

Java: Classpath et Packages

Code: java-util

Originaux

url: <http://tecfa.unige.ch/guides/tie/html/java-util/java-util.html>

url: <http://tecfa.unige.ch/guides/tie/pdf/files/java-util.pdf>

Auteurs et version

- Daniel K. Schneider - Vivian Synteta
- Version: 1.3 (modifié le 11/4/01 par DKS)

Prérequis

- Connaissances de base du système (fichiers *.bat)

Modules

Module technique suivant: java-intro

Objectifs

- Utilisation du JDK de SUN
- définition d'un classpath
- faire des simples scripts pour initialiser un environnement
- faire des simples scripts pour lancer une application sur la ligne de commande
- packages et archives

1. Table de matières détaillée

1.	Table de matières détaillée	3
2.	Trouver l'exécutable de Java	4
3.	Introduction au classpath, aux packages et path du système	5
3.1	Le principe du classpath	5
3.2	Mécanisme classpath de base	6
	Définir le classpath avec la variable d'environnement "CLASSPATH"	6
	Définir le classpath avec une option	8
	Applets	9
	Classpath, packages spéciaux et browsers WWW	10
3.3	Initialiser Java et Classpath avec un script	11
4.	Définir un classpath pour la compilation ou l'exécution	12
	Définir le classpath pour la compilation	12
4.1	Classpath pour exécuter un programme	13
4.2	Ecrire des scripts pour lancer une application Java	14
5.	Définir le classpath pour compiler (javac) dans le JDE/Emacs	15
5.1	Définition globale au niveau "projet" (solution conseillée)	15
5.2	Définir le classpath pour run / compile dans le JDE/Emacs	16
6.	Les packages et archives	17
	Installation de packages (autre que ceux de Sun)	18
	Utilisation de packages dezipés	19
	Utilisation de packages de type *.jar ou *.zip	20
6.1	Applications et applets avec plusieurs fichiers et archives	21

2. Trouver l'exécutable de Java

- Suivant l'installation de votre machine, le système ne trouvera pas les exécutables de Java: `java`, `javac`, `jre` etc. (ou encore une mauvaise version).
- Le répertoire des binaires doit être dans le "path" des exécutables de votre machine.
- Windows:
 - Vous pouvez éditer le fichier `autoexec.bat` pour y insérer le chemin. On déconseille, car il est préférable de décider coup par coup avec quelle version de java vous voulez travailler.

Stratégie conseillée sous Windows:

- Editez le fichier `autoexec.bat` et mettez ajoutez un chemin pour un répertoire `c:\bin` à la dernière ligne du fichier (pas nécessaire à TECFA!)

```
SET PATH=%PATH%;c:\bin
```

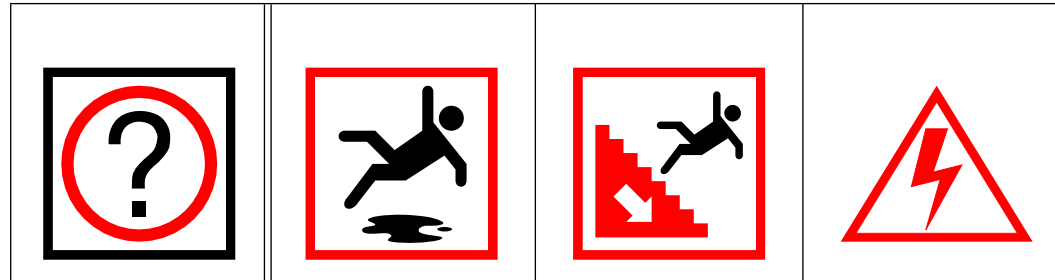
- Faites un fichier `java2.bat` qui insère votre java DEVANT le chemin

```
set path=c:\java\jdk1.3\bin;%path%
echo Les variables d'environnement ont ete mise a jour pour travailler
avec java2 (jdk1.3 dans W:\jdk1.3)
```

A chaque fois où vous travaillez avec java, ouvrez une fenêtre dos et tapez "java2". Cette commande va exécuter le fichier `*.bat` et initialiser l'environnement.

3. Introduction au classpath, aux packages et path du système

Faites votre choix:



3.1 Le principe du classpath

- Pour compiler ou exécuter un programme java qui importe de packages (externes au "core") ou qui utilise des classes additionnelles à vous, il faut indiquer où se trouvent ces classes, soit
 - un répertoire
 - un archive (de type *.jar ou *.zip)
- Le mécanisme du classpath change selon la version du JDK et l'outil utilisé.
- Quand java (javac, jre, etc.) se plaint, il faut vérifier le classpath et la documentation:
 - par ex: <http://tecfa2.unige.ch/guides/java/jdk1.2/docs/tooldocs/tools.html>

3.2 Mécanisme classpath de base

- Quand le compilateur ne trouve pas une classe, il faut rajouter l'endroit où elle se trouve (ou plutôt le package où elle se trouve) dans le classpath
- Vérifiez comment une application se comporte lorsque vous la lancez dans un autre répertoire et sur une autre machine et avec une autre version de Java.

A. Définir le classpath avec la variable d'environnement "CLASSPATH"

- Normalement pas besoin d'inclure les classes du système (classes.zip).
 - Mais si elles se trouvent au mauvais endroit (ou si Windows n'arrive pas à le trouver), il FAUT les inclure (!)
- Il faut inclure le répertoire où se trouve vos classes (votre programme) !!!
 - Si vous lancez java/javac dans le répertoire de travail: "."
- Vous pouvez définir un classpath dans autoexec.bat, mais c'est déconseillé parce certains programmes sont configurés pour travailler avec une installation standard (Java de Microsoft).

Exemple 3-1: Définitions de classpath

(Windows) :

Syntaxe: **set CLASSPATH=path1;path2 ...**

```
set CLASSPATH=C:\java\MyClasses;C:\java\OtherClasses
```

```
set CLASSPATH=C:\java\MyClasses\mysql.jar;.
```

(Unix/csh):

Syntaxe: **setenv CLASSPATH path1:path2:**

```
setenv CLASSPATH /local/java/classes/mm-jdbc/mysql.jar:/local/java/
classes/xml-sun/xml.jar:/local/java/JSDK2.0/lib/jsdk.jar:.
```

(Unix/sh/bash/ksh):

Syntaxe: **set CLASSPATH=path1:path2:**

```
export CLASSPATH=/local/java/applis/Xsl/xsl-standalone-0.7.1b1.jar
```

ou:

```
CLASSPATH=./lib/ldap.jar:./lib/jndi.jar:./lib/providerutil.jar:./
lib/ldapbp.jar
```

```
export CLASSPATH;
```

Note:

- Vous pouvez combiner l'initialisation d'un classpath avec celle d'un environnement Java (voir: 2. "Trouver l'exécutable de Java" [4])

B. Définir le classpath avec une option

- Avantage: on ne dépend pas d'un script d'initialisation (ev. inadaptée)
- Pour l'interface emacs voir 5. "Définir le classpath pour compiler (javac) dans le JDE/Emacs" [14]
- Voir les détails dans 4. "Définir un classpath pour la compilation ou l'exécution" [11]

Compilateur Sun (JDK)

Syntaxe: `javac -classpath path1;path2;... Fichier.java`
`javac -classpath c:\java\classes\javagently.jar;. Controller.java`

Lanceur Java (JDK, application)

Syntaxe: `java -classpath path;path2;... Fichier`
`java -classpath c:\java\classes\javagently.jar;. Controller`

Java runtime engine (JDK, application)

Syntaxe: `jre -cp`
`java -cp c:\java\classes\javagently.jar;. Controller`

***Notez:** Le point (.) à la fin du chemin définit se réfère au répertoire courant.

C. Applets

- On définit un "applet" depuis une page HTML avec la balise <APPLET>. Dans cette balise, le paramètre "archive" dit où se trouvent les packages ou classes externes, le paramètre "code" dit où se trouve votre programme (par défaut c'est dans le même répertoire que le fichier *.html)

Le paramètre ARCHIVE

Syntaxe: ARCHIVE="path1,path2" (VIRGULE pour séparer !)

Le paramètre CODE

Syntaxe: CODE="path"

Exemple 3-2: Inclure des archives et des classes dans une applet

url: <http://tecfa.unige.ch/guides/java/staf2x/ex/xml/sun-gui/>

- On inclut l'archive project.jar (qui est dans le même répertoire) et l'archive xml.jar qui se trouve dans le répertoire dessus)
- Les classes project/GuiDemo.class sont dans l'archive project.jar
- Le tag "applet" dans le fichier *.html

```
<APPLET
  ALT="Disabled Java, did we? Don't do that!"
  HEIGHT=460 WIDTH=620
  ARCHIVE="project.jar,../xml.jar"
  CODE="project/GuiDemo.class" >
```

D. Classpath, packages spéciaux et browsers WWW

- Le browser WWW doit connaître les classes que vous importez
 - Par exemple le package "javagently" ne marchera pas avec un applet (même si votre classpath est bien défini car votre browser Web utilise son propre Java).
 - Il faut les déclarer dans l'applet (voir C. "Applets" [9])
- Certains packages ne marchent pas avec des applets:
 - des plugins (comme CosmoPlayer) installent des classes sur votre disque dur (il faut s'assurer que ces classes sont toujours là)
- Certains packages sont trop gros pour les distribution avec un applet
 - Par exemple Swing 1.1 avec la VM standard de Netscape 4.x (qui est un java 1.1.x)
 - Il faut donc les COPIER dans le répertoire "classes" de votre browser

3.3 Initialiser Java et Classpath avec un script

Exemple 3-3: Java Home et classpath sous MS-DOS (win et NT)

```
@echo off
set JAVA_HOME=C:\soft\jdk1.3
set path=%JAVA_HOME%\bin;%path%
echo java 2 avec xerces, xalan, eai et jdbc dans Q:\guides\java\classes\
set
CLASSPATH=.;q:\guides\java\classes\xerces.jar;q:\guides\java\classes\xalan.jar;q:\guides\java\classes\mysql.jar;q:\guides\java\classes\vrml-cosmo-eai.jar
echo path : %path%
echo java est installé en %java_home%
echo CLASSPATH: %CLASSPATH%
cd q:\guides\java\staf2x\ex
```

url: <http://tecfa.unige.ch/guides/java/staf2x/inits/initstaf.bat>

- **Note:** Ce script nécessite que vous montiez la partition tecfa:/web sur q:
- Il est conseillé de mettre ces scripts dans un endroit se trouvant dans le path, ainsi on peut taper cette commande partout
- Sous Unix vous faites pareil (selon le shell que vous utilisez)

4. Définir un classpath pour la compilation ou l'exécution

A. Définir le classpath pour la compilation

- Conseil: mettez le minimum dans le système
 - Jdk 1.1.7 et Jdk 1.2 ont par défaut les classes du core et le répertoire courant dans le classpath
- Pour travailler avec d'autres packages (MySQL, XML, VRML, les vôtres):
 - il faut les inclure dans le CLASSPATH (sinon Java ne les trouve pas!)
 - alternative idiote: unzipper tout le package dans le répertoire courant
- Pour redéfinir le classpath, utilisez:
 - (a) soit les fonctionnalités de votre outil,
 - (b) soit l'option "classpath" de 'javac' et de 'java',
 - (c) soit la variable d'environnement CLASSPATH
 - Dans les 2 premiers cas, il faut savoir que vous écrasez le classpath par défaut ci-dessus (avec Jdk 1.1.7 ou Jdk 1.2 en tout cas).
- Archive de classes fréquemment utilisées: <http://tecfa.unige.ch/guides/java/classes/>

Exemple 4-1: Classpath typique sous Windows pour le JDK 1.2

```
javac -classpath "c:\java\classes\javagently.jar;." Controller.java
```

- Alternativement, vous auriez pu unzipper javagently.jar dans le répertoire

4.1 Classpath pour exécuter un programme

- Même principe que pour compiler:
 - Sans classpath défini, il met le classpath standard
 - Votre programme doit trouver les classes non-standard
 - Il cherche les classes externes dans un sousrépertoire du répertoire d'exécution. Si les classes se trouvent ailleurs (même dans une archive qui se trouve dans le même répertoire), il faut redéfinir le classpath ou mieux utiliser -cp pour rajouter des classes !!
 - Dans la doc: "If -classpath and -cp are not used and CLASSPATH is not set, the user class path consists of the current directory (.)"

Exemple 4-2: Classpath "run" typique sous Windows pour le JDK 1.2

(permet de travailler avec le package JavaGently)

Avec le lanceur java

```
java -classpath "c:\java\classes\javagently.jar;." Controller
```

Avec le lanceur jre

```
jre -cp "c:\java\classes\javagently.jar;." Controller  
jre -cp ../javagently.jar Controller
```

4.2 Ecrire des scripts pour lancer une application Java

- Une application qui s'utilise réellement ne se lance pas à la main
 - Il faut écrire un fichier bat (Windows) ou shell (Unix)
 - Lorsque vous téléchargez une appli de Internet, ils vous disent en règle générale comment il faut faire.

Exemple Unix: fichier /local/go/vrmlint sous Unix:

```
#!/bin/csh -f
source /local/env/java117.csh
jre -cp /local/java/applis/vrmlint/vorlon.jar trapezium.vorlon $*
```

Exemple Windows hypothétique (vous pouvez éliminer le "pause")

```
Echo si vous voyez apparaitre le message "Espace d'environnement insuffisant"
Echo Cliquez avec le bouton droit sur le fichier WeatherChart.bat
Echo Demandez les Propriétés et dans l'onglet Mémoire
Echo Changez l'environnement initial en 4096
Echo Cliquez sur Ok puis relancez le fichier WeatherChart.bat
Echo en double-cliquant dessus.
```

pause

```
SET PATH=C:\Sylvere\WeatherChart
SET CLASSPATH=.;C:\Progra~1\Jdk1.2;C:\Sylvere\WeatherChart
```

```
c:\Progra~1\Jdk1.2\Bin\JAVA "WeatherChart"
```

5. Définir le classpath pour compiler (javac) dans le JDE/Emacs

5.1 Définition globale au niveau "projet" (solution conseillée)

- Pour manipuler cet interface il faut utiliser le bouton droit ou du milieu (appuyer gauche+droit sur des souris à 2 boutons)
- Emacs ne change pas le CLASSPATH mais appelle java avec -classpath
 - ECRASE donc le classpath par défaut
 - il faut éventuellement remettre les classes du core API comme dans l'exemple ci-dessous !)
- Si cela vous stresse, définissez le CLASSPATH avant de lancer emacs.

Exemple (sous Unix, ajustez pour Windoze !)

1. Menu JDE->Options->Project
2. Scroller vers "Global Classpath" et ouvrir (avec right ou middle click)
3. Right-click (ou middle -click) sur [INS] et rajouter path par path

Classpath: *

```
[INS] [DEL] Path: /comm/tecfa/www/guides/java/examples/gently/javagently.jar
```

```
[INS] [DEL] Path: /local/java/jdk1.1.7/classes.zip
```

```
[INS] [DEL] Path: .
```

```
[INS]
```

```
[State]: you have edited the value as text, but you have not set the option.
```

4. Pour sauver, right-clicker sur [State]

- soit sauver juste pour la session, soit forever

.....

[State]: you have set this option, but not saved it for future sessions.

- Si vous avez indiqué un classpath "forever", ne l'oubliez pas, mais à priori avoir défini trop de chemins ne gêne pas (juste un chemin qui pointe vers une vieille version

5.2 Définir le classpath pour run / compile dans le JDE/Emacs

1. Menu JDE->Options->Run et JDE->Options->Compile
2. Scroller vers "Classpath" et ouvrir (avec right ou middle click)
3. ... le reste comme ci-dessus !

6. Les packages et archives

- Sun a défini un mécanisme pour nommer, installer et importer des packages.
 - chaque package important possède un nom unique (comme les noms Internet).
- Types de packages:
 - "Core": viennent avec le JDK et les JRE
 - "Extensions": s'installent dans le JDK/JRE (Java 1.2 seulement)
 - "Archives externes": tout ce qui vient d'ailleurs
- Ces packages sont normalement distribués avec des archives *.jar
- Exemples de noms de packages:
 - java.io
 - org.tecfa.campus
 - javagently
 - COM.Ora.writers.david.widgets

Archives externes populaires à Tecfa

- mysql.jar (bibliothèque MySQL)
- xml.jar et xml4j.jar (bibliothèques xml de Sun et de IBM)
- swing1.1.jar (bibliothèque Swing pour Java 1.1 de Sun)

A. Installation de packages (autre que ceux de Sun)

- Il faut choisir un endroit (class directory) pour installer ces packages
 - Vous pouvez les mettre où vous voulez. Étant donné qu'ils s'agit de noms uniques pour chaque package sur Internet, un bon endroit est l'endroit où se trouve le JDK. (genre c:\Program Files\java\classes)
- Normalement à TECFA ils sont:
 - Pour unix: dans /local/java/classes (les scripts d'initialisation: /local/env)
 - Sur le web (classes populaires): <http://tecfa.unige.ch/guides/java/classes/>
- Ensuite deux solutions, soit:
 - Placer le fichier *.jar ou *.zip dans votre répertoire des classes Java
 - Dézipper ou déjarer ces archives exactement SOUS cet endroit (déconseillé)
 - Voir -B. "Utilisation de packages dézippés" [18] et -C. "Utilisation de packages de type *.jar ou *.zip" [19] pour les détails ...
- Note: si vous avez juste besoin de classes externes pour juste compiler un seul fichier, vous avez intérêt à dézipper l'archive SOUS le répertoire où vous travaillez. Ainsi, pas besoin de définir un classpath.

B. Utilisation de packages dézippés

- Dans le classpath il faut mettre le répertoire "mère" du package dézippé !
 - Il ne faut donc PAS mettre les sous-répertoires du package dans le classpath mais juste le répertoire dessus !!
- Les exemples suivants sont pour Unix.
 - Ajustez pour Windows (par exemple c:\home\daniel\java)

Exemple 6-1: Le package javagently dézippé

- Le répertoire du livre "JavaGently" est ici:

```
/comm/tecfa/www/guides/java/examples/gently/
```

- Il contient le sous-répertoire javagently
(qui s'appelle obligatoirement "javagently" à cause du package name)

```
tecfasun4:examples 37) ls -la gently/javagently
total 20
drwxrwxr-x   2 schneide tecfa      512 Dec  3 10:42 ./
drwxrwxr-x  18 schneide tecfa      512 Dec  3 10:36 ../
-rw-r-----   1 schneide tecfa    3506 Dec  3 10:42 Text.class
-rw-r-----   1 schneide tecfa    3975 Dec  3 10:42 Text.java
```

- Le CLASSPATH ou -classpath doit contenir:

```
/comm/tecfa/www/guides/java/examples/gently/
javac -classpath "/comm/tecfa/www/guides/java/examples/gently/" Controller.java
    et PAS:
/comm/tecfa/www/guides/java/examples/gently/javagently
```

C. Utilisation de packages de type *.jar ou *.zip

- Il n'est pas utile de dézipper une archive de type *.jar ou de type *.zip bien fait.
 - En gros, un fichier *.jar est à priori "bien fait", un fichier *.zip n'est peut-être qu'une archive de fichiers (et pas fait spécialement pour Java)
- Il faut mettre chemin/fichier *.jar ou *.zip dans le classpath !
 - Attention: s'il n'est pas correctement zippé (comme par exemple le package JavaGently cela ne marche pas, j'ai du utiliser l'outil "jar")

6.1 Applications et applets avec plusieurs fichiers et archives

- On reprend l'exemple 3-2 "Inclure des archives et des classes dans une applet" [9]
 - Tous les fichiers se trouvent dans le répertoire project. Ces fichiers peuvent être compilés en une seule fois, s'ils sont dans le classpath
 - L'argument -d de javac met les fichiers *.class dans un autre répertoire
- Voici la ligne de compilation (sous Unix)

```
javac -g -d classes -classpath /local/java/classes/xml-ea2/xml.jar:/local/java/jdk1.2/lib/classes.zip project/PlayElement.java project/ProjectElement.java project/SceneElement.java project/ShowElement.java project/TitledElement.java project/GuiDemo.java
```

- Voici l'instruction pour faire l'archive:

```
cd classes; jar cvf ../project.jar *
```

- Voir aussi: C. "Utilisation de packages de type *.jar ou *.zip" [19]

- Voici la structure complète du répertoire APRES compilation:

```
tecfasun4:sun-gui 236) ls -R
.:
Makefile          TECFA.README      example2.html     project.jar
Makefile.ORI      classes/          example2.html.ORI show.props
README           example1.html     play.props
README.html       example1.html.ORI project/

./classes:
project/

./classes/project:
GuiDemo$1.class      ProjectElement.class
GuiDemo$ErrorDialog.class SceneElement.class
GuiDemo$ErrorPrinter.class ShowElement$MaskedNode.class
```

```
GuiDemo$TreePanel.class      ShowElement.class
GuiDemo.class                TitledElement.class
PlayElement.class

./project:
GuiDemo.java                 ProjectElement.java   ShowElement.java
PlayElement.java            SceneElement.java     TitledElement.java
```

•

Exemple 6-2: Le package javagently comme *.jar file

- **Fait avec**

```
jar cfv javagently.jar javagently
```

- **javagently.jar doit être dans le classpath, exemple (une ligne!):**

```
javac -classpath "/comm/tecfa/www/guides/java/examples/gently/ javagently.jar:/local/java/
jdk1.1.7/lib/classes.zip" Controller.java
```