**JAXB2 Maven Plugin:**

This plugin uses the Java API for XML Binding (JAXB), version 2+, to generate Java classes from XML Schemas (and optionally binding files) and to create XML Schemas from annotated Java classes.

The jaxb2-maven-plugin has four main goals, listed below. Detailed information about each goal can be found on their respective documentation page.

* jaxb2:schemagen: Creates XML Schema Definition (XSD) file(s) from annotated Java sources.

<https://www.mojohaus.org/jaxb2-maven-plugin/Documentation/v2.2/schemagen-mojo.html>

* jaxb2:testSchemagen: Creates XML Schema Definition (XSD) file(s) from annotated Java test sources.

<https://www.mojohaus.org/jaxb2-maven-plugin/Documentation/v2.2/testSchemagen-mojo.html>

* jaxb2:xjc: Generates Java sources from XML Schema(s).

<https://www.mojohaus.org/jaxb2-maven-plugin/Documentation/v2.2/xjc-mojo.html>

* jaxb2:testXjc: Generates Java test sources from XML Schema(s).

<https://www.mojohaus.org/jaxb2-maven-plugin/Documentation/v2.2/testXjc-mojo.html>

Requirement:

* Requires a Maven project to be executed.
* By default jaxb2-maven-plugin will use: generate-sources lifecycle-phase to generate files.
* You have to use generate-sources maven command to generate files
* The plugin sample for the Schema to Java Conversion & vice-versa

Parameters used in plugin:

* outputDirectory: Directory where your generated files will be generated.

In our project it’s in src/main/java of the project

Ex: <outputDirectory>${basedir}/src/main/java</outputDirectory>

Default value: ${project.build.directory}/generated-sources/jaxb.

* clearOutputDir:

Removes all files from the output directory before running XJC.

Type: Boolean

Required: No

Default: true

* noGeneratedHeaderComments:

Corresponding XJC parameter: no-header.

Suppress the generation of a file header comment that includes some note and timestamp. Using this makes the generated code more diff-friendly.

Type: Boolean

Since: 2.0

Required: No

Default: false

* generateEpisode:

Corresponding XJC parameter: episode.

Generate an episode file from this compilation, so that other schemas that rely on this schema can be compiled later and rely on classes that are generated from this compilation. The generated episode file is really just a JAXB customization file (but with vendor extensions.)

If this parameter is true, the episode file generated is called

META-INF/sun-jaxb.episode, and included in the artifact.

Type: Boolean

Since: 2.0

Required: No

Default: true

* sourceType:

Defines the content type of sources for the XJC. To simplify usage of the JAXB2 maven plugin, all source files are assumed to have the same type of content.

This parameter replaces the previous multiple-choice boolean configuration options for the jaxb2-maven-plugin (i.e. dtd, xmlschema, relaxng, relaxng-compact, wsdl), and corresponds to setting one of those flags as an XJC argument.

Type: org.codehaus.mojo.jaxb2.javageneration.SourceContentType

Since: 2.0

Required: No

Default: XmlSchema

* sources:

Parameter holding List of XSD paths to files and/or directories which should be recursively searched for XSD files. Only files or directories that actually exist will be included (in the case of files) or recursively searched for XSD files to include.

We have to specify list of xsd or wsdl file paths within child tag <source>

<sources>

<source>....</source>

.....

...

</source>

* packageName:

The package under which the source files will be generated. Quoting the XJC documentation: "Specifying a target package via this command-line option overrides any binding customization for package name and the default package name algorithm defined in the specification".

Type: java.lang.String

Required: No

The jaxb2-maven-plugin is compatible with m2e and integrates with the Eclipse build, providing incremental build support in the IDE. This requires m2e v1.1 or later.

The sample plugin which will generate Schema files from WSDL file

<plugin>

<groupId>org.codehaus.mojo</groupId>

<artifactId>jaxb2-maven-plugin</artifactId>

<version>2.3</version>

<executions>

<execution>

<id>xsd</id>

<goals>

<goal>xjc</goal>

</goals>

<configuration>

<packageName>${jaxb2.context.xsd}</packageName>

<outputDirectory>${basedir}/src/main/java</outputDirectory>

<clearOutputDir>false</clearOutputDir>

<noGeneratedHeaderComments>true</noGeneratedHeaderComments>

<generateEpisode>false</generateEpisode>

</configuration>

</execution>

<execution>

<id>wsdl</id>

<goals>

<goal>xjc</goal>

</goals>

<configuration>

<packageName>${jaxb2.context.wsdl}</packageName>

<sources> <source>src/main/resources/wsdl/avi\_wsdl.wsdl</source>

</sources>

<outputDirectory>${basedir}/src/main/java</outputDirectory>

<clearOutputDir>falsef</clearOutputDir>

<noGeneratedHeaderComments>true</noGeneratedHeaderComments>

<generateEpisode>false</generateEpisode>

<sourceType>wsdl</sourceType>

</configuration>

</execution>

<execution>

<id>avi-wsdl</id>

<goals>

<goal>xjc</goal>

</goals>

<configuration>

<packageName>${jaxb2.context.avi.wsdl}</packageName>

<sources>

<source>src/main/resources/wsdl/wsdl11\_MS.wsdl</source>

</sources>

<outputDirectory>${basedir}/src/main/java</outputDirectory>

<clearOutputDir>falsef</clearOutputDir>

<noGeneratedHeaderComments>true</noGeneratedHeaderComments>

<generateEpisode>false</generateEpisode>

<sourceType>wsdl</sourceType>

</configuration>

</execution>

</executions>

</plugin>

<properties>

…….

……..

<jaxb2.context.xsd>com.bshg.hip.cus.mr.erp.model.generated</jaxb2.context.xsd>

<jaxb2.context.wsdl>com.bshg.hip.cus.mr.erp.model.generated.envelop</jaxb2.context.wsdl>

<jaxb2.context.avi.wsdl>com.bshg.hip.cus.mr.erp.model.generated.envelop.avi</jaxb2.context.avi.wsdl>

</properties>

${jaxb2.context.avi.wsdl} – The value for ${jaxb2.context.avi.wsdl} is specified inside properties tag ${jaxb2.context.avi.wsdl} will store the wsdl file in the package structure specified within tag <jaxb2.context.wsdl>

<jaxb2.context.wsdl>com.bshg.hip.cus.mr.erp.model.generated.envelop</jaxb2.context.wsdl>

Similarly things works for ${jaxb2.context.xsd}

References:

<https://www.mojohaus.org/jaxb2-maven-plugin/Documentation/v2.2/>

<https://www.mojohaus.org/jaxb2-maven-plugin/Documentation/v2.2/xjc-mojo.html>