

Edit on GitHub (<https://github.com/apache/struts-site/edit/main/source/contributors/building-with-maven.md>)
← Back to Contributors Guide ([index.html](#))

Building with Maven

- Installing
- Other phases
- Maven Tips
 - Skip exporting pages from Confluence
 - Skip creating assemblies
 - Offline mode
 - OutOfMemoryError
 - Skipping test execution
 - Debugging build failures
- First time building

Maven 3.0.0 or later (<http://maven.apache.org>) is required to build Struts 2

First, let's review some Maven basics. Maven uses the notion of a build *life-cycle* to which plugins can attach. Plugins are similar to Ant tasks. When a Maven build is invoked, we specify a point in the life-cycle up to which the build *should* proceed. The *compile* phase comes before *test* , and *test* comes before *package* , and *package* comes before *install* . Once we have Maven setup, we can invoke the Struts build, and specify which phase the build should use.

Installing

The *install* phase builds up the project (“package”), and installs any JARs it needs into your local repository, e.g.:

```
~/m2/repository
```

Once installed, the JARs can be used by any other Maven project you build. The *install* phase is the **default** phase if none is specified.

To run a basic install, change to the root of the source distribution, and enter

```
mvn
```

That's it! Maven will download any dependencies the build needs; run all unit tests; package up the JARs; then install the new JARs locally. For your convenience, copies of the JARs can be found in the *target* directories of each module. For example, after the build, the main JAR can be found at

```
core/target/struts2-core-2.0-SNAPSHOT.jar
```

Sometimes, licensing restrictions prevent Maven from downloading all the JARs that a build might need. For example, JavaMail and Activation, can only be downloaded from Sun. When this happens, Maven will display a helpful message that explains how to install these JARs manually. After downloading the required JAR, follow the instructions to install it to your local repository. Once installed, the JAR is available to all your Maven builds, not just Struts.

Initial Build Successful

```

[INFO] -----
[INFO] Reactor Summary:
[INFO]
[INFO] Struts 2 Bill of Materials ..... SUCCESS [ 1.188 s]
[INFO] Struts 2 2.5.24-SNAPSHOT ..... SUCCESS [ 2.249 s]
[INFO] Struts 2 Core ..... SUCCESS [01:13 min]
[INFO] Struts Plugins ..... SUCCESS [ 0.232 s]
[INFO] Struts 2 Configuration Browser Plugin ..... SUCCESS [ 1.374 s]
[INFO] Struts 2 Sitemesh Plugin ..... SUCCESS [ 1.429 s]
[INFO] Struts 2 Tiles Plugin ..... SUCCESS [ 2.303 s]
[INFO] Struts 2 DWR Plugin ..... SUCCESS [ 0.905 s]
[INFO] Struts 2 Spring Plugin ..... SUCCESS [ 2.232 s]
[INFO] Struts 2 Convention Plugin ..... SUCCESS [ 4.540 s]
[INFO] Struts 2 JUnit Plugin ..... SUCCESS [ 4.297 s]
[INFO] Struts 2 JSON Plugin ..... SUCCESS [ 5.384 s]
[INFO] Struts 2 Bean Validation Plugin ..... SUCCESS [ 3.345 s]
[INFO] Struts 2 Webapps ..... SUCCESS [ 0.391 s]
[INFO] Struts 2 Showcase Webapp ..... SUCCESS [ 6.104 s]
[INFO] Struts 2 REST Plugin ..... SUCCESS [ 4.058 s]
[INFO] Struts 2 Rest Showcase Webapp ..... SUCCESS [ 1.524 s]
[INFO] Struts 2 CDI Plugin ..... SUCCESS [ 2.758 s]
[INFO] Struts 2 Embedded JSP Plugin ..... SUCCESS [ 8.111 s]
[INFO] Struts 2 GXP Plugin ..... SUCCESS [ 1.137 s]
[INFO] Struts 2 Jasper Reports Plugin ..... SUCCESS [ 4.392 s]
[INFO] Struts 2 Java Templates Plugin ..... SUCCESS [ 2.666 s]
[INFO] Struts 2 JFreeChart Plugin ..... SUCCESS [ 3.169 s]
[INFO] Struts 2 OSGi Plugin ..... SUCCESS [ 3.209 s]
[INFO] Struts 2 Oval Plugin ..... SUCCESS [ 3.113 s]
[INFO] Struts 2 Pell Multipart Plugin ..... SUCCESS [ 0.842 s]
[INFO] Struts 2 Plexus Plugin ..... SUCCESS [ 1.003 s]
[INFO] Struts 2 Portlet Plugin ..... SUCCESS [ 5.210 s]
[INFO] Struts 2 Portlet Tiles Plugin ..... SUCCESS [ 0.999 s]
[INFO] DEPRECATED: Struts 2 Sitegraph Plugin ..... SUCCESS [ 2.634 s]
[INFO] Struts 2 TestNG Plugin ..... SUCCESS [ 2.081 s]
[INFO] Struts OSGi Bundles ..... SUCCESS [ 0.083 s]
[INFO] Struts 2 OSGi Admin Bundle ..... SUCCESS [ 1.237 s]
[INFO] Struts 2 OSGi Demo Bundle 2.5.24-SNAPSHOT ..... SUCCESS [ 1.157 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 02:39 min
[INFO] Finished at: 2020-08-14T11:24:15+02:00
[INFO] -----

```

Other phases

There are other phases that can be useful when working with Maven. The *package* phase will just JAR (or WAR) the modules. The *test* phase will only execute unit-tests. The *compile* phase will only build the source-code (but not the test sources). And the *clean* phase will remove all artifacts, typically the entire

target directory.

Maven Tips

A few helpful tips for using Maven are provided:

Skip exporting pages from Confluence

The whole Struts2 documentation resides on Confluence and during normal build process is exported with SiteExporter tool locally to html files. To perform that, you must have an account on Confluence. In most cases you don't need to export the whole Confluence, so you can skip this, just type command:

```
mvn clean install -DskipWiki
```

Skip creating assemblies

During normal development cycle you don't need to create assembly packages which consist of all the libraries, example apps and docs - you can skip them as well:

```
mvn clean install -DskipAssembly
```

Offline mode

If you are disconnected from the Internet or simply wish to make your build faster, pass in the **-o** argument and Maven won't check for new modules to download.

```
mvn -o
```

OutOfMemoryError

If you get an OutOfMemoryError while running the tests:

```
// Bash
export MAVEN_OPTS=-Xmx512m

// Windows
set MAVEN_OPTS=-Xmx512m
```

Skipping test execution

Although this shouldn't ever happen, sometimes tests do fail and you need to build the framework anyway. If there's a problem, you can pass in the skip tests parameter.

```
mvn -Dmaven.test.skip=true
```

Of course, if you find tests are failing, please submit a patch to fix them!

Debugging build failures

Struts 2 depends on the current snapshot artifact of XWork 2. Unfortunately, if XWork 2 is modified in a significant way, the Struts build doesn't check for and retrieve a new version of the XWork snapshot jar, resulting in compilation or unit test failures.

The solution is to clear out any stale XWork 2 jars in your local Maven repository, usually by removing the directory

```
~/.m2/repository
```

This will force the Struts Maven build to pull down a fresh copy of XWork and hopefully resolve your problem.

First time building

In some cases it has been seen that Maven will complain if a module doesn't exist, even though it is part of the current build. Often, the missing module turns up when executing

```
mvn package
```

A simple fix for this is to run

```
mvn install
```

instead. If you have to do this, it will probably only be a one time thing.

Copyright © 2000-2025 The Apache Software Foundation (<https://www.apache.org/>). Apache Struts, Struts, Apache, the Apache feather logo, and the Apache Struts project logos are trademarks of The Apache Software Foundation. All Rights Reserved.

Logo and website design donated by SoftwareMill (<https://softwaremill.com/>).

Follow @x (https://twitter.com/x?ref_src=twsrc%5Etfw)