

# Using the Custom Ant Tasks

Ant is a scripting language you can use to automate the building process for Java code. Ant is written in XML. Within an Ant script, you define a *target*. Each target contains a set of *tasks*. Some of the standard Ant tasks include:

- Get latest version from source control
- Compile the source code
- Create jar (Java archive) files

For example:

```
<Project name="sample project">
  <target name="build">
    <cvsversion ... > <!--get latest version &#61664;
    <javac ... > <!--compile the source code &#61664;
  </target>
</Project>
```

Natural Business Services includes several custom Ant tasks. These tasks help generate and deploy Java classes based on business services created on the server.

---

## Prerequisites

The custom Ant tasks provided with Natural Business Services require the following prerequisites:

- Ant 1.6
- The following .jar files on the class path:
  - junit-4.3.1.jar
  - entirex.jar
  - nbsrt4j.jar
  - ndvserveraccess.jar
  - nbsAnt.jar
  - velocity-dep-1.4.jar
  - nbs.ui.jar
- CentraSite .jar file(s)

This file(s) is only required if deploying to CentraSite.

## Global Properties

The custom Ant tasks use the following global properties. These properties must be set using the **Property** node in Ant:

Property	Value
nbs.userID	User ID to connect to the Natural Business Services (NBS) server.
nbs.connectionID	Connection ID (located in the <i>DispatchClient.Config</i> file).
nbs.password	Password for the specified user ID.
nbs.config.location	Location of the <i>DispatchClient.Config</i> file.
nbs.target.vm	Java Virtual Machine (VM) version to target (original or enhanced).
nbs.templates	Location of the template (*.vm) files used by Natural Business Services (for example: <install dir>/templates).

## Supplied Ant Tasks

This section describes the Ant tasks supplied with Natural Business Services. All tasks are located in the com.softwareag.nbs.ant.tasks package.

The supplied Ant tasks are:

- Axis2Deploy
- CentraSiteRegister
- CreateAAR
- CreateClass
- PrintConfig
- Regenerate

### Axis2Deploy

This task deploys an Axis2 Web service archive file (.aar) to an Axis2 web application server. The attributes for this task are:

Attribute	Optional	Description
aarfilename	No	Archive file to deploy.
targetfolder	No	Name of the folder in which to copy the .aar file.

## CentraSiteRegister

This task registers a Web service with CentraSite. The attributes for this task are:

Attribute	Optional	Description
description	No	Brief description of the Web service to register.
organization	No	Name of the CentraSite organization.
server	No	URL for the CentraSite server (for example: <i>http://&lt;hostname&gt;:53305/CentraSite/CentraSite</i> ).
userid	No	User ID for CentraSite.
password	No	Password for CentraSite.
wsdl	No	URL for the WSDL (Web Service Description Language) file for the Web service.

## CreateAAR

This task creates a Web service archive file (.aar) for deployment to Axis2. The attributes for this task are:

Attribute	Optional	Description
destfile	No	Path for the .aar file to create.
servicestemplate	No	Path for the .xml template to use ( <i>services.xml.vm</i> ).

## Child Nodes

Child Node	Description and Attributes
fileset	Standard Ant fileset node. Each fileset node is used to include files in the archive. The <code>dir</code> attribute must be specified.  Any files found by the fileset will be added to the .aar file with a path relative to the <code>dir</code> attribute. For example, if the <code>dir</code> attribute is <i>c:\temp</i> and a file is found at <i>c:\temp\A\b\c.class</i> , the archive will contain <i>A\b\c.class</i> , with <i>A</i> and <i>B</i> as folders.
webservice	Web service entry to add to the <i>services.xml</i> file.

## CreateClass

This task creates a Java class based on a business service on the server.

### Note:

If the `webservice` attribute is set to true, the CreateClass task will not generate the .aar file because the Java code has not yet been compiled. Use the CreateAAR task instead.

The attributes for this task are:

Attribute	Optional	Description
axisversion	Yes	Version of Axis files to generate. The default is "Axis" (i.e., not Axis2).
destdir	Yes	Path for the directory in which to store the generated files.
metadir	Yes	Path for the directory in which to store the specifications ( <i>.nbsMetadata</i> file).
domain/servicename/version	Yes	Specifications for the business service.
junit	Yes	Indicates whether to generate Junit test(s). The default is "false".
package	Yes	Java package to use for the generated class.
webservice	Yes	Indicates whether to generate files to expose the generated class as a Web service. The default is "false".
failonerror	Yes	Indicates whether the CreateClass task will pass or fail when an error occurs while creating the client proxy class. The default is "true". <ul style="list-style-type: none"> <li>● If this attribute is set to "false" and an error occurs, the task will continue.</li> <li>● If this attribute is set to "true" and an error occurs, the task will end.</li> </ul>
multithread	Yes	Indicates whether to use multiple threads to create the classes. The default is "true"; for each service found on the server, a separate thread will be created to download metadata for the business service and create the class.  The advantage to this option is that if the user has a slow connection to the server, several calls can be made at once to download metadata for different services.  To disallow this option, set this attribute to "false".

### Child Nodes

Child Node	Description and Attributes
serviceset	List containing business services to generate.
serverchange	List of business services based on server activity.

## PrintConfig

This task prints a copy of the configuration settings used by Natural Business Services. The following example shows output from this task:

```
[nbs.printconfig] Configuration
[nbs.printconfig] Configuration File: S:/NBS52/Runtime/DispatchClient.config
[nbs.printconfig] ConnectionID: 52-NEW-DEV
[nbs.printconfig] User ID: DEV
[nbs.printconfig] Password: ****
[nbs.printconfig] Templates: S:\NBS52\Java\Workspace3.3\com.softwareag.nbs.ui\templates
```

## Regenerate

This task regenerates files created by the CreateClass task. The attributes for this task are:

Attribute	Optional	Description
destdir	Yes	Path to the directory containing the files to regenerate.
metadir	Yes	Path to the directory containing the specifications (folder in which the <i>.nbsMetadata</i> files are located).

## Child Nodes

Child Node	Description and Attributes
fileset	List of .java or .nbsMetadata files to regenerate. If the file is .java, the Regenerate task searches for the corresponding .nbsMetadata file in the directory specified in the metadir attribute.

## Child Node Types

This section describes the Child node types for the Ant tasks supplied with Natural Business Services. All node types are located in the com.softwareag.nbs.ant package.

The Child node types are:

- ServerChange
- ServiceSet
- WebService

## ServerChange

This task uses filtering criteria (such as domain, user ID, and date) to search the Natural Business Services log file on the server and find any changes to business services.

**Note:**

All purge changes are ignored because you cannot generate a class for a business service that has been purged.

The attributes for this task are:

Attribute	Optional	Description
domain	Yes	Limits the list of services to a particular domain.
duration	Yes	<p>Finds changes made during the specified interval. The duration format is:</p> <p>[count][interval]</p> <p>Valid intervals are:</p> <ul style="list-style-type: none"><li>● H (hour)</li><li>● M (minute)</li><li>● d (day)</li><li>● w (week)</li><li>● m (month)</li><li>● y (year)</li></ul> <p>For example, "2d" indicates changes made in the past two days. The default is "1m" (1 month).</p>
userid	Yes	Lists the changes made by the specified user.

**ServiceSet**

This task lists the business services available for the specified connection ID and domain. The attributes for this task are:

Attribute	Optional	Description
connectionid	Yes	Connection ID for which to list the services. If this attribute is not specified, the nbs.connectionID global property will be used.
domain	No	Natural Business Services domain containing the specified services.
includepattern	Yes	Regular expression pattern used to identify a business service name. If a name matches this pattern, it will be included.
excludepattern	Yes	Regular expression pattern used to identify a business service name. If a name matches this pattern, it will be excluded.

**Note:**

If the includepattern and excludepattern attributes are both specified, includepattern will be applied before excludepattern.

**WebService**

This task specifies an Axis Web service. The attributes for this task are:

Attribute	Optional	Description
name	No	Name of the Web service.
classfullname	No	Full class name (including package) of the class for the Web service.

**Example of Using Custom Ant Tasks**

The following XML example demonstrates the use of custom Ant tasks supplied with Natural Business Services:

```
<?xml version="1.0" ?>
- <project name="Test" default="build" basedir=".>
- <!--
TODO: change the below path to match your nbs installation folder
-->
<property name="nbs.install" value="C:\Program Files\Software AG\Natural Business Services\V5.3" />
- <!--
TODO: Modify the below settings to match your configuration
-->
- <target name="init">
<property name="nbs.userID" value="GUEST" />
<property name="nbs.connectionID" value="Default Broker" />
<property name="nbs.password" value="" />
<property name="nbs.config.location" value="${nbs.install}\dispatchclient.config" />
<property name="nbs.target.jvm" value="1.4" />
<property name="nbs.templates" value="${nbs.install}\templates\java" />
</target>
- <!--
jars to be used while ant is running
-->
- <path id="runtimeJars">
- <fileset dir="${nbs.install}\javaruntime">
<include name="*.jar" />
</fileset>
- <fileset dir="${nbs.install}\EclipsePlugin\plugins">
<include name="com.softwareag.nbs.ui_*.jar" />
</fileset>
</path>
- <!--
jars to be used by the compiler
-->
- <path id="compileJars">
- <fileset dir="${nbs.install}\javaruntime">
<include name="*.jar" />
</fileset>
</path>
- <!--
NBS Task definitions
-->
<taskdef classname="com.softwareag.nbs.ant.tasks.Test" name="nbstest" classpathref="runtimeJars" loaderref="NBSRef" />
<taskdef classname="com.softwareag.nbs.ant.tasks.CreateClass" classpathref="runtimeJars" name="nbs.createclass" loaderref="NBSRef" />
<taskdef classname="com.softwareag.nbs.ant.tasks.Regenerate" name="nbs.regenerate" classpathref="runtimeJars" loaderref="NBSRef" />
<taskdef classname="com.softwareag.nbs.ant.tasks.CreateAAR" name="nbs.createaar" classpathref="runtimeJars" loaderref="NBSRef" />
<taskdef classname="com.softwareag.nbs.ant.tasks.Axis2Deploy" name="nbs.axis2deploy" classpathref="runtimeJars" loaderref="NBSRef" />
<taskdef classname="com.softwareag.nbs.ant.tasks.CentraSiteRegister" name="nbs.centrasite" classpathref="runtimeJars" loaderref="NBSRef" />
```

```

- <!--
NBS Type definitions
-->
<typedef name="serviceset" classname="com.softwareag.nbs.ant.ServiceSet" classpathref="runtimeJars" loaderref="NBSRef" />
<typedef name="serverchange" classname="com.softwareag.nbs.ant.ServerChange" classpathref="runtimeJars" loaderref="NBSRef" />
<typedef name="webservice" classname="com.softwareag.nbs.ant.WebService" classpathref="runtimeJars" loaderref="NBSRef" />
- <target name="build" depends="init">
- <!--
Sample create class
-->
- <nbs.createclass destdir="c:\temp\root\src" metadir="c:\temp\root\.metadata" domain="DEMO" servicename="product" version="1.1.1" package="ant1.ant2" junit="true" webservice="true" axisversion="2">
  <serviceset domain="DEMO" includePattern="Calc.**" />
  <serverchange duration="1y" userid="PMRUSR" domain="DEMO" />
</nbs.createclass>
- <!--
Sample regenerate
-->
- <nbs.regenerate destdir="c:\temp\root\src" metadir="c:\temp\root\.metadata">
- <fileset dir="c:\temp\root\src" casesensitive="false">
  <include name="**/*.java" />
</fileset>
</nbs.regenerate>
- <!--
Compile the above generated classes
-->
- <mkdir dir="c:\temp\bin" />
<javac executable="C:\Program Files\Java\jdk1.5.0_11\bin\javac.exe" fork="true" classpathref="compileJars" compiler="javac1.5" destdir="c:\temp\bin" srcdir="c:\temp\root\src" />
</target>
</project>

```