codehaus.cargo</groupId>
        <artifactId>cargo-maven2-plugin</artifactId>
        <configuration>
          <container>
            <containerId>orion2x</containerId>
            <zipUrlInstaller>
              <url>http://www.orionserver.com/distributions/orion2.0.5.zip</url>
              <downloadDir>${project.build.directory}/downloads</downloadDir>
              <extractDir>${project.build.directory}/extracts</extractDir>
            </zipUrlInstaller>
          </container>
          <configuration>
            <home>${project.build.directory}/orion2x/container</home>
          </configuration>
        </configuration>
      </plugin>
      <executions>
        <execution>
          <id>start-container</id>
          <phase>pre-integration-test</phase>
          <goals>
            <goal>start</goal>
          </goals>
        </execution>
        <execution>
          <id>stop-container</id>
          <phase>post-integration-test</phase>
          <goals>
            <goal>stop</goal>
          </goals>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>
</profile>
```

```
</profiles>
</project>
```

Then to start the tomcat 5.x container you would type `mvn -P tomcat5x install`. if you want to start both containers you would type `mvn -P tomcat5x,orion2x install`.

If you want to define a profile as the default you can use the `<activation>` element with an activation strategy. For example if you want a profile to be always on, use:

```
<profiles>
  <profile>
    <id>tomcat5x</id>
    <activation>
      <activeByDefault>>true</activeByDefault>
    </activation>
  [...]
[...]
```

## Starting Tomcat in security mode

Cargo supports passing system properties [Passing system properties](#). So, to start Tomcat in security mode, you need to specify two system properties:

- `java.security.manager`
- `java.security.policy`

For instance,

```
[...]
<plugins>
  [...]
  <plugin>
    <groupId>org.codehaus.cargo</groupId>
    <artifactId>cargo-maven2-plugin</artifactId>
    <configuration>
      <container>
        <containerId>tomcat5x</containerId>
        <home>${catalina.home}</home>
        [...]
        <systemProperties>
          <java.security.manager>default</java.security.manager>
          <java.security.policy>${catalina.home}\conf\catalina.policy</java.security.policy>
        </systemProperties>
      </container>
    </configuration>
  </plugin>
  [...]
</plugins>
[...]
```

## How to get Tomcat 5 working with the Java 5 XML Parsers

**Tip submitted by Ben P.**

Imagine that you have some XML jars in the `common/endorsed` folder of Tomcat and you have edited your `catalina.bat` file to specify some extra JVM opts to specify the XML parser. Here's how to achieve the same using Cargo:

```
<plugin>
  <groupId>org.codehaus.cargo</groupId>
  <artifactId>cargo-maven2-plugin</artifactId>
  <configuration>
    <container>
      <containerId>tomcat5x</containerId>
      <home>${tomcat.home}</home>
      <dependencies>
        <dependency>
          <location>${tomcat.home}/common/endorsed/xalan.jar</location>
        </dependency>
        <dependency>
          <location>${tomcat.home}/common/endorsed/xercesImpl.jar</location>
        </dependency>
      </dependencies>
      <systemProperties>
        <javax.xml.parsers.DocumentBuilderFactory>
          org.apache.xerces.jaxp.DocumentBuilderFactoryImpl
        </javax.xml.parsers.DocumentBuilderFactory>
        <javax.xml.parsers.SAXParserFactory>
          org.apache.xerces.jaxp.SAXParserFactoryImpl
        </javax.xml.parsers.SAXParserFactory>
      </systemProperties>
    </container>
    <configuration>
      <type>existing</type>
      <home>${tomcat.home}</home>
    </configuration>
  </configuration>
</plugin>
```

This shows how to add classpath elements to a Cargo container using the `<dependencies>` element.