

What's new for Java developers in Symbian OS, Version 8.0

Version 1.0

May 2004

1. Introduction

The release of Symbian OS Version 8.0 was announced at 3GSM in February 2004. Version 8.0 brings significant new additions to the Java platform. The new functionality is summarised below.

- Migration to CLDC 1.1
- Implementation of CLDC 1.1 HI Virtual Machine
- Mobile Media API
- File Connection API
- Mobile 3D Graphics API
- JTWI Compliance
- Support for the J2ME UEI interface

Since not all these are self explanatory we will flesh out each in a little bit more detail in the next section.

2. Additions to the Java Platform in Version 8.0

2.1 CLDC 1.1

With the introduction of Symbian OS v8.0 the Java implementation moves from a CLDC 1.0 configuration to CLDC 1.1 (JSR 139). CLDC 1.1 is an incremental release, fully backward compatible with CLDC 1.0. The main changes added by the CLDC 1.1 specification are:

- addition of floating point support
- Calendar, Date and TimeZone classes redesigned to be more J2SE compliant
- various minor library changes and bug fixes
- tightening of the specification

The minimum memory footprint of the CLDC rises from 160 for v1.0 to 192 kilobytes for v1.1.

2.2 CLDC 1.1 HI Virtual Machine

The previous release of Symbian OS added support for Sun's high performance CLDC 1.0 Hotspot Implementation (HI). CLDC HI is a highly optimized Virtual Machine (VM) incorporating many advanced technologies such as Dynamic Adaptive Compilation, previously only available in desktop Java VMs. The many optimisations available in the CLDC HI VM lead to nearly an order of magnitude better performance than the standard [KVM](#). Since Symbian OS, Version 8.0 brings in support for the CLDC 1.1 specification it also upgrades to Sun's CLDC 1.1 HI VM. In addition to supporting the CLDC 1.1 specification the CLDC 1.1 HI Virtual Machine adds in further performance enhancements compared to the CLDC 1.0 HI.

2.3 Mobile Media API

The Mobile Media API (JSR 135) is a J2ME optional package defining APIs to support playback, recording and capture of audio and video media, as well as tone generation. Licensee implementations of JSR 135 have been available on Symbian OS-based phones for some time. From Symbian OS Version 8.0, Symbian are providing a fully-featured MMAPI implementation as standard. One of the main features is that the content types supported are not considered as a closed set but extensible via dynamic loading of plug-ins to the underlying native multimedia framework.

Symbian provides a rich default implementation including a wide range of Controls, including support for RecordControls in the context of `capture://audio` and `capture://video` opening up the possibility of recording video clips from Java for the first time. However, since among the key features of the implementation are extensibility and customizability the functionality available via the Mobile Media API may vary from phone to phone. The important point to note is Symbian's MMAPI imposes few limitations on the licensee regarding multimedia support from Java.

2.4 File Connection API

Version 8.0 brings to Symbian OS an implementation of the File Connection J2ME optional package from JSR 75. The goal of the File Connection API is to provide access to file systems on devices and/or mounted memory cards. File system connectivity is implemented through the Generic Connection Framework via calls of the form `Connector.open("file://...")`

The implementation of File Connection API will for the first time permit MIDlets to write or read files from common storage, allowing files to be shared between MIDlets and native applications. Note however that access to files or directories deemed private or sensitive may be still be restricted (e.g. access to files and directories that are private to another application, system configuration files and OS specific files and directories).

Also `Connector.open("file://...")` is treated as a restricted API under the MIDP 2.0 security model.

2.5 Mobile 3D Graphics

The Mobile 3D Graphics API (JSR-184) implementation provides functionality for rendering 3D animations of complex objects and scenes. JSR-184 offers an easy-to-use high level API (called retained mode access) where the developer works with screen graphs composed of objects, lights and camera, which are then rendered by the system based on the position of the lights and camera. Additionally, JSR-184 provides a low level API interface (immediate mode access) which provides functions similar to an Open GL subset and allows developers to draw objects directly.

Licensees opting to take JSR-184 must provide a native Open GL ES v1.0 library implementation. JSR-184 also requires the floating point support provided by CLDC 1.1

2.6 JTWI compliance

The Java Technology for the Wireless Industry (JTWI) is an initiative aimed at defining an industry-standard Java platform for mobile phones.

The JTWI specification concerns three main areas:

- it provides a minimum set of optional packages (JSRs) that a compliant device should support
- it defines what optional features within these component JSRs must be implemented on a JTWI-compliant device
- it provides clarification of component JSR specifications, where appropriate.

The minimum component APIs required for JTWI compliance are:

CLDC 1.0 or 1.1

MIDP 2.0

Wireless Messaging API

Mobile Media API if multimedia functionality is exposed to Java

The JTWI requirements also impose requirements on implementations. For example the CLDC 1.0/1.1 implementation must allow a MIDlet suite to create a minimum of ten running threads; and the MIDP 2.0 implementation must allow creation of at least five independent record stores. For a full discussion of all the requirements see the [JTWI specification](#).

2.7 UEI Support

The J2ME Unified Emulator Interface specification (UEI) defines a set of command line extension that allows UEI compliant emulators to seamlessly integrate with standard tools and IDEs such as Sun One Studio, Mobile Edition, JBuilder and CodeWarrior, Wireless Edition. Symbian OS, Version 8.0, introduces UEI support into Symbian's generic MIDP emulator, enabling licensees to produce SDKs which more tightly integrate into the popular IDEs.

Symbian's UEI implementation currently supports:

- adding a Symbian MIDP emulator to the IDE's list of installed emulators.

- setting it as the default emulator in the IDE
- running and debugging a MIDlet from within the IDE

3. Discussion

The additions to the Java Platform in Version 8.0 give developers a particularly rich J2ME implementation to target as shown in Figure 1 below.

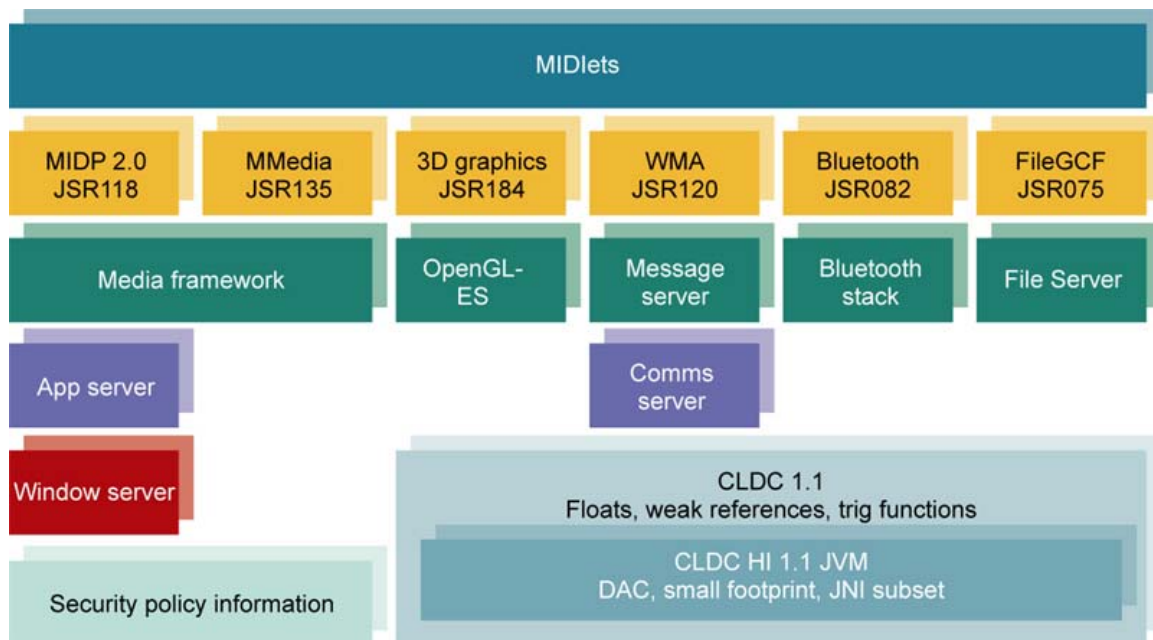


Figure 1 Symbian's Java Platform as of Symbian OS v8.0

For instance, the additions for the first time opens up the possibility that media files, perhaps captured from Java using the Mobile Media API, can be shared with other applications (including native ones) via the File Connection API, not just trapped in the Java sandbox. Gaming applications, already one of the successes of MIDP, will become even richer with the possibility of 3D effects via the Mobile 3D graphics API. And with support for the J2ME UEI interface J2ME development for Symbian OS devices will become even easier through the better integration of SDKs with standard tools. Finally, the adoption of JTWI should assist those developers interested in targeting their MIDlets at the widest possible market.

4. Resources

CLDC 1.1 Specification

<http://jcp.org/en/jsr/detail?id=139>

“The CLDC HotSpot Implementation Virtual Machine”

http://java.sun.com/products/cldc/wp/CLDC-HI_whitepaper-March_2004.pdf

Mobile Media API Specification

<http://jcp.org/en/jsr/detail?id=135>

Working with the Mobile Media API Part 1 and 2

http://www.symbian.com/developer/techlib/papers/java_MIDP.html

File Connection API

<http://jcp.org/en/jsr/detail?id=75>

Mobile 3D Graphics API

<http://jcp.org/en/jsr/detail?id=184>

JTWI Compliance

<http://jcp.org/en/jsr/detail?id=185>

Want to be kept informed of new articles being made available on the Symbian Developer Network?

[Subscribe to the Symbian Community Newsletter.](#)

The Symbian Community Newsletter brings you, every month, the latest news and resources for Symbian OS.