

Mobile and Wireless Systems Programming

Development tools overview



Using tools

- Java ME emulation platform
- Tools to automate :
 - Compiling
 - Obfuscating
 - Pre-verifying
 - Packaging
- Debugger, profiler...

WTK : Wireless ToolKit

- Java WTK (Wireless ToolKit) for CLDC (version : 3.0)
- Desktop program which simulate the behavior of Java ME-specific programs
- Built on the top of Windows, Linux...
- Setup of deployment platform parameters in order to simulate hardware and software capabilities
- Emulates Wireless Messaging, Bluetooth connections...
- Sun but also manufacturers (Nokia, Samsung, Sony Ericsson)

WTK : KToolBar

KToolBar is a minimal development environment for developing MIDlet suites. From the KToolBar, you can :

- Create a new project or open an existing one
- Build, run, and debug your MIDlet
- Fine tune your MIDlet application
- Package your project files
- Modify the attributes of your MIDlet suite

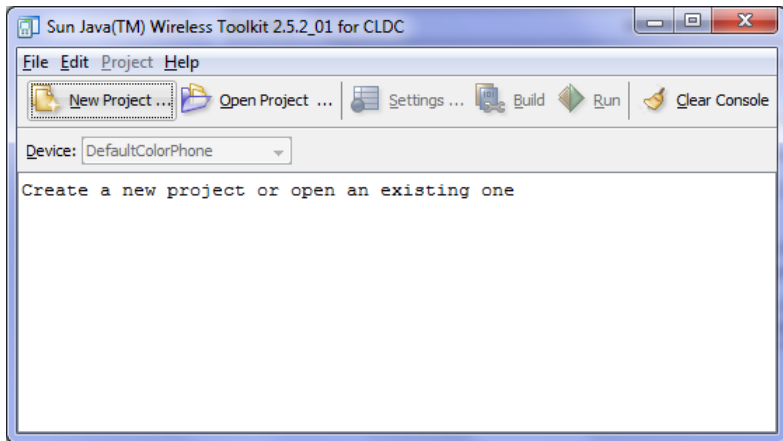
WTK : Wireless ToolKit

- kToolbar
- emulator
- profiler, debugger, build manager...
- ...

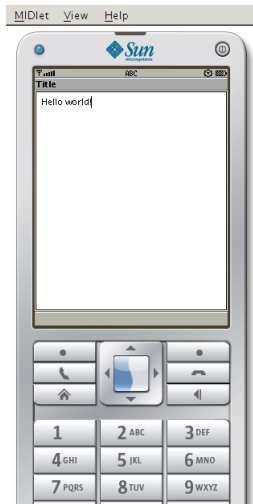
File or Directory	Description
BinaryLicense.html	License agreement.
BinaryReleaseNotes.html	Release notes.
index.html	WTK documentation.
appdb\	database files (RMS files, ME keystore files. . .)
apps\	demo applications created by the KToolBar.
bin\	Batch files and executables for the tools.
docs\	API documentation.
lib\midpapi.zip	CLDC and MIDP API classes (used during the compilation and byte-code preverification).
sessions\	profiling, memory monitoring, and network monitoring session files.
wkllib\devices\	property files for devices emulated by the Emulator.

Figure: J2ME WTK Directory Contents

Ktoolbar



Emulator



Example devices

- DefaultColorPhone
- DefaultGrayPhone
- MinimumPhone
- ...

Can be customized through `wktlib\devices\` files

Main emulators

- Sun : <http://java.sun.com/javame/sdk/index.jsp>
- Motorola : <http://developer.motorola.com>
- Sony Ericsson : <http://developer.sonyericsson.com>
- BlackBerry : <http://www.blackberry.com>
- Nokia : <http://www.forum.nokia.com/>
- Samsung : <http://innovator.samsungmobile.com>
- MicroEmulator : <http://www.microemu.org>
- ...

Batch

- Simple system calls (javac, preverify, emulator...)
- Very small project
- Not adapted, not flexible...
- To avoid
- Example : emulator
-Xdescriptor :C :\J2MEWTK\apps\example\bin\example.jad

Shell

- More flexible than Batch
- System calls
- Can be used

Ant and Antenna

- XML based
- Adapted for Java ME development
- Flexible
- Widely used
- More information : <http://antenna.sourceforge.net/>

Ant-based projects structure

Directory	Contents
src	Application source code
test	Unit test code
lib	Project dependencies
build	Any files generated by the build process
buildclasses	Compiled Java classes
buildtest-classes	Compiled unit tests
dist	Distribution files, i.e JAR or JAD files

Ant example

Listing 1 – build.xml

```
<project>
  ...
  <target name="compile" depends="init">
    <javac srcdir="src" destdir="build/classes"/>
  </target>

  <target name="package" depends="compile">
    <jar destfile="dist/myJar.jar" basedir="build/classes"/>
  </target>
  ...
</project>
```

Eclipse

- Eclipse MTJ (Mobile Tools for Java) project for Java ME
- Very flexible
- Eclipse tools :
 - Debugger
 - Profiler
 - Intellisense
 - ...
- Widely used (with antenna)
- More information : <http://www.eclipse.org>

Netbeans

- Mobility Pack
- Ant based configuration script
- Netbeans tools :
 - Debugger
 - Profiler
 - Preprocessor
 - ...
- Widely used
- More information : <http://www.netbeans.org>

Other tools

- NeoMAD
- J2mePolish
- Celsius
- Javaground
- ...

Tests

- J2MEUnit : a unit testing framework for J2ME application
- Made by hand on real devices

Conclusion

- Emulators are useful but insufficient (need real devices)
- Some links :
 - <http://www.eclipse.org/dsdp/mtj/>
 - <http://java.sun.com/javame/downloads/sdk30.jsp>
 - <http://j2meunit.sourceforge.net/>