

AntTasks

Ant Tasks

There is a great deal of "folklore" knowledge about the individual Ant tasks that is not clear from the specification of the tasks offered in the [Ant Manual](#).

The Sync Task (present since 1.6)

Here is an example of some lore of this type: The Sync task will delete 'dangling files' and 'dangling directories' in the target directory.

The Property Task

<http://ant.apache.org/manual/CoreTasks/property.html>

The Prefix Attribute

The Problem:

I have to access some properties (from a file), which are already defined in my build file. To be able to access the properties I can use the `prefix` attribute of the property task.

`framework.properties`

```
#Release information about the framework
#Thu Oct 14 16:25:12 CEST 2004
build.number=115
release.version=0.4
release.name=framework
```

Ant Example Target

```
<target name="read.properties">
  <!-- Read the properties from the release of the framework -->
  <property file="framework.properties" prefix="framework"/>
  <echo message="${framework.build.number}"/>
  <echo message="${framework.release.version}"/>
  <echo message="${framework.release.name}"/>
</target>
```

Output

```
Buildfile: C:\build.xml
read.properties:
[echo] 115
[echo] 0.4
[echo] framework
BUILD SUCCESSFUL
Total time: 3 seconds
```

The AntCall Task

<http://ant.apache.org/manual/CoreTasks/antcall.html>

Sometimes on the mailinglists there is a question on how to invoke a special target depending on the **value** of a property. Because there could be more than two values, you can't use the usual if/unless-attributes of the target.

But the solution is quite easy: compound the name of the target by a common term and the value itself and let antcall do the job.

Ant Example Target

```

<project default="call">
  <target name="foo-upload"><echo>FOO upload</echo></target>
  <target name="bar-upload"><echo>BAR upload</echo></target>
  <target name="test-upload"><echo>TEST upload</echo></target>

  <target name="call">
    <input message="Please choose the server "
      validargs="foo,bar,test"
      addproperty="server"
      defaultvalue="test"
    />
    <antcall target="${server}-upload" />
  </target>
</project>

```

You have only to ensure, that there are valid values for that property, e.g. a *blabla* would call the non existing *blabla-upload* which results in an error. But the input task here will ensure that.

BTW - the same can be used for loading platform specific properties:

```

<property file="${os.name}.properties" />
<property file="default.properties" />

```

The Exec Task

On Windows the Exec task does not find the executable file if its directory was double quoted in the PATH environment variable. (Execute failed ... error 2: which means file not found.)

The Unzip Task

How to unzip the content of a subdirectory of a zip file without the path to the subdirectory. For example, if a zip file has a directory dir1/dir2, how to unzip dir2 into dir3 so that dir3 contains only dir2, not dir1/dir2?

The XSLT Task

This task allows you to perform XML transforms using the built-in XSLT system. XSLT 2.0 can also be done using the Saxon system.

Here is a demonstration [UsingAntWithXSLT20AndSaxon](#).

[Category](#)[Category](#)