



Apache Ant

Leonardo Gresta Paulino Murta
leomurta@ic.uff.br

Índice

- Introdução
- Instalação
- Descritor
- Projeto
- Alvos
- Tarefas
- Propriedades
- Exemplo

Introdução

- Sistema de apoio à construção (*building*) escrito em Java;
- Semelhante ao Make;
 - Sem suas limitações para desenvolvimento multi-plataforma (diferentes sistemas operacionais);
- Descritores escritos em XML;
- Exibição de resultados em texto, XML, e-mail, etc;
- Ampla variedade de comandos disponíveis;
- Extensível via classes Java;
- Utilizado pela maioria das IDEs de programação.

Instalação

- Home-Page:
 - <http://ant.apache.org>;
 - <http://nant.sourceforge.net> (versão para .NET/Mono);
- Embutido em IDEs (Eclipse, NetBeans, JBuilder, etc.)
 - Suporte para a interpretação de descritores;
 - Suporte para a construção de descritores via wizards (em alguns casos);
- Instalação no Debian Linux:
 - # `apt-get install ant`
 - Nenhuma configuração é necessária!!!

Descritor

- Usualmente, os descritores de construção Ant são definidos via arquivo build.xml;
- Todo descritor de construção Ant é composto por três elementos principais:
 - Project: Representa o projeto em questão;
 - Target: Representa um alvo de construção (e.g.: compila, testa, empacota, etc.);
 - Task: Representa uma tarefa específica a ser automatizada (e.g.: javac, ftp, ssh, cvs, junit, etc.)



Projeto

- Project é o elemento de mais alto nível de um descritor de construção;
- O elemento project é composto de:
 - Nome do projeto;
 - Alvo default do projeto;
 - Diretório base;
 - Sub-elemento de descrição;
- Exemplo:

```
<project name="Alo Mundo" default="executa" basedir=".">  
  <description>Projeto de teste do Ant.</description>  
  ...  
</project>
```

Alvos

- Target representa os possíveis alvos de construção;
- O elemento target é composto de:
 - Nome do alvo;
 - Dependências de execução;
 - Condições (if e unless);
 - Descrição sucinta;

- **Exemplo:**

```
<target name="executa" depends="init,compila">
```

```
...
```

```
</target>
```

Alvos

- Um alvo é executado uma única vez;
 - Mesmo que exista mais de uma dependência;
- Todas as dependências são executadas antes do alvo em questão, na ordem informada;
- Exemplo:

```
<target name="A"/>  
<target name="B" depends="A"/>  
<target name="C" depends="B,A"/>
```
- Resultado da execução de C: A,B,C;

Tarefas

- Task representa tarefas a serem executadas em um alvo;
- O elemento task é composto de:
 - Nome da tarefa;
 - Parâmetros para a sua execução;
- Tarefas são agrupadas em três tipos:
 - Internas: incluídas no Ant;
 - Opcionais: disponíveis via bibliotecas adicionais;
 - Feitas pelo usuário;
- Exemplos:

```
<javac srcdir="${src}" destdir="${build}"/>
```

Tarefas (Internas)

■ Exemplos de tarefas internas existentes:

- Ant;
- BuildNumber;
- Copy;
- CVS;
- Delete;
- EAR;
- Echo;
- Exec;
- Input;
- Jar;
- Java;
- Javac;
- Javadoc;
- Mail;
- Mkdir;
- Move;
- SignJar;
- Sql;
- Tempfile;
- TStamp;
- Unzip;
- War;
- Xslt;
- Zip;

Tarefas (Opcionais)

- Exemplos de tarefas opcionais existentes:
 - .NET;
 - Clearcase;
 - Continuous;
 - EJB;
 - FTP;
 - JavaCC;
 - JspC;
 - JUnit;
 - JUnitReport;
 - Perforce;
 - PvcS;
 - Rpm;
 - ServerDeploy;
 - Scp;
 - Sshexec;
 - Starteam;
 - Telnet;
 - SourceSafe;

Propriedades

- São definidas no descritor de construção ou em um descritor auxiliar: `build.properties`;
 - Representadas por tuplas nome/valor;
 - Acessadas por `${nome}`;
 - Exemplo de propriedades pré-definidas:
 - `basedir`;
 - `ant.project.name`;

■ Exemplos:

```
<property file="${basedir}/build.properties"/>
```

```
  No build.properties: builddir = ${basedir}/build
```

```
<property name="builddir" value="${basedir}/build"/>
```

Exemplo

- Sistema simples em Java (SE):
 - Código fonte em “src”;
 - Código compilado e biblioteca em “build”;
- Alvos definidos:
 - Init;
 - Compila;
 - Executa;
 - Clean;

Exemplo

- **Propriedades (build.properties):**

```
mainclass = AloMundo  
srcdir = ${basedir}/src  
builddir = ${basedir}/build
```

- **Projeto (build.xml):**

```
<project name="Alo Mundo" default="executa" basedir=".">  
  
  <description>Projeto de teste do Ant.</description>  
  
  <!-- PROPRIEDADES -->  
  <property file="${basedir}/build.properties" />
```



Exemplo

■ Alvo init:

```
<!-- ALVO INIT -->  
<target name="init">  
    <mkdir dir="${srcdir}" />  
    <mkdir dir="${builddir}" />  
</target>
```

■ Alvo compila:

```
<!-- ALVO COMPILA -->  
<target name="compila" depends="init">  
    <javac srcdir="${srcdir}" destdir="${builddir}" />  
</target>
```



Exemplo

■ Alvo executada:

```
<!-- ALVO EXECUTA -->  
<target name="executa" depends="init,compila">  
    <java classname="${mainclass}">  
        <classpath path="${builddir}"/>  
    </java>  
</target>
```

■ Alvo clean:

```
<!-- ALVO CLEAN -->  
<target name="clean">  
    <delete dir="${builddir}"/>  
</target>
```



Exemplo

- Utilização (1ª execução):

```
$ ant
```

```
Buildfile: build.xml
```

```
init:
```

```
    [mkdir] Created dir: /alomundo/build
```

```
compila:
```

```
    [javac] Compiling 1 source file to /alomundo/build
```

```
executa:
```

```
    [java] Alo Mundo!!!
```

```
BUILD SUCCESSFUL
```

```
Total time: 2 seconds
```



Exemplo

- Utilização (demais execuções):

```
$ ant
```

```
Buildfile: build.xml
```

```
init:
```

```
compila:
```

```
executa:
```

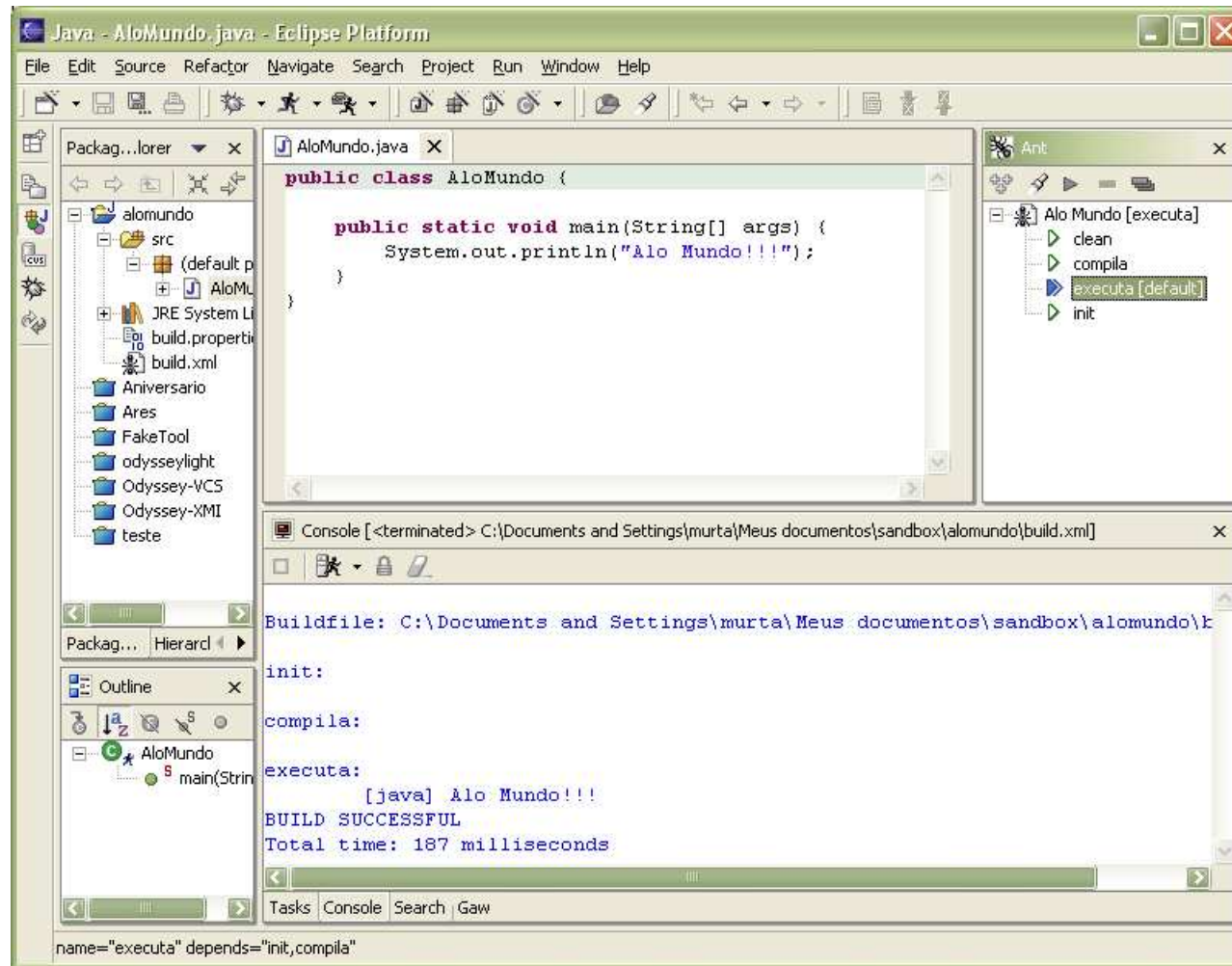
```
    [java] Alo Mundo!!!
```

```
BUILD SUCCESSFUL
```

```
Total time: 1 second
```



Exemplo



Utilização do Ant na IDE Eclipse

Exercício

- Grupos de ± 5 pessoas;
- Definir uma rotina de construção que contemple:
 - Recursos (figuras, sons, etc.);
 - Dependências de bibliotecas;
- Permitindo:
 - Compilação e Rebuilding;
 - Execução de testes (JUnit, NUnit, CppUnit, etc.);
 - Execução;
 - Geração de documentação (javadoc, cppdoc, etc.);
 - Empacotamento (jar, zip, etc.);
- Discutir as diferenças entre as rotinas propostas;

Exercício

- Mesmos grupos do exercício anterior;
- Definir uma rotina de construção que, fazendo uso da rotina anterior, contemple:
 - Integração contínua;
 - Disponibilização de resultados em site;
 - Liberação do produto;
- Discutir as diferenças entre as rotinas propostas;

Ant

Leonardo Gresta Paulino Murta
leomurta@ic.uff.br