Gradle

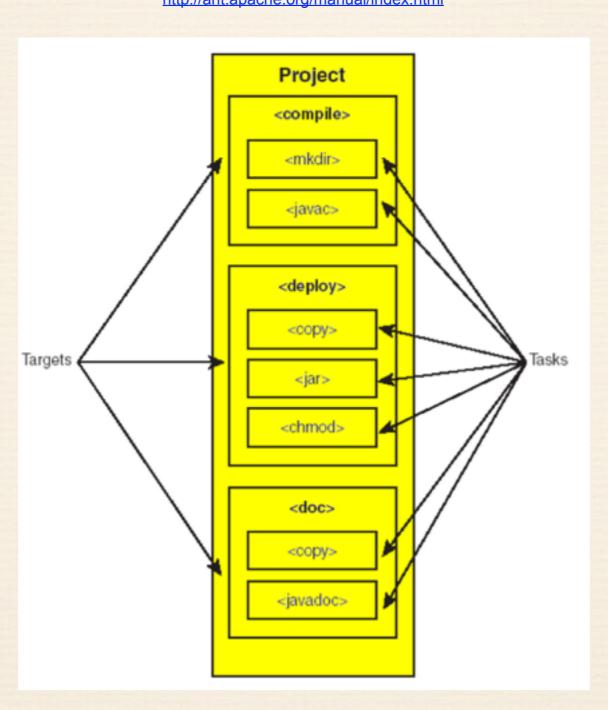


### Build

- \* clean
- \* javac
- \* test (java org.junit.runner.JUnitCore [test class name])
- packaging (jar, war, ear)
- deploy

## Ant

http://ant.apache.org/manual/index.html



# Ant Example

```
<description>
 simple example build file
</description>
<!-- set global properties for this build -->
property name="src" location="src"/>
cproperty name="build" location="build"/>
cproperty name="dist" location="dist"/>
 <target name="init">
  <!-- Create the time stamp -->
  <tstamp/>
  <!-- Create the build directory structure used by compile -->
  <mkdir dir="${build}"/>
 </target>
<target name="compile" depends="init"
   description="compile the source">
  <!-- Compile the java code from ${src} into ${build} -->
  <javac srcdir="${src}" destdir="${build}"/>
 </target>
<target name="dist" depends="compile"
    description="generate the distribution">
  <!-- Create the distribution directory -->
  <mkdir dir="${dist}/lib"/>
  <!-- Put everything in ${build} into the MyProject-${DSTAMP}.jar file -->
 <jar jarfile="${dist}/lib/MyProject-${DSTAMP}.jar" basedir="${build}"/>
 </target>
 <target name="clean"
   description="clean up">
  <!-- Delete the ${build} and ${dist} directory trees -->
 <delete dir="${build}"/>
  <delete dir="${dist}"/>
</target>
</project>
```

- mvn [options] [<goal(s)>] [<phase(s)>]
- http://maven.apache.org/guides/getting-started/maven-in-five-minutes.html
- http://maven.apache.org/guides/introduction/introduction-to-thelifecycle.html

### Clean Lifecyle

pre-clean

#### clean

post-clean

Default Lifecyle	
validate	test-compile
initialize	process-test-classes
generate-sources	test
process-sources	prepare-package
generate-resources	package
process-resources	pre-integration-test
compile	integration-test
process-classes	post-integration-test
generate-test-sources	verify
process-test-sources	install
generate-test-resources	deploy
processs-test-resources	

### Site Lifecyle

pre-site

site

post-site

site-deploy

Goals Phases resources:resources process-resources compiler:compile compile process-classes process-test-resources resources:testResources test-compile compiler:testCompile surefire:test test prepare-package jar:jar package Note: There are more phases than shown above, this is a partial list

- <dependency>
  - \* scope
    - \* compile
    - \* provided
    - \* runtime
    - \* test
    - system

### Gradle

- Groovy Based (Language)
- \* Easy Extension
- Wrapper (Remove Gradle Dependency)
- \* Support Maven, Ant, Ivy

# Java Plugin

compileJava classes compileTestJava testClasses processTestResources build uploadArchives jar assemble