

CS320 Web and Internet Programming

Introduction to Ant

Chengyu Sun
California State University, Los Angeles

Build

- ◆ Compilation
- ◆ Maintenance
- ◆ Deployment

Background

- ◆ A build tool for Java
- ◆ Developed by James Duncan Davidson, who also wrote Tomcat
- ◆ Ant stands for
 - Another Neat Tool (ANT), or
 - Ant, the small creature who can *build* big things

Why Ant

- ◆ vs. `javac *.java`
 - Re-compile the modified files
 - Re-compile the files affected by the modified files (Dependency)
- ◆ vs. scripts
 - Easier to use
 - Platform independent

Why Not Make

- ◆ Compiler is invoked once for every source file
 - Fine for C/C++ compilers
 - Bad for Java compiler – JVM startup overhead
- ◆ Platform dependent
 - `make`, `gmake`, `nmake` ...
 - relies on shell command for most tasks
- ◆ Syntax problem
 - Tab

Ant installation

- ◆ Download from <http://ant.apache.org>
- ◆ Unzip to a local directory
- ◆ Environment variables
 - `JAVA_HOME`: JDK directory
 - `ANT_HOME`: Ant directory
 - Add `$ANT_HOME/bin` (`%ANT_HOME%/bin`) to `PATH`

Ant Basics

- ◆ Build file
- ◆ Property
- ◆ Target
- ◆ Task
- ◆ Type

Build File

- ◆ build.xml
- ◆ An XML file
- ◆ <project>
 - n *name* – name of the project
 - n *basedir* – the directory where the project is located

Properties

- ◆ <property>
 - n name
 - n value
- ```
<property name="tomcat.home" value="d:\tomcat"/>
```
- ```
<fileset dir="${tomcat.home}">  
  <include name="common/lib/**/*.*jar"/>  
  <include name="shared/lib/**/*.*jar"/>  
  <include name="server/lib/**/*.*jar"/>  
</fileset>
```

Targets and Tasks

- ◆ Targets
 - n Things we want to do
 - w compile, install, clean ...
- ◆ Tasks
 - n "Commands" that can be used to accomplish a *target*
 - n <javac>, <copy>, <move>, <depend>
 - n More portable than actual shell commands

Multiple Targets

- ◆ More targets
 - n separate source and class directories
 - n remove compiled classes
- ◆ Default target
- ◆ Target dependency

Example 1

- ◆ <target name="compile">
- ◆ <target name="init">
- ◆ <target name="clean">

Types

- ◆ Mostly related to describing files and directories
 - n <fileset>
 - n <path>

<fileset>

- ◆ A set of files
- ◆ <fileset>
 - n dir – root directory of this file set
 - n <include>
 - n <exclude>

<fileset> Examples

```
<fileset dir="d:/tomcat/common/lib"/>
<fileset dir="d:/tomcat/common/lib">
  <include name="*.jar"/>
</fileset>

<fileset dir="d:/tomcat/common/lib">
  <include name="**/*.jar"/>
</fileset>

<fileset dir="d:/tomcat/common/lib">
  <include name="**/*.jar"/>
  <exclude name="**/naming*.jar"/>
</fileset>
```

<path>

- ◆ A list of files and directories
- ◆ <path>
 - n files: <fileset>, <filelist>
 - n directories: <dirset>
 - n other paths: <pathelement>, <path>
- ◆ <classpath>

Example 2

- ◆ Specify a classpath that include
 - n Java class libraries
 - n Tomcat class libraries
 - w common/lib/*.jar
 - w shared/lib/*.jar
 - w server/lib/*.jar
 - n Additional class libraries
 - w lib/*.jar

Using Ant for Your Projects

- ◆ You don't have to be an Ant expert
- ◆ Start with a sample build file
 - n Understand it
 - n Adapt it
- ◆ Ant Manual
 - n <http://ant.apache.org/manual/index.html>