

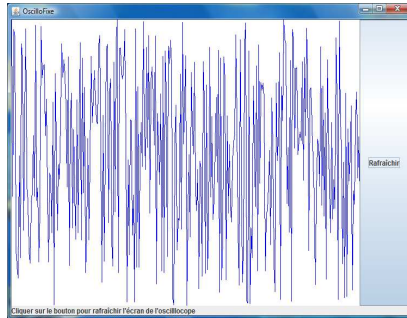


Introduction à JNI

CREATION : 2010/11/28

MISE A JOUR : 2010/11/29

Java Native Interface (JNI) est un mécanisme permettant d'interfacer un programme Java avec un code natif. La communication peut se faire dans les deux sens mais je vous propose d'appeler une méthode/fonction écrite en C/C++.



Code Java

Réutiliser le code permettant d'afficher l'équivalent d'un oscilloscope simple (si vous ne l'avez pas fait le code est disponible en téléchargement sur <http://www.isima.fr/~loic>). Je vous propose d'externaliser le code de la fonction renvoyant soit un réel entre 0 et 1, soit un nombre entre 0 et une borne donnée.

```
public static native double fonction_native();
```

Le code natif doit se trouver dans une bibliothèque dynamique qu'il faut charger avant d'en avoir besoin. Si la bibliothèque s'appelle libNOM.so, il faut ajouter les lignes suivantes :

```
static {
    System.load("NOM_AVEC_CHEMIN_COMPLET");
    System.loadLibrary("NOM");
}
```

Voilà c'est presque terminé, la dernière étape est la création d'un fichier include pour le programme C/C++. Il faut donner en paramètre le nom du fichier class (sans extension) **contenant la déclaration** de la fonction native.

```
javah -jni fichier_class
```



Code C/C++

Regarder le fichier include généré par javah. Il contient la déclaration de la fonction dont on doit donner la définition.

Créer un fichier C faisant le travail demandé. Il faut inclure stdlib pour pouvoir utiliser la fonction rand() et la constante symbolique RAND_MAX.

Le plus « compliqué » reste la création de la bibliothèque avec la ligne suivante :

```
gcc fichier.c -o libNOM.so -shared -Wl,-soname,NOM.so
-I/.../include -I/...include/linux -static -lc
```

Pour les répertoires donnés avec l'option -I, il faut trouver le répertoire contenant jni.h (dans un sous-répertoire de /usr/local sur etud).

Pour etud, il faut enlever l'option -static et ajouter -fPIC

Il ne reste plus qu'à exécuter le programme java avec la machine virtuelle. Si vous avez un message d'erreur contenant l'emplacement de la bibliothèque, il faut s'assurer que le LD_LIBRARY_PATH soit correct.

```
LD_LIBRARY_PATH=`pwd`
export LD_LIBRARY_PATH
```

Attention, etud est une machine 64bits, il faut être sûr de lancer une JVM 64 bits (avec l'option -d64 par exemple)

Compléments

Si cela vous intéresse, vous pouvez lire le tutoriel rapide de Sun/Oracle :

<http://java.sun.com/developer/onlineTraining/Programming/JDCBook/jni.html>