

# Deployment

The plugin provides various methods of deployment to Tomcat:

- [Deploying a WAR file](#)
- [Deploying an exploded WAR directory](#), with an optional context.xml file
- [Deploying an in-place WAR directory](#), with an optional context.xml file
- [Deploying a context.xml file](#)
- [Running a WAR project](#)

These are described in more detail below.

## Deploying a WAR file

The simplest way to deploy a WAR project to Tomcat is to type:

```
mvn tomcat:deploy
```

This goal will assemble and deploy the WAR file to Tomcat manager using HTTP PUT.

## Using a different WAR file location

To specify a different WAR file location, add a plugin configuration block to your pom.xml as follows:

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>tomcat-maven-plugin</artifactId>
  <configuration>
    <warFile>path/to/my/warFile.war</warFile>
  </configuration>
</plugin>
```

The default location is

`${project.build.directory}/${project.build.finalName}.war`.

## Using a context.xml file

If you need to specify a context.xml file when deploying a WAR file to Tomcat, then it must be included within the WAR. The simplest way to achieve this is by adding it to your webapp resources:

```
src
|_ main
|_ webapp
|_ META-INF
|_ context.xml
```

## Deploying an exploded WAR directory

To avoid building a WAR file upon deployment, a WAR directory can instead be deployed to Tomcat by typing:

```
mvn war:exploded tomcat:exploded
```

## Using a different WAR directory location

To specify a different WAR directory location, add a plugin configuration block to your pom.xml as follows:

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>tomcat-maven-plugin</artifactId>
  <configuration>
    <warDirectory>path/to/my/warDir</warDirectory>
  </configuration>
</plugin>
```

The default location is

`${project.build.directory}/${project.build.finalName}.`

## Using a context.xml file

To supply a context.xml when deploying a WAR directory, add a plugin configuration block to your pom.xml as follows:

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>tomcat-maven-plugin</artifactId>
  <configuration>
    <mode>both</mode>
  </configuration>
</plugin>
```

The default context.xml file use is located at `src/main/webapp/META-INF/context.xml`.

## Using a different context.xml file location

To specify a different context.xml file location, add a plugin configuration block to your pom.xml as follows:

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>tomcat-maven-plugin</artifactId>
  <configuration>
    <contextFile>path/to/my/contextFile.xml</contextFile>
  </configuration>
</plugin>
```

The default location is

`${project.build.directory}/${project.build.finalName}/META-INF/context.xml`.

## Deploying an in-place WAR directory

To avoid copying resources to the build directory, the webapp source directory can be deployed to Tomcat by typing:

```
mvn war:inplace tomcat:inplace
```

## Using a different WAR directory location

To specify a different WAR directory location, add a plugin configuration block to your pom.xml as follows:

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>tomcat-maven-plugin</artifactId>
  <configuration>
    <warSourceDirectory>path/to/my/warSourceDir</warSourceDirectory>
  </configuration>
</plugin>
```

The default location is `${basedir}/src/main/webapp`.

## Using a context.xml file

To supply a context.xml when deploying a WAR directory to Tomcat, add a plugin configuration block to your pom.xml as follows:

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>tomcat-maven-plugin</artifactId>
  <configuration>
    <mode>both</mode>
  </configuration>
</plugin>
```

The default context.xml file used is located at `src/main/webapp/META-INF/context.xml`.

## Using a different context.xml file location

To specify a different context.xml file location, add a plugin configuration block to your pom.xml as follows:

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>tomcat-maven-plugin</artifactId>
  <configuration>
    <contextFile>path/to/my/contextFile.xml</contextFile>
  </configuration>
</plugin>
```

The default location is

`${project.build.directory}/${project.build.finalName}/META-INF/context.xml`.

## Deploying a context.xml file

To simply deploy just a context.xml file to Tomcat:

1. Add a plugin configuration block to your pom.xml as follows:

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>tomcat-maven-plugin</artifactId>
  <configuration>
    <mode>context</mode>
  </configuration>
</plugin>
```

2. Deploy the context.xml file by typing:

```
mvn tomcat:deploy
```

The default context.xml file used is located at `src/main/webapp/META-INF/context.xml`.

## Using a different context.xml file location

To specify a different context.xml file location, add a plugin configuration block to your pom.xml as follows:

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>tomcat-maven-plugin</artifactId>
  <configuration>
```

```
        <contextFile>path/to/my/contextFile.xml</contextFile>
    </configuration>
</plugin>
```

## The default location is

`${project.build.directory}/${project.build.finalName}/META-INF/context.xml`.

## Running a WAR project

A WAR project can be run under an embedded Tomcat server by typing:

```
mvn tomcat:run
```

To stop the embedded server, press `CTRL+C`.