

Camel Maven Archetypes

Camel Maven Archetypes

Camel is distributed with the following archetypes for Maven end users.

Archetype Supported

Archetype	Description
camel-archetype-activemq	This archetype is used to create a new Maven project for Camel routes with Apache ActiveMQ embedded.
camel-archetype-blueprint	Camel 2.8: This archetype is used to create a new Maven project for Camel routes to be running in OSGi using Blueprint.
camel-archetype-component	This archetype is used for creating a new Maven project for Camel Components . Use this if there is a component missing in Camel that you want to create yourself.
camel-archetype-cdi	Camel 2.16: This archetype is used to create a new Maven project for Camel routes in the Java Container using CDI to configure components, endpoints and beans
camel-archetype-component-scala	Deprecated - Camel 2.10: This archetype is used for creating a new Maven project for Camel Components using Scala. Use this if there is a component missing in Camel that you want to create yourself. Deprecated to be removed in Camel 2.16 onwards.
camel-archetype-exf-code-first-blueprint	Deprecated - Camel 2.12.2: This archetype is used creating a new Maven project for Camel with CXF exposing a web service using code first style. This project is for OSGi Blueprint containers. Removed from Camel 2.18 onwards.
camel-archetype-exf-contract-first-blueprint	Deprecated - Camel 2.12.1: This archetype is used creating a new Maven project for Camel with CXF exposing a web service using contract first style. This project is for OSGi Blueprint containers. Removed from Camel 2.18 onwards.
camel-archetype-dataformat	Camel 2.9: This archetype is used for creating a new Maven project for Camel Data Formats . Use this if there is a data format missing in Camel that you want to create yourself.
camel-archetype-groovy	Camel 2.11: This archetype is used to create a new Maven project for Camel using Groovy DSL.
camel-archetype-java	This archetype is used to create a new Maven project for Camel routes using Java DSL.
camel-archetype-scala	This archetype is used to create a new Maven projects for building Camel routes in the Scala DSL .
camel-archetype-scr	This archetype is used to create a new Maven project for building Camel routes using OSGi SCR Declarative Services
camel-archetype-spring	This archetype is used to create a new Maven project for Camel routes using Spring XML.
camel-archetype-spring-boot	Camel 2.16: This archetype is used to create a new Maven project for Camel routes using Spring Boot.
camel-archetype-spring-dm	Camel 2.8: This archetype is used to create a new Maven project for Camel routes to be running in OSGi using Spring DM.
camel-archetype-war	Deprecated - Camel 2.6 or older: This archetype is used to create a new Camel project that deploys the Camel Web Console, REST API, and your routes as a WAR file.
camel-archetype-webconsole	Deprecated - Camel 2.7: This archetype is used to create a new Camel project that deploys the Camel Web Console, REST API, and your routes as a WAR file.
camel-archetype-web	Camel 2.7: This archetype is used to create a new Camel project that deploys Camel Routes using Spring or Java DSL as a WAR file.

The maven coordinates for these archetypes is the following:

```
<groupId>org.apache.camel.archetypes</groupId>
<artifactId>${archetype-name}</artifactId>
<version>${camel-version}</version>
```

When using these archetypes, make sure your package name is not org.apache.camel (or a sub package of this) as this will instruct Camel to search in its own packages for your routes. This can also cause Camel not to start, as shown in CAMEL-1197.

Snapshot archetypes

If you would like to use an archetype from an unreleased version of Camel, you just need to let the maven-archetype-plugin know where to look for it. For example, say someone wanted to create a WAR based project based on Camel 2.9.0. He'd need to use the following command

```
mvn archetype:generate \  
-DarchetypeGroupId=org.apache.camel.archetypes \  
-DarchetypeArtifactId=camel-archetype-web \  
-DarchetypeVersion=2.9.0 \  
-DarchetypeRepository=https://repository.apache.org/content/groups/snapshots-group
```

Notice that archetypes will now be downloaded from [Apache's snapshot repo](#) instead of [Maven central](#).

When the project is created, you may need to add another repository to the pom.xml file. This is to load a SNAPSHOT version of the camel-maven-plugin

```
<pluginRepositories>  
  <pluginRepository>  
    <releases>  
      <enabled>false</enabled>  
    </releases>  
    <snapshots/>  
    <id>Apache Snapshot Repository</id>  
    <url>http://repository.apache.org/snapshots</url>  
  </pluginRepository>  
</pluginRepositories>
```

Examples

See [Creating a new Spring based Camel Route](#) for a sample how to use the camel-archetype-java and camel-archetype-spring archetypes.