

Технологии Java

Apache Ant

<http://kgeorgiy.info/courses/java-advanced/>

Содержание

1. Введение
2. Задания
3. Цели
4. Проекты
5. Свойства
6. Множества файлов
7. Пути
8. Заключение

Часть 1

Введение

Apache Ant

- Язык описания build-файлов
- Основан на XML
- Межплатформенный
- Open Source (Apache License)

- Сайт <http://ant.apache.org>

Основные понятия

- Строительные блоки
 - Задание
 - Цель
 - Проект
- Концепции
 - Свойства
 - Пути
 - Множества файлов

Часть 2

Задания

Задание (task)

- Элементарная операция build-файла
 - Тип задания
 - Аргументы задания
 - Задание завершается успешно либо неуспешно
- Примеры
 - Откомпилировать все `.java` файлы в каталоге `sources` и поместить результат в каталог `classes`
 - Упаковать классы из каталога `classes` в `.jar` файл

Описание заданий

- Задание задается XML-элементом
 - Тип задания – имя элемента
 - Аргументы задания – атрибуты элемента и вложенные элементы
- Пример задания
 - `<javac srcdir="source" destdir="classes"/>`
 - `<exec dir="src" executable="cmd.exe" output="dir.txt"><arg line="/c dir"/></exec>`

Компиляция

- Задание `javac` – компилирует исходный код на Java
- Атрибуты
 - `srcdir` – каталог с исходными кодами
 - `destdir` – каталог для `.class` файлов
 - `includes?` – какие файлы компилировать
 - `excludes?` – какие файлы не компилировать
 - `classpath` – classpath для компилятора
- Пример
`<javac srcdir="source" destdir="classes"/>`

Применение `includes` и `excludes`

- Атрибут `includes`
 - Список масок файлов, которые включаются в множество
 - По умолчанию – все файлы
- Атрибут `excludes`
 - Список масок файлов, которые НЕ включаются в множество
 - По умолчанию – пустое множество
- `Excludes` имеет приоритет

Маски файлов

- Специальные символы
 - ? – любой символ имени
 - * – любая последовательность символов имени
 - ** – любой каталог
- Примеры
 - Все .java файлы в текущем каталоге
`*.java`
 - Все .java файлы в текущем каталоге и подкаталогах
`**/*.java`

Создание архивов

- Задания `jar/zip` – создают `.jar/.zip` файлы
- Атрибуты
 - `destfile` – имя файла архива
 - `basedir` – корневой каталог архива
 - `update?` – обновлять/перезаписывать
 - `includes/excludes`
- Примеры
 - `<jar destfile="compress.jar" basedir="classes" includes="**/*.class"/>`
 - `<zip destfile="pics.zip" basedir="images" includes="**/*.jpg"/>`

Генерация JavaDoc

- Задание `javadoc` – компилирует JavaDoc документацию
- Атрибуты
 - `sourcepath` – где брать исходники
 - `destdir` – куда сохранять документацию
 - `packagenames` – имена пакетов
- Пример

```
<javadoc destdir="docs/api"  
    sourcepath="src/;moduleA/src"  
    packagenames= "api.*" />
```

Запуск java-программ

- Задание `java`
- Атрибуты
 - `classname` – имя основного класса
 - `classpath` – classpath
 - `input?` – имя файла для `stdin`
 - `output?` – имя файла для `stdout`
- Вложенные элементы: `arg`
- Пример

```
<java classname="test.Main"
  classpath="classes.jar">
```

Аргументы командной строки

- Элемент `arg`
- Атрибуты
 - `value?` – значение аргумента
 - `file?` – имя файла
 - `path?` – путь
 - `line?` – аргумента разделенные пробелами
- Примеры
 - `<arg value="-h"/>`
 - `<arg file="files.lst"/>`
 - `<arg path="classes:lib/ant.jar"/>`

Запуск программ

- Задача `exec` – запускает произвольную программу
- Атрибуты
 - `executable` – имя исполняемого файла
 - `dir?` – имя рабочего каталога
 - `input?` – имя файла для `stdin`
 - `output?` – имя файла для `stdout`
- Вложенные элементы: `arg`
- Пример

```
<exec dir="src" executable="cmd.exe"
  output="dir.txt">
  <arg line="/c dir"/> </exec>
```

Часть 3

Цели

Цель (target)

- Набор заданий, исполняемых последовательно
 - Зависимости между целями
 - Компиляция одного модуля зависит от другого
 - Упаковка производится после компиляции и т.п.
 - Условное выполнение
 - Выполнение не всех действий
 - Успешное / неуспешное завершение

Описание целей

- Описываются XML-элементом `target`
- Атрибуты
 - `name` – имя цели
 - `description?` – описание цели

- Пример

```
<target name="build"  
    description="Строит приложение">  
    <javac ... />  
    <jar ... />  
</target>
```

Зависимости между целями

- Атрибут `depends` – список целей, от которых зависит данная, разделенных запятыми

- Примеры

```
<target name="build" depends="init, compile, pack">  
  <echo message="Build competed"/>
```

```
</target>
```

```
<target name="build-module-C"  
  depends="build-module-A, build-module-B">
```

```
...
```

```
</target>
```

Условное выполнение

- Атрибут `if` – цель выполняется если указанное свойство установлено
- Атрибут `unless` – цель выполняется если указанное свойство НЕ установлено
- Пример

```
<target name="build-module-C-from-A"  
  if="module-A-present">
```

```
<target name="build-module-C-from-scratch"  
  unless="module-A-present">
```

```
<target name="build-module-C"  
  depends="build-module-C-from-A, build-module-C-from-  
  scratch">
```

Часть 4

Проекты

Проект

- Набор целей и задач
- При запуске
 - выполняются все задачи
 - выполняется одна или несколько целей
 - Успешное / неуспешное завершение

Описание проекта

- Описываются XML-элементом `project`
- Атрибуты
 - `name?` – имя проекта
 - `default?` – имя цели по умолчанию
 - `basedir?` – базовая директория

- Пример

```
<project name="MyFirstProject" default="all"  
  basedir=".">
```

```
...
```

```
</project>
```

Запуск Ant

- Из командной строки
 - `ant <опции> <цели>`
- Из Java
 - `java org.apache.tools.ant.Main [опции] [цели]`
 - `java org.apache.tools.ant.launch.Launcher [опции] [цели]`

Опции Ant

Основные опции	
-lib	Где искать библиотеки
-buildfile	имя build-файла
-noclasspath	Игнорировать classpath
Информация об Ant	
-help	Помощь по опциям
-version	Версия Ant
Выводимая информация	
-quiet	Мало информации
-verbose	Много информации
-debug	Отладочная информация

Организация build-файла

- Основной файл – обычно `build.xml`
- Вспомогательные файлы подключаются элементом `import`
 - Атрибут `file` – имя файла
 - Атрибут `optional` – обязателен ли

Часть 5

Свойства

Свойства

- Набор пар имя-значение
- Свойства могут использоваться в атрибутах-аргументах задач
- Пример использования свойств

```
<javac
```

```
    srcdir="${project.dir}/sources"
```

```
    destdir=" ${project.dir}/classes"
```

```
/>
```

Встроенные свойства

Свойства задаваемые Ant	
<code>basedir</code>	Полный путь к базовой директории проекта
<code>ant.file</code>	Полный путь к build-файлу
<code>ant.version</code>	Полный
<code>project.name</code>	Имя проекта
<code>ant.java.version</code>	Версия Java
Свойства задаваемые Java	
<code>java.home</code>	Путь к директории Java
<code>java.class.path</code>	Classpath
<code>os.name</code>	Название операционной системы
<code>os.version</code>	Версия операционной системы
<code>user.home</code>	Домашний каталог пользователя
<code>user.dir</code>	Текущий каталог

Установка свойств

- Задание `property` устанавливает значение свойства
- Атрибуты
 - `name` – имя свойства
 - `value?` – значение свойства
 - `location?` – путь к файлу или каталогу
- Примеры
 - `<property name="module-A-present" value=""/>`
 - `<property name="project.dir" location="project"/>`
 - `<property name="source.dir" location="{project.dir}/source"/>`

Загрузка свойств

- Задание `property` загружает свойства из `.property` файла
- Атрибуты
 - `file?` – путь к файлу
 - `url?` – URL файла
 - `resource?` – путь к ресурсу
- Примеры
 - `<property file="build.properties"/>`
 - `<property url="file:///./project.properties"/>`
 - `<property resource="resources/project.properties"/>`

Пример .property-файла

build.compiler=jikes

deploy.server=lucky

deploy.port=8080

deploy.url=http://\${deploy.server}:\${deploy.port}/

Проверка наличия файла

- Задание `available` устанавливает свойство если файл/ресурс доступен
- Атрибуты
 - `property` – имя свойства
 - `file?` – имя файла
 - `resource?` – имя ресурса
 - `type?` – тип (`file` или `dir`)
 - `value?` – устанавливаемое значение
- Пример
 - `<avalilable property="module-A-present" file="lib/module-a.jar"/>`

Задание `tstamp`

- Задание `tstamp` Устанавливает свойства переменные `DSTAMP` и `TSTAMP` и время запуска для других программ
- Установленные значения применяются другими заданиями
- Пример `tstamp`

Проверка изменения файлов

- Задание `uptodate` устанавливает свойство если файл/ресурс доступен
- Атрибуты
 - `property` – имя свойства
 - `srcfile` – имя проверяемого файла
 - `targetfile` – имя файла с которым сравнивают
 - `value?` – устанавливаемое значение
- Пример
 - `<uptodate property="module-A-ok" srcfile="ModuleA.java" targetfile="module-a.jar"/>`

Часть 6

Наборы файлов

Наборы файлов

- `<fileset>` – набор файлов
 - `dir` – базовый каталог
 - `file` – множество из одного файла
 - `includes` / `excludes` – маски включения / исключения файлов
 - `<patternsets>` – наборы шаблонов
 - Вложенные наборы файлов

Наборы шаблонов

- `<patternset>` – набор шаблонов
 - `includes` / `excludes` – маски включения / ИСКЛЮЧЕНИЯ
 - `if` / `unless` – работа в зависимости от значений СВОЙСТВ

Наборы каталогов

- `<dirset>` – набор каталогов
 - `dir` – базовый каталог
 - `file` – множество из одного файла
 - `includes` / `excludes` – маски включения / исключения файлов
 - `<patternsets>` – наборы шаблонов
 - Вложенные наборы каталогов

Список файлов

- `<filelist>` – набор каталогов
 - `dir` – базовый каталог
 - `files` – список имен файлов
 - `<name>` – имена файлов
- Наличие файлов не проверяются

Часть 7

Пути

Наборы путей

- Строковое описание
 - Части разделяются : или ;
- Структурное описание
 - `<path>` -- Набор путей
 - `<classpath>` -- Java classpath

Элементы путей

- `<pathelement path="..." />` -- включить набор путей
- `<pathelement location="..." />` -- включить одно местоположение
- `<fileset>`, `<dirset>` -- набор файлов/каталогов

Пример задания пути

```
<classpath>  
  <pathelement path="{classpath}"/>  
  <pathelement location="{lib.dir}ant.jar"/>  
  <dirset dir="{build.dir}">  
    <include name="**/classes"/>  
  </dirset>  
</classpath>
```

Именованные пути

- Создание именованного пути
 - `<path id="...">...</path>`
- Ссылка на именованный путь
 - `<path refid="..."/>`
 - `<classpath refid="..."/>`

Пример. Именованные пути

```
<path id="project.class.path">  
  <pathelement location="lib/" />  
  <pathelement path="{java.class.path}" />  
  <pathelement path="{additional.path}" />  
</path>  
<rmic ...>  
  <classpath refid="project.class.path" />  
</rmic>  
<javac ...>  
  <classpath refid="project.class.path" />  
</javac>
```

Часть 8

Заключение

ССЫЛКИ

- Ant Homepage // <http://ant.apache.org>
- Ant Manual // <http://ant.apache.org/manual/index.html>
- Overview of Ant tasks // <http://ant.apache.org/manual/tasksoverview.html>

Вопросы