

External tasks

by Ovidiu Predescu, Jeff Turner

NOTICE: Copyright © 2002-2003 Ovidiu Predescu and Jeff Turner. All rights reserved.
The Anteater manual may be reproduced and distributed in whole or in part, in any medium, physical or electronic, so long as this copyright notice remains intact and unchanged on all copies.

1. External tasks

Anteater provides few more tasks which do not fit into any of the categories above. They are used to provide additional functionality, useful when writing tests.

Some of these tasks are provided by [Ant Contrib](#) project, distributed with Anteater.

The following tasks are available to Anteater:

- [foreach](#) - iterate over a collection of values

1.1. foreach

See the [ant-contrib](#) documentation for info about this task.

1.2. jelly

Many Anteater tasks can have nested content. For example, the [contentEquals](#) element can take nested text, which will constitute the HTTP body:

```
<contentEquals>This is the HTTP body</contentEquals>
```

Anteater provides a way to dynamically generate this body text, using [Jelly](#) scripts. One simply replaces the static text with a `<jelly>` element:

```
<contentEquals>
  <jelly script="generateResponse.jelly"/>
</contentEquals>
```

Anteater will run the Jelly script, and use its output as the nested content.

By default, Anteater does not come with all the jars required to use Jelly. To jelly-enable an Anteater installation, run the following command:

```
[UNIX]    anteater -f $ANTEATER_HOME/resources/scripts/install-jelly.xml
[Windows] anteater -f %ANTEATER_HOME%\resources\scripts\install-jelly.xml
```

Or alternatively, run the following Anteater script:

```
<project name="jelly-install" default="main" basedir=".">
  <target name="main">
    <ant dir="{anteater.resources}/scripts" antfile="install-jelly.xml"/>
  </target>
</project>
```

Where ANTEATER_HOME is where you installed Anteater.

You will be prompted to enter the URL of a jar repository. If you just press enter, the default will be used, and the install is automatic from there on. Alternatively, if you have a [Maven](#) repository locally, you can try pointing the script at this.

Attribute name	Type	Default value	Description
script	string		File path a Jelly script to run.
url	string		URL of a Jelly script to run.
output	string		Specifies the path to a file in which to store the script results.

Table 1: Attributes

Elements allowed inside jelly: none