Getting started with Java ME

Getting your first Java ME application (MIDlet) running on a real device requires a number of steps. This Wiki page is a guide through that process. As it is targeted at first-time MIDlet developers, some things are simplified, and more detailed information can be found elsewhere.

11 Jul 2010

Is Java ME the right tool for your requirements?

Java ME is a limited subset of the standard Java (Java SE) available on the desktop computers, with some additional mobile phone-related APIs. There are a number of limitations you need to keep in mind:

- MIDlets run in a sandbox because of security reasons. (There are confirmations when using certain functionalities, like networking or sending SMS messages.)
- There is no JNI (Java Native Interface) so you cannot extend the capabilities of Java ME environment on the phone.
- There are no Swing or AWT classes. MIDlets use their own (simplistic) UI classes.
- The capabilities of the Java ME environment vary widely, meaning that the phones have different set of optional APIs implemented (examples include access to files, access to phone-book, video/audio recording, 3D graphics, etc.)

So first of all, before starting anything else, think whether Java ME is the right choice for you, or whether you should instead use Flash Lite for your application (runs both on Series 40 and S60 phones), or whether you should program in Python (S60 only) or in native Symbian C++ (for S60 phones only). (However, native C++ development for Symbian OS is quite tedious and error-prone, so just for productivity reasons, Java ME might be a better choice.)

If Java ME seems to be the way to go, you can find a lot of good information in this Wiki or in Nokia Developer's Java Discussion Forums. There is also documentation available elsewhere on the Nokia Developer website, as well at Sun Web site and IBM Developer Works Web site.

Overview of the development process

First note that you have to code, compile and package the MIDlet on your development computer. You cannot create a MIDlet on the phone itself (yet).

To write MIDlet code, use whatever text editor or IDE (Integrated Development Environment) you prefer. (But see the Nokia-supported IDEs below.) Then compile the code with the Javac compiler which comes with a Java SE SDK (JDK). Even though you do not need to worry about it right now, your code is also pre-verified at this stage.

Before being able to install the MIDlet on a real device, you still need to package the MIDlet classes and resources to a JAR (Java Archive) file and generate a JAD (Java Application Descriptor) file. The JAR file contains all the code needed to run the application. The JAD file is a text file containing descriptive information about the MIDlet (size, needed APIs, MIDlet name, Vendor name, optional signing information), which the phone (and user) can use to decide whether or not to download the MIDlet at all. You need only to install the JAR file on the phone to be able to run the MIDlet; however, it is strongly recommended to use both of the files to install the application.

At this stage, you can test the MIDlet on your development workstation using the phone emulator, and then most probably make changes, then recompile and repackage it. When you are satisfied with the result, you can transfer the generated JAR and JAD files to a real phone, install the app, and test it. (You should always test a MIDlet on a real device before distributing it.)

You can also use some online resources to test your application on real devices, such Remote Device Access (RDA) or Device Anywhere.

You should also download other vendors' emulators (Motorola, Sony Ericsson, Samsung, etc.) and test your application using them.

Required hardware

The configuration of your computer depends on the tools you plan to use. Specific information can be found in each product's release notes or web site.

Required software for S60

For Java ME development for the S60 platform, the following tools are needed:

- a Java SE (Standard Edition) Development Kit (JDK)
- a Wireless Toolkit
- a Nokia Symbian/S60 SDK for Java
- an IDE that supports mobile Java development; popular ones are NetBeans and Eclipse Pulsar

For details of installing the required tools, see Installing Java ME development tools for S60, as well as the information made available on the tools' web pages.

(C) Copyright Nokia 2013. All rights reserved.

Emulators and SDKs for Series 40 and Series 80

- Java Tools for Nokia Series 40
- Java Tools for Nokia Series 80

It should be noted that Series 80 is an old platform that is no longer supported by Nokia Developer.

Nokia Asha SDK 1.0 (beta) (14 May 2013): Nokia Asha SDK 1.0 (beta) is available for download. This SDK enables you to target your Java apps at phones based on Nokia Asha software platform 1.0. SDKs for Series 40 (including *Nokia SDK 2.0 for Java*), can be downloaded from here.

Coding and compiling your first MIDlet

See Hello World in Java ME, Creating your first MIDlet using EclipseME, Developing MIDlets using Pulsar and Eclipse and NetBeans.

Typical pitfalls

Compiling the Java ME code with wrong Java compliance option

The correct compliance is 1.3 or 1.4. Compiling with 5.0 or 6.0 compliance makes it impossible to install the MIDlet.

Compiling and packaging the MIDlet for incorrect CLDC or MIDP version

This is especially true when using Sun Wireless Toolkit, which by default builds the MIDlet for MIDP 2.1. However, most of the current phones have only MIDP 2.0 available. (Note, that you can install on a phone a MIDlet with lower CLDC/MIDP version than what the phone supports, but you cannot install a MIDlet with higher CLDC/MIDP version on the phone. Check the supported CLDC and MIDP versions on FN device specification pages.)

See also

- The excellent Nokia Developer Library one of the best compilations of tutorials and documentation available

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.