|  |  |
| --- | --- |
|  | **2014** |
|  | The Company  Martien Huijsmans |

|  |
| --- |
| **[Java-JNI]** |
| Usage |

Inhoud

[Introduction 1](#_Toc391071920)

[Inspiring References 1](#_Toc391071921)

# Introduction

This document captures experiences in build on JNI

# Tooling

Maven, install on debian  
<http://tutorialforlinux.com/2014/03/18/how-to-install-latest-apache-maven-on-debian-squeeze-6-32-64bit-linux-easy-guide/>

Project that deals with loading native libraries  
<https://github.com/scijava/native-lib-loader>   
Impression: (how to say this) there is not a single baseline towards which you work.

# Inspiring References

Maven2-jni  
<http://docs.codehaus.org/display/MAVENUSER/Projects+With+JNI>  
Key aspect: package native library onto a jar and extract library in runtime.  
Used ant-maven-plugin.

<http://www.tricoder.net/blog/?p=197>  
Includes a maven project ready to use:  
<http://www.tricoder.net/files/jniMavenExample.zip>   
It uses the native-maven-plugin

<http://stackoverflow.com/questions/10071058/adding-a-jni-library-to-the-local-maven-repository>  
Discussion with different options and links.  
Points to maven3 project: <https://github.com/peter-lawrey/Java-Thread-Affinity>  
The project uses make to build a library.  
The project also has .so files in the main/resources folder.  
Native library is packaged into the jar and in runtime is copied (as resource) to disk.  
MD5 is used to compare existing library to resource.  
This is an interesting project, e.g. It lock a thread/program to a certain core.  
See: <https://github.com/peter-lawrey/Java-Thread-Affinity/blob/master/src/main/java/com/higherfrequencytrading/affinity/impl/NativeAffinity.java>

Recent nar-maven-plugin project  
<https://github.com/imagej/imagej-launcher>

Use of SWIG (<http://www.swig.org/exec.html>) is an option.  
It also has a maven plugin and integrates with the nar-maven-plugin.  
<http://mark.donszelmann.org/maven-swig-plugin/>