



JAVA TUTORIAL

#INDEX POSTS

#INTERVIEW QUESTIONS

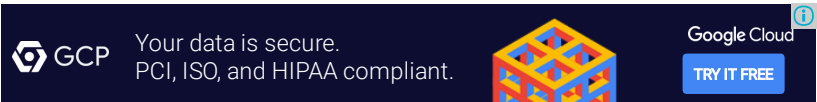
RESOURCES

STORE

HOME » MAVEN » TOMCAT7-MAVEN-PLUGIN – TOMCAT MAVEN PLUGIN TO DEPLOY WAR

tomcat7-maven-plugin – Tomcat Maven Plugin to Deploy WAR

JULY 19, 2016 BY PANKAJ — 8 COMMENTS



Today we will look into Tomcat Maven Plugin (tomcat7-maven-plugin) that helps us in deploying our WAR files to Tomcat easily.

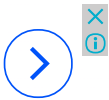
Tomcat Maven Deploy

Recently I was developing a web application and after doing the build, I used to copy the generated WAR file to Tomcat webapps directory. It was time consuming, boring and sometimes I used to miss the updated WAR file to copy and notice it later on.

Then I decided to look for some configurations through which I can configure Maven build to deploy the generated WAR file into tomcat automatically.

Devcloud - Application Hosting - Host Apps On Our Cloud Servers

Test Your New Application On Our Cloud Servers. 10 Day Free Trial. Sign Up Now!



My first approach was to use Maven plugins to copy the generated WAR file to tomcat deploy directory.

For example, below was the pom.xml file for my simple web application, I have removed all the unwanted stuffs to keep it clear.

```
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>MyWebapp</groupId>
  <artifactId>MyWebapp</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>war</packaging>
  <build>
    <sourceDirectory>src</sourceDirectory>
    <plugins>
      <plugin>

<artifactId>maven-war-plugin</artifactId>
<version>2.3</version>
```

```
<warSourceDirectory>WebContent</warSourceDirectory>

<failOnMissingWebXml>>false</failOnMissingWebXml>
    </configuration>
  </plugin>
  <plugin>
    <artifactId>maven-compiler-
plugin</artifactId>
    <version>3.1</version>
    <configuration>
      <source>1.7</source>
      <target>1.7</target>
    </configuration>
  </plugin>
</plugins>
</build>
</project>
```

I used `maven-antrun-plugin` to add another task for the `integration-test` phase where I am copying the WAR file to my local tomcat directory.



```
    <artifactId>maven-compiler-
plugin</artifactId>
    <version>3.1</version>
    <configuration>
      <source>1.7</source>
      <target>1.7</target>
    </configuration>
  </plugin>

  <plugin>
    <artifactId>maven-antrun-
plugin</artifactId>
    <executions>
      <execution>
        <phase>integration-
test</phase>
        <configuration>
          <tasks>
            <copy
file="target/${pom.build.finalName}.war"
tofile="${deployFolder}/${pom.build.finalName}.war" />
          </tasks>
        </configuration>
      </execution>
    </executions>
  </plugin>
```

```
        </goals>
      </execution>
    </executions>
  </plugin>
</plugins>
<finalName>${project.artifactId}</finalName>
</build>
</project>
```

The change was really simple, create a property to define the tomcat deploy directory and add **finalName** configuration to skip version details from the WAR file.

mvn clean integration-test package – it will generate the WAR file and deploy into tomcat too.

mvn clean package – It will just build the project, will not try to deploy it into tomcat container.

1. How to Build a Website	▶
2. Create a Website	▶
3. Cloud Hosting Services	▶
4. How to Make a Web Page	▶
5. Apache Tomcat Server	▶

Above approach solved by problem, but there were two shortcomings;

- 1. Since tomcat deploy folder is hardcoded in the pom.xml, if I change tomcat server location then I would have to change it here too. Also because it's my user configuration, I can't check in this code into repository.
- 2. I can't use this approach to deploy the application into any remote server.

Tomcat7 Maven Plugin



Because of above shortcomings, I looked online for better solutions and found that there is a plugin already for this job – **Tomcat Maven Plugin**.

We can use this plugin with some configuration settings to deploy our application into Tomcat server using their management console API.

{Flash Sale} - Udemy Courses at Lowest Price of \$10 or INR 640 - Only for 6 Hours

Check it Out Now

Before we can use Apache Tomcat Maven Plugin, we need to perform some configuration settings. Let's look at these configurations one by one. Note that I am using Tomcat-7, if you are using any other version then some configurations might need to be changed accordingly.

1. **Adding Manager Roles and User in Tomcat 7:** We need to add `manager-gui` and `manager-script` roles in `tomcat-users.xml` file and create a user with these roles. Just add below lines to your `tomcat-users.xml` file and restart the server.

`TOMCAT-HOME/conf/tomcat-users.xml`



```
<role rolename="manager-gui" />
<role rolename="manager-script" />
<user username="tomcat" password="tomcat" roles="manager-gui,manager-script" />
```

2. **Adding server to Maven settings.xml file:** Next step is to add a server in maven settings.xml file, as shown below.

`$HOME/.m2/settings.xml`

```
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0
    http://maven.apache.org/xsd/settings-1.0.0.xsd">
  <servers>
    <server>
      <id>Tomcat</id>
      <username>tomcat</username>
      <password>tomcat</password>
    </server>
  </servers>
</settings>
```

Notice that username and password should be same as configured in the earlier step.

tomcat7-maven-plugin example

3. **Maven Project pom.xml changes:** Final step is to configure Tomcat Maven Plugin in the web application `pom.xml` file. Just add below lines to your `pom.xml` plugins section.

```
<plugin>
  <groupId>org.apache.tomcat.maven</groupId>
  <artifactId>tomcat7-maven-plugin</artifactId>
  <version>2.2</version>
```

Shares

f

G+



{Flash Sale} - Udemy Courses at Lowest Price of \$10 or INR 640 - Only for 6 Hours

Check it Out Now

```
<server>Tomcat</server>

<path>/MyWebapp</path>

</configuration>

</plugin>
```

tomcat6-maven-plugin example

If you are using Tomcat-6, below configurations should be used instead.

```
<plugin>

  <groupId>org.apache.tomcat.maven</groupId>
  <artifactId>tomcat6-maven-plugin</artifactId>
  <version>2.2</version>
  <configuration>
    <url>http://localhost:9090/manager</url>
    <server>Tomcat</server>
    <path>/MyWebapp</path>
  </configuration>
</plugin>
```

That's it, now we can use below commands to deploy, undeploy, redeploy our application through maven command.

Tomcat 7 Maven Plugin Deploy Commands

- mvn tomcat7:deploy
- mvn tomcat7:undeploy
- mvn tomcat7:redploy

Tomcat 6 Maven Plugin Deploy Commands

For Tomcat-6, you should use below commands.

- mvn tomcat6:deploy
- mvn tomcat6:undeploy
- mvn tomcat6:redploy

Below is the sample output produced by the deploy command for tomcat7-maven-plugin.

```
pankaj:MyWebapp pankaj$ mvn tomcat7:deploy
[INFO] Scanning for projects...
[INFO]
[INFO] -----
-----
[INFO] Building MyWebapp 0.0.1-SNAPSHOT
[INFO] -----
-----
[INFO]
[INFO] >>> tomcat7-maven-plugin:2.2:deploy (default-cli) @ MyWebapp
>>>
[INFO]
[INFO] --- maven-resources-plugin:2.5:resources (default-resources) @
MyWebapp ---
[debug] execute contextualize
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered
```

{Flash Sale} - Udemy Courses at Lowest Price of \$10 or INR 640 - Only for 6 Hours

Check it Out Now

```
/Users/pankaj/Documents/workspace/j2ee/MyWebapp/src/main/resources
[INFO]
[INFO] --- maven-compiler-plugin:3.1:compile (default-compile) @
MyWebapp ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-resources-plugin:2.5:testResources (default-
testResources) @ MyWebapp ---
[debug] execute contextualize
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered
resources, i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory
/Users/pankaj/Documents/workspace/j2ee/MyWebapp/src/test/resources
[INFO]
[INFO] --- maven-compiler-plugin:3.1:testCompile (default-testCompile)
```

You can go to your Tomcat Manager App and confirm that application is deployed or you can directly access the application in the browser to confirm it. That's all for tomcat maven plugin and tomcat7-maven-plugin example.

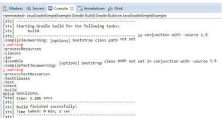
Powered by Google



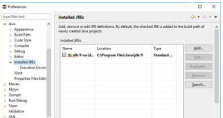
Google Cloud Platform.



Gradle vs Maven



Gradle Tutorial



Java SE 9: Develop and Test...

(Ad) Google Cloud P...

journaldev.com

journaldev.com

journaldev.com



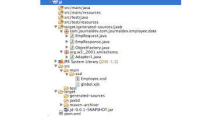
Java 9 Modules

journaldev.com



Spring REST Example Tutorial, Spring Restful...

journaldev.com



jaxb2-maven-plugin XJC example to...

journaldev.com



How to generate XSD from Java Class

journaldev.com

FILED UNDER: MAVEN

About Pankaj

If you have come this far, it means that you liked what you are reading. Why not reach little more and connect with me directly on **Google Plus**, **Facebook** or **Twitter**. I would love to hear your thoughts and opinions on my articles directly.
Recently I started creating video tutorials too, so do check out my videos on **Youtube**.

<< jQuery dequeue()

jQuery AJAX JSP Servlet Java Example >>

Comments

fred says
MAY 20, 2016 AT 7:01 AM

{Flash Sale} - Udemy Courses at Lowest Price of \$10 or INR 640 - Only for 6 Hours

Check it Out Now

org.apache.tomcat.maven
tomcat7-maven-plugin
2.2

[https://localhost:8443/manager/text](https://localhost:8443/manager/text/$[project.build.finalName])
[/\\$\[project.build.finalName\]](#)

Reply

SARVESH says

FEBRUARY 2, 2016 AT 11:14 PM

I am following the same for deployment. But my environment is Ubuntu.
I made changes in /etc/maven/settings.xml and added user in tomcat-users.xml.
But when I fire mvn tomcat7:deploy from terminal, I get error following error.

[[INFO] -----
[[INFO] BUILD FAILURE
[[INFO] -----
[[INFO] Total time: 2.859s
[[INFO] Finished at: Wed Feb 03 12:43:29 IST 2016
[[INFO] Final Memory: 9M/108M
[[INFO] -----
[[ERROR] No plugin found for prefix 'tomcat7' in the current project and in the plugin groups
[org.apache.maven.plugins, org.codehaus.mojo] available from the repositories [local
(/root/.m2/repository), central (<http://repo.maven.apache.org/maven2>)] -> [Help 1]
org.apache.maven.plugin.prefix.NoPluginFoundForPrefixException: No plugin found for prefix 'tomcat7' in
the current project and in the plugin groups [org.apache.maven.plugins, org.codehaus.mojo] available
from the repositories [local (/root/.m2/repository), central (<http://repo.maven.apache.org/maven2>)]

Reply

ghost_rider says

JANUARY 26, 2016 AT 3:34 PM

I have errors, when i try did like in tutorial. When i run tomcat manual: "Cannot invoke Tomcat manager:
Software caused connection abort" and when i didn't run it manual: "Cannot invoke Tomcat manager:
Connection refused: connect"

Reply

Raghu says

JANUARY 23, 2016 AT 10:32 PM

If I want to deploy on a remote machine how to deploy? How to know where to deploy? In other words,
how to make sure CATALINA service is running on remote machine and that how to make the automation
(in script) to confirm whether the jar/war can be deployed?

Reply

{Flash Sale} - Udemy Courses at Lowest Price of \$10 or INR 640 - Only for 6 Hours

Check it Out Now

Hitesh Patel says

NOVEMBER 1, 2015 AT 5:46 PM

Thanks a lot. It's working for me as you suggested.

Reply

Arasu says

APRIL 15, 2015 AT 1:32 AM

Dear Pankaj,

First i would like to thank for all your awesome examples..And then Can you briefly explain What is maven? Where it's used?

Thanks in advance.

Reply

Marija says

DECEMBER 18, 2014 AT 4:18 AM

Thank you very much! Very helpful tutorial!

Reply

Khan says

DECEMBER 1, 2014 AT 11:05 PM

Hey Pankaj , do you have detailed explination about maven, where to use?why and when and how ?

Reply

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

{Flash Sale} - Udemy Courses at Lowest Price of \$10 or INR 640 - Only for 6 Hours

Check it Out Now

Email *

Website

POST COMMENT

Google Custom Search

SEARCH

×

AdChoices



**Build the websites.
Automate the
infrastructure.**

FREE HOSTING ACCOUNT >



PANTHEON[®]
Website Management Platform

AdChoices



**UNLIMITED 24X7
SUPPORT FOR DRUPAL
& WORDPRESS SITES.**

Try Panttheon for Free!



PANTHEON[®]
Website Management Platform

STAY UPDATED!

Name

E-Mail Address

I AM IN!

RECOMMENDED TUTORIALS

Java Tutorials

> [Java IO](#)

{Flash Sale} - Udemy Courses at Lowest Price of \$10 or INR 640 - Only for 6 Hours

Check it Out Now

- > [Multithreading in Java](#)
- > [Java Logging](#)
- > [Java Annotations](#)
- > [Java XML](#)
- > [Collections in Java](#)
- > [Java Generics](#)
- > [Exception Handling in Java](#)
- > [Java Reflection](#)
- > [Java Design Patterns](#)
- > [JDBC Tutorial](#)

Java EE

- > [Servlet JSP Tutorial](#)
- > [Struts2 Tutorial](#)
- > [Spring Tutorial](#)
- > [Hibernate Tutorial](#)
- > [Primefaces Tutorial](#)
- > [Apache Axis 2](#)
- > [JAX-RS](#)
- > [Memcached Tutorial](#)