Meecrowave Maven

Coordinates:

```xml
<plugin>
  <groupId>org.apache.meecrowave</groupId>
  <artifactId>meecrowave-maven-plugin</artifactId>
  <version>${meecrowave.version}</version>
</plugin>
```

💡 most of the configuration is inherited from meecrowave-core.

Here are the available options (see core configuration for the details):

Name	Default	Property
antiResourceLocking	false	${meecrowave.antiResourceLocking}
cdiConversation	false	${meecrowave.cdiConversation}
clientAuth	-	${meecrowave.clientAuth}
conf	-	${meecrowave.conf}
context		${meecrowave.context}
defaultSSLHostConfigName	-	${meecrowave.default-ssl-hostconfig-name}
deleteBaseOnStartup	true	${meecrowave.deleteBaseOnStartup}
dir	-	${meecrowave.dir}
forceLog4j2Shutdown	true	${meecrowave.force-log4j2-shutdown}
host	localhost	${meecrowave.host}
http2	-	${meecrowave.http2}
httpPort	8080	${meecrowave.http}
httpsPort	8443	${meecrowave.https}
initializeClientBus	true	${meecrowave.initialiaze-client-bus}
injectServletContainerInitializer	true	${meecrowave.servlet-container-initializer-injections}
jaxrsAutoActivateBeanValidation	true	${meecrowave.jaxrs-beanvalidation}
jaxrsDefaultProviders	-	${meecrowave.jaxrs-default-providers}
jaxrsLogProviders	false	${meecrowave.jaxrsLogProviders}
jaxrsMapping	/*	${meecrowave.jaxrsMapping}

Name	Default	Property
jaxrsProviderSetup	true	${meecrowave.jaxrs-provider-setup}
jaxwsSupportIfAvailable	true	${meecrowave.jaxws-support}
jsonbBinaryStrategy	-	${meecrowave.jsonb-binary-strategy}
jsonbEncoding	UTF-8	${meecrowave.jsonb-encoding}
jsonbIJson	false	${meecrowave.jsonb-ijson}
jsonbNamingStrategy	-	${meecrowave.jsonb-naming-strategy}
jsonbNulls	false	${meecrowave.jsonb-nulls}
jsonbOrderStrategy	-	${meecrowave.jsonb-order-strategy}
jsonbPrettify	false	${meecrowave.jsonb-prettify}
jsonpBufferStrategy	QUEUE	${meecrowave.jsonp-buffer-strategy}
jsonpMaxReadBufferLen	65536	${meecrowave.jsonp-max-read-buffer-size}
jsonpMaxStringLen	10485760	${meecrowave.jsonp-max-string-length}
jsonpMaxWriteBufferLen	65536	${meecrowave.jsonp-max-write-buffer-size}
jsonpPrettify	false	${meecrowave.jsonp-prettify}
jsonpSupportsComment	false	${meecrowave.jsonp-comments}
keepServerXmlAsThis	-	${meecrowave.keepServerXmlAsThis}
keyAlias	-	${meecrowave.keyAlias}
keystoreFile	-	${meecrowave.keystoreFile}
keystorePass	-	${meecrowave.keystorePass}
keystoreType	JKS	${meecrowave.keystoreType}
loggingGlobalSetup	true	${meecrowave.logging-global-setup}
meecrowaveProperties	meecrowave.properties	${meecrowave.meecrowave-properties}
modules	${project.build.outputDirectory}	
project	${project}	
quickSession	true	${meecrowave.quickSession}
reloadGoals	-	${meecrowave.reload-goals}

Name	Default	Property
scanningExcludes	-	${meecrowave.scanning-exclude}
scanningIncludes	-	${meecrowave.scanning-include}
scanningPackageExcludes	-	${meecrowave.scanning-package-exclude}
scanningPackageIncludes	-	${meecrowave.scanning-package-include}
serverXml	-	${meecrowave.serverXml}
session	${session}	
sharedLibraries	-	${meecrowave.shared-libraries}
skip	-	${meecrowave.skip}
skipHttp	-	${meecrowave.skipHttp}
ssl	-	${meecrowave.ssl}
sslProtocol	-	${meecrowave.sslProtocol}
stopPort	8005	${meecrowave.stop}
tempDir	-	${meecrowave.tempDir}
tomcatAccessLogPattern	-	${meecrowave.tomcatAccessLogPattern}
tomcatAutoSetup	true	${meecrowave.tomcatAutoSetup}
tomcatFilter	-	${meecrowave.tomcatFilter}
tomcatJspDevelopment	false	${meecrowave.tomcatJspDevelopment}
tomcatNoJmx	true	${meecrowave.tomcatNoJmx}
tomcatScanning	true	${meecrowave.tomcatScanning}
tomcatWrapLoader	false	${meecrowave.tomcatWrapLoader}
useClasspathDeployment	true	${meecrowave.force-classpath-deployment}
useLog4j2JulLogManager	true	${meecrowave.log4j2-jul-bridge}
useShutdownHook	true	${meecrowave.shutdown-hook}
useTomcatDefaults	true	${meecrowave.useTomcatDefaults}
watcherBouncing	0	${meecrowave.watcher-bounding}

Name	Default	Property
webResourceCached	true	${meecrowave.webResourceCached}
webSessionCookieConfig	-	${meecrowave.session-cookie-config}
webSessionTimeout	-	${meecrowave.session-timeout}
webXml	-	${meecrowave.webXml}
webapp	${project.basedir}/src/main/webapp	${meecrowave.webapp}

Run

`mvn meecrowave:run` enables you to start a server configured in your `pom.xml`. Once started, you have a few commands you can use to interact with the server:

- `quit`/`exit`: shutdown properly the server.
- `reload` (since 1.2.9): optionally relaunch a maven compilation - see configuration - and reload the web context.

Bundling

```
mvn meecrowave:bundle
```

This maven goal will create a zip/tar.gz (configurable) ready to launch your application.

The layout if the following one:

```
.
├──── bin
│      └──── meecrowave.sh
├──── conf
│      ├──── log4j2.xml
│      └──── meecrowave.properties
├──── lib
│      └──── *.jar
├──── logs
│      └──── meecrowave.logls l
└──── temp
```

conf folder supports out of the box these files (but you can add your own if needed):

- log4j2.xml
- server.xml

- meecrowave.properties

You can acess the root folder of the distribution reading the system property `meecrowave.base`.

`meecrowave.properties` supports all configuration options of the cli (just remove the `--`): http://openwebbeans.apache.org/meecrowave/meecrowave-core/cli.html.

Note: you can also split like Tomcat MEECROWAVE_BASE and MEECROWAVE_HOME if you want to extract meecrowave stack in a separate folder (from the application) to be able to update it independently.

Shading

Plain Maven shade plugin enables you to build fatjar with Maven.

Points to take care are:

1. merge CXF resources

2. merge Log4j2 resources

3. merge OpenWebBeans resources

4. merge SPI files (META-INF/services)

5. ensure signatures are still valid in the aggregated jar

```xml
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-shade-plugin</artifactId>
  <version>3.2.4</version>
  <executions>
    <execution>
      <id>bundle</id>
      <phase>package</phase>
      <goals>
        <goal>shade</goal>
      </goals>
      <configuration>
        <shadedClassifierName>fat</shadedClassifierName>
        <shadedArtifactAttached>true</shadedArtifactAttached>
        <dependencyReducedPomLocation>${project.build.directory}/reduced-pom-bundle.xml</dependencyReducedPomLocation>
        <transformers>
          <transformer implementation=
"org.apache.maven.plugins.shade.resource.ManifestResourceTransformer">
            <mainClass>org.apache.meecrowave.runner.Cli</mainClass>
          </transformer>
          <transformer implementation=
"org.apache.maven.plugins.shade.resource.AppendingTransformer">
            <resource>META-INF/cxf/bus-extensions.txt</resource>
          </transformer>
```

```xml
          <transformer implementation=
"org.apache.maven.plugins.shade.resource.AppendingTransformer">
            <resource>log4j2.component.properties</resource>
          </transformer>
          <transformer implementation=
"com.github.edwgiz.maven_shade_plugin.log4j2_cache_transformer.PluginsCacheFileTransfo
rmer" />
          <transformer implementation=
"org.apache.maven.plugins.shade.resource.properties.OpenWebBeansPropertiesTransformer"
/>
          <transformer implementation=
"org.apache.maven.plugins.shade.resource.ServicesResourceTransformer"/>
        </transformers>
        <filters>
          <filter>
            <artifact>*:*</artifact>
            <excludes>
              <exclude>META-INF/*.SF</exclude>
              <exclude>META-INF/*.DSA</exclude>
              <exclude>META-INF/*.RSA</exclude>
              <exclude>META-INF/LICENSE.txt</exclude>
              <exclude>META-INF/LICENSE</exclude>
              <exclude>META-INF/NOTICE.txt</exclude>
              <exclude>META-INF/NOTICE</exclude>
            </excludes>
          </filter>
        </filters>
      </configuration>
    </execution>
  </executions>
  <dependencies>
    <dependency>
      <groupId>com.github.edwgiz</groupId>
      <artifactId>maven-shade-plugin.log4j2-cachefile-transformer</artifactId>
      <version>2.13.3</version>
    </dependency>
  </dependencies>
</plugin>
```