

MavenBeehivePlugin

What's the Maven Plugin do?

The Beehive Maven plugin does several things for you:

- No more bothering with ant build scripts.
- Runs APT over your controls producing the Beehive generated classes (for your main source code and tests).
- Helps you generate WARs for your application

Installing the Plugin

The plugin can be installed like so:

```
$ maven plugin:download -Dmaven.repo.remote=http://dist.codehaus.org \
-DartifactId=beehive-maven-plugin -DgroupId=controlhaus -Dversion=1.0-SNAPSHOT
```

After that you should be ready to go!

Setting up your Maven Project

To set up your Maven project to use the plugin, you just add a couple lines to your project.properties file telling Maven to use Java 5 and the plugin you just installed.

```
maven.compile.source=1.5
maven.compile.target=1.5

beehive.compile.enable=true
```

Building and Testing Your Application

You can build and test your application with the normal maven commands, "maven jar" and "maven test".

Building and Testing Webapps

The plugin attempts to leverage the work that the WAR plugin has done to make building webapps easy. Please read the [Maven War plugin documentation](#) on how to set up Maven for web applications.

Currently there is only support for the "war:inplace" command (the rest are coming soon!). When the Beehive Maven plugin is installed, war:inplace will process your JPF files and build the app. Then just add your webapp directory to Tomcat to be loaded. To add your application to Tomcat create a file in TOMCAT_HOME/conf/Catalina/localhost called "yourapp.xml". The file contents should contain something like:

```
<?xml version='1.0' encoding='utf-8'?>
<Context docBase="c:/eclipse/workspace/ch-blog-demo/resources" path="/demo" reloadable="true">
</Context>
```