Java ME

Overview

Java Platform Micro Edition or Java ME (formerly J2ME) refers to a a collection of technologies and specifications to develop applications and games for (usually) mobile devices using Java Technology. Most of these technologies have been specified in open manner in Java Community Process by Nokia, Sun Microsystems, Motorola, Ericsson, Siemens, and others. There are many implementations available in different platforms. Nokia has internally developed Java ME implementations for both Series 40 and S60 platforms. These implementations are continually being refined with new APIs and other functionality.

Nokia phones have an extensive **Java ME** API set. For reference documentation, tutorials, and examples, see Nokia Java ME Developer's Library.

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.

Architecture

The architecture includes:

- The Java language
- System Libraries and Configurations (CLDC or CDC)
- Profiles and UI Libraries (MIDP or Personal Profile)
- Java Optional Packages

See also the J2ME Mindmap.

MIDP 2.0 with CLDC is the common architecture found in the mass market. CDC with Personal Profile has more to offer to developers, but there aren't many devices yet with this configuration.

If you are using MIDP, you are developing MIDlets.

Java ME configurations

Connected Limited Device Configuration
The CLDC covers devices below 512k size of memory for the JVM environment and removes JNI, user-defined class loaders, some class verification items of the class verification process (the reason why we MIDlet Preverify MIDlets compiled to be deployed on CLDC devices) and reflection to fit into this small footprint. The JVM on CLDC devices is called KVM (K-VirtualMachine), nicknamed K-VirtualMachine for kilobyte. The application lifecycle concepts from javaTV and the installation concepts of javaPhone found their way into this Connected Limited Device Configuration.

Connected Device Configuration
The CDC covers devices above the 512k size of memory for the JVM environment. The garbage collector is a separate pluggable item into the CVM deployed on CDC devices. The threads are implemented not as OS specific but as fully implemented threads within the CVM and are known as Green Threads. Threads can also be implemented as OS native threads. Class file verification takes place in the CVM on the device just as in Java SE java platform on desktop systems. The JVM on CDC devices is known as a CVM or the C-VirtualMachine.

Java ME profiles

Profiles provide APIs that focus on a specific group of devices. These devices typically have the same or similar user intefaces (screen and inputs), how the device connects to the network, how the device stores data, and etc. Currently, the Profile Name indicates the Family name of this group of devices or a common functionality with this group of devcies. For example, the MIDP (MID Profile) is for the group of Mobile Information Devices. Java ME profiles are:

Mobile Information Device Profile
The Mobile Information Device Profile (MIDP) extends and enhances the CLDC to provide a Java environment to develop applications on Mobile information appliances and voice communication devices. The versions of MIDP are:

- MIDP 1.0
- MIDP 2.0
- MIDP 3.0

Foundation Profile
The Foundation Profile is what can be called a vertical specification profile. A vertical specification profile specifies the targeted family of devices by memory size and the optional profiles that may plugin more functionality in addition to this profile. The GUI APIs are often Profiles plugged in on top of this Profile. Typical rich network devices include set top boxes(iTV), auto-computing(telematics), and etc.

Personal Profile
This Personal Profile provides high web fidelity to device with a rich network connection and can use the heavy weight applet classes that were part of the PersonalJava spec. While Personal Profile, JSR62, supports PersonalJava1.1.x and PersonalJava1.2.x of the PersonalJava Application Environment; Personal Profile 1.1 supports the adoption of the subset of J2SE1.4 APIs as Personal Profile 1.0 was based on the subset of J2SE1.3 APIs.

Personal Basis Profile
This Personal Basis Profile provides a basic graphical user interface, ie the AWT lighweight APIs, on top of the CDC and Foundation Profile and is considered as a subset to the Personal Profiles. This profile also provides the xlet classes required for JavaTV.

Information Module Profile
This Information Module Profile supports embedded network connected devices without a graphical user interface. This profile is a subset of the MIDP1.0 profile without the LCDUI classes of MIDP1.0. The typical devices target are modems, household devices, and industrial metering devices. While IMP is a subset of MIDP1.0, IMP-NG is a subset of MIDP2.0.

Digital Set Top Box Profile
This Digital Set Top Box Profile is a subset of JavaTV and is targeted for the smaller set top boxes that deploy OCAP. Thus, this profile differs from the javaTV(CDC, PP PBP) setup in that its on top of CLDC not CDC to support a smaller device memory foot print. However, several concepts are borrowed from the JavaTV items in Personal Basis Profile such as the xlet lifecycle.

(C) Copyright Nokia 2013. All rights reserved.

How to start

See Getting started with Java ME.

Java ME APIs in Nokia devices

See Additional APIs available for Java ME and Mobile Service Architecture for more information on APIs.

- FileConnection API (JSR-75 Optional package)
  - javax.microedition.io.file
- PIM API (JSR-75 Optional package)
  - javax.microedition.pim
- Bluetooth API (JSR-82)
  - javax.bluetooth
  - javax.obex
- MIDP 2.0 (JSR-118)
  - java.lang
  - java.util
  - javax.microedition.io
  - javax.microedition.lcdui
  - javax.microedition.lcdui.game
  - javax.microedition.media
  - javax.microedition.media.control
  - javax.microedition.midlet
  - javax.microedition.pki
  - javax.microedition.rms
- MMAPI (JSR-135)
  - javax.microedition.media
  - javax.microedition.media.control
  - javax.microedition.media.protocol
- CLDC 1.1 (JSR-139)
  - java.io
  - java.lang
  - java.lang.ref
  - java.util
  - javax.microedition.io
- WSA (RPC Package) (JSR-172)
  - java.rmi
  - javax.microedition.xml.rpc
  - javax.xml.namespace
  - javax.xml.rpc
- WSA (Parser Package) (JSR-172)
  - javax.xml.parsers
  - org.xml.sax
  - org.xml.sax.helpers
- SATSA API (JSR-177)
  - java.lang
  - java.rmi
  - java.security
  - java.security.spec
  - javacard.framework
  - javacard.framework.service
  - javacard.security
  - javax.crypto
  - javax.crypto.spec
  - javax.microedition.apdu
  - javax.microedition.io
  - javax.microedition.jcrmi
  - javax.microedition.pki
  - javax.microedition.securityservice
- Location API (JSR-179)
  - javax.microedition.location
- SIP API (JSR-180)
  - javax.microedition.sip
- M3G API (JSR-184)
  - javax.microedition.m3g
- WMAPI (JSR-205)
  - javax.wireless.messaging
- M2G API (JSR-226)
  - javax.microedition.m2g

(C) Copyright Nokia 2013. All rights reserved.

- org.w3c.dom
- org.w3c.dom.events
- org.w3c.dom.svg
- AMMS API (JSR-234)

Future APIs

eSWT

Nokia is developing Embedded Standard Widget Toolkit (eSWT) for S60 Java platform. For more information see Eclipse eRCP project pages. It is also listed in Nokia Open Source website.

See also

- Java How to's

External links

- Wikipedia's definition
- Sun's overview of Java ME
- Nokia Developer's main Java page
- Nokia Java ME Developer's Library
- Wikibooks's definition of Java ME

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.