

JavaCard applet development with NetBeans

© 2010 by LaBAK FI MUNI

Installation process

Following software will be used

- NetBeans 6.8 or later (<http://netbeans.org/> [<http://netbeans.org/>])
- Java 2 SDK, Standard Edition (J2SE), v 1.3.1_20 (http://java.sun.com/products/archive/j2se/1.3.1_20/index.html [http://java.sun.com/products/archive/j2se/1.3.1_20/index.html])
- GPShell 1.4.2 (<http://sourceforge.net/projects/globalplatform/files/> [<http://sourceforge.net/projects/globalplatform/files/>])
- AntTask (<http://ant.apache.org/bindownload.cgi> [<http://ant.apache.org/bindownload.cgi>])
- Java Card Development Kit 2.2.2 (<http://www.oracle.com/technetwork/java/javacard/downloads/index-jsp-138488.html#devkit> [<http://www.oracle.com/technetwork/java/javacard/downloads/index-jsp-138488.html#devkit>])

1. Install NetBeans
2. Install Java 2 SDK (JDK 1.3)
3. Install JavaCard 2.2.2
4. Install GPShell
5. Create new java project
6. Set configuration parameters in "nbproject/project.properties"
 - jc.home - path to Java Card Development Kit
 - jc.export_files - path to Java Card export files (for verifier), usually \${jc.home}\\api_export_files
 - gpshell.home - path to GPShell
 - gpshell.cmd - path to GPShell executable

Example of configuration:

```
# Java Card Development Kit home directory
jc.home=c:\\Program Files\\java_card_kit-2_2_1
jc.export_files=${jc.home}\\api_export_files
# GPShell positioned in project_path\\bin\\GPShell-1.4.2
gpshell.home=${basedir}\\bin\\GPShell-1.4.2
gpshell.cmd=${gpshell.home}\\GPShell.exe

annotation.processing.enabled=true
annotation.processing.enabled.in.editor=false
annotation.processing.run.all.processors=true
javac.processorpath=\
    ${javac.classpath}

application.desc=Sample Java Card project
application.title=SmartHelloWorld

... rest of project.properties file automatically generated for new project
```

1. Modify file 'build.xml' to contain following Ant tasks

build.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project name="SmartHelloWorld" default="default" basedir="." xmlns:j2seproject3="http://www.netbeans.org/ns/j2se-project/3">
  <description>Builds, tests, and runs the project SmartHelloWorld.</description>
  <import file="nbproject/build-impl.xml"/>
  <target name="-post-init">
    <path id="jc-tasks" >
      <pathelement path="lib/jctasks.jar"/>
    </path>
    <taskdef name="apdutool"
      classname="com.sun.javacard.ant.tasks.APDUToolTask"
      classpathref="jc-tasks" />
    <taskdef name="capgen"
      classname="com.sun.javacard.ant.tasks.CapgenTask"
      classpathref="jc-tasks" />
    <taskdef name="convert"
      classname="com.sun.javacard.ant.tasks.ConverterTask"
      classpathref="jc-tasks" />
    <taskdef name="verifyexport"
      classname="com.sun.javacard.ant.tasks.VerifyExpTask"
      classpathref="jc-tasks" />
    <taskdef name="verifycap"
      classname="com.sun.javacard.ant.tasks.VerifyCapTask"
      classpathref="jc-tasks" />
    <taskdef name="verifyrevision"
      classname="com.sun.javacard.ant.tasks.VerifyRevTask"
      classpathref="jc-tasks" />
    <taskdef name="scriptgen"
      classname="com.sun.javacard.ant.tasks.ScriptgenTask"
      classpathref="jc-tasks" />

    <path id="jc-classpath" >
      <pathelement path="${jc.home}/lib/converter.jar"/>
      <pathelement path="${jc.home}/lib/offcardverifier.jar"/>
      <pathelement path="${jc.home}/lib/scriptgen.jar"/>
      <pathelement path="${jc.home}/lib/apdutool.jar"/>
      <pathelement path="${jc.home}/lib/apduio.jar"/>
    </path>
  </target>
</project>
```

```

        <pathelement path="lib/comm.jar"/>
    </pathelement path="."/>
</target>

<target name="convert" depends="init">
    <copy todir="${build.classes.dir}">
        <fileset dir="${jc.export_files}">
            <filename name="**/*.exp"/>
        </fileset>
    </copy>

    <convert
        dir="${build.classes.dir}"
        Configfile="${src.dir}/com/sun/javacard/samples/HelloWorld/HelloWorld.opt"
        nobanner="true">
        <classpath refid="jc-classpath"/>
    </convert>
    <verifycap Verbose="true" CapFile="${build.classes.dir}/com/sun/javacard/samples/HelloWorld/javacard/HelloWorld.cap">
        <exportfiles file="${jc.export_files}/java/lang/javacard/lang.exp" />
        <exportfiles file="${jc.export_files}/javacard/framework/javacard/framework.exp" />
        <exportfiles file="${jc.export_files}/javacard/security/javacard/security.exp" />
        <exportfiles file="${jc.export_files}/javacardx/crypto/javacard/crypto.exp" />
        <exportfiles file="${build.classes.dir}/com/sun/javacard/samples/HelloWorld/javacard/HelloWorld.exp" />
        <classpath refid="jc-classpath"/>
    </verifycap>

</target>

<target name="jar" depends="init,compile,convert">
</target>

<target name="run-script" depends="init">
    <exec executable="${gpshell.cmd}">
        <arg value="scripts\\helloInstallGP211_JCOP41_cap.txt"/>
    </exec>
</target>
</project>

```

Developing new Java Card applet

1. Change Java build platform from 1.6 (default) to 1.3
 - *Project_name*→Libraries→*R-Click*→Properties→Manage Platforms→Add platform→*point to JDK 1.3 folder*
 - Choose JDK1.3 from list in Properties→Libraries→Java Platform
2. Write/Replace code with JavaCard applet (e.g., <http://www.wrinkl.de/Javacard/JCMini.html> [<http://www.wrinkl.de/Javacard/JCMini.html>])
3. Add JavaCard libraries
 - *Project_name*→Libraries→*R-Click*Add library→Create library
 - *javacard SDK folder*lib/api.jar
 - *javacard SDK folder*lib/javacardframework.jar
 - *javacard SDK folder*api_export_files
 - add created JavaCard library to project
4. Try to compile project (Project should compile without errors)

Converting and uploading applet to smart card

- Create 'helloInstallGP211_JCOP41_cap.txt' script file for GPShell suitable for your card. Example for JCOP 4.1 cards:

```

# Usable for:
# NXP JCOP 4.1, 3.1 and 2.1 cards

mode_211
enable_trace
establish_context
card_connect
select -AID a000000003000000
#select -AID A000000018434D00

open_sc -security 1 -keyind 0 -keyver 0 -mac_key 404142434445464748494a4b4c4d4e4f -enc_key 404142434445464748494a4b4c4d4e4f // Open secure channel

delete -AID a00000006203010c0101

delete -AID a00000006203010c01
install -file build/classes/com/sun/javacard/samples/HelloWorld/javacard/HelloWorld.cap -sdAID A000000018434D00 -nvCodeLimit 4000
select -AID a00000006203010c01
send_apdu -APDU B040000009010203040506070809

card_disconnect
release_context

```

- Try to rebuild your project (Run→Clean and build project). Project is compiled and converted, but not uploaded to smart card (use until java card applet compile and convert without error)
- Following steps should occur with output visible in NetBeans output console:
 1. Build of java card application, *.jar file(s) are created
 2. Conversion to java card format (convert task), *.jar and *.cap files are created in *project_path\build\...* directory
- Try to test your project (Run→Test project). Use when applet is compiled and converted without error.
- Following steps should occur with output visible in NetBeans output console:
 1. Build of java card application, *.jar file(s) are created

2. Conversion to java card format (convert task), *.jar and *.cap files are created in *project_path\build\...* directory
3. Upload to smart card (GPShell output)

Usage hints

- Both compilation and testing can be executed from NetBeans (Build or Test commands from menu) and are handled by Ant and configured in "build.xml".
- When compilation is done (compiled files are in build/classes/), task "convert" is executed and performs conversion and verification. Converted file can be found in build\classes\com\sun\javacard\samples\HelloWorld\javacard. This action can be configured in task "convert" in build.xml.
- Testing executes script "scripts/helloInstallGP211_JCOP41_cap.txt" in GPShell and tries to install HelloWorld on compatible Java Card. This action can be configured in task "run-script" in build.xml.

public/javacardcompilation2.txt · Last modified: 2010/09/26 09:57 by petrS

Except where otherwise noted, content on this wiki is licensed under the following license:CC Attribution-Noncommercial-Share Alike 3.0

Unported [http://creativecommons.org/licenses/by-nc-sa/3.0/]