vert.x Maven Plugin

Kamesh Sampath,Roland Huß

Version 1.0-SNAPSHOT, 2016-11-15

Vert.X Maven Plugin

Chapter 1. Introduction

This is a Maven plugin for packaging and running vert.x applications

1.1. Packaging

The vert.x application needs to be packaged as fat or uber jar, which can then be executed using `java -jar <my-app-fat.jar>`. The packaging also need to add certain **MANIFEST.MF** entires that controls the vert.x application launching. The plugin takes care of adding the required entries to the **MANIFEST.MF**.

The following are the Manifest entries that will be added based on the configuration elements:

Table 1. Package configuration

Property	Manifest Attribute	Remarks
vertx.verticle	Main-Verticle	
vertx.launcher	Main-class	If custom launcher is used the Main-Class is always defaulted to `io.vertx.core.Launcher`

To package the project as vert.x application, execute the following maven command

```
mvn clean package vertx:package
```

For more information on packaging refer to vertx:package

1.2. Running

The vert.x application could also be run without building building the fat jars, typically during the development mode. The vertx:run goal helps in running the application. The application will be run in either forked or non-forked mode based on the configuration of the goal.

To start the application vert.x application, execute the following maven command

```
mvn clean package vertx:run
```

For more information on configuration options refer to vertx:run

Chapter 2. Common configuration

All goals share the following configuration:

Table 2. Package configuration

Element	Description	Default	Property
`verticle`	Main verticle to start up		`vertx.vert icle`
`launcher`	Vert.x launcher to use	`io.vertx.c ore.Launch er`	`vertx.laun cher`

Chapter 3. Maven Goals

This plugin supports the following goals which are explained in detail in the next sections.

Table 3. Plugin Goals

Goal	Description
vertx.package	Package Vert.x applications
vertx:run	Start a Vert.x application

3.1. vertx.package

This goal will package a vert.x application as fat or uber jar with its dependencies bundled as part of the jar

3.1.1. Configuration

The package goal does not have any additional configuration other the ones mentioned in Common Configuration

3.2. vertx:run

This goal allows to run the vert.x applicaiton as part of the maven build. The applicaiton run can either be run as forked and non-forked process. By default the applicaiton is run in non-forked mode.

Table 4. Run configuration

Element	Description	Property	Default
classesDirectory*	The project classes directory where the sources will be compiled to.		`${project.build.outputDirectory}`
config	the application configuration file path that will be passed to the vertx launcher as `-conf`. If a yaml file is configured then it will be converted to json by the plugin	vertx.config	`${basedir}/src/main/${project.artifactId}.json` or `${basedir}/src/main/${project.artifactId}.yaml`
fork	Controls whether the application will be run in forked mode or non-forked mode	fork	false
redeploy	controls whether vertx redeploy is enabled		false

Element	Description	Property	Default
redeployPatterns	The ant based pattern for scanning changes for redeployment. If redeploy is **true** and redeployPatterns is empty then a default value is will be applied. This is list of values following standard maven list/array configuration		`src/main/**/*.java`
runArgs	the additional arguments that needs to be passed to JVM	vertx.jvmArguments	
workDirectory	The working directory of the running process of the application	vertx.directory	`${project.basedir}`

All elements with * is required