# Oracle® Java Micro Edition Software Development Kit

Release Notes

Release 3.4 for Windows

**E25309-08**

September 2013

**Contents**



## New in This Release

The following items are new in Oracle Java ME SDK 3.4:

- Support for Qualcomm Orion Internet of Everything (IoE) hardware platform, including the ability to attach it to Java ME SDK via a serial (USB) cable.
- Support for Qualcomm Orion IoE device in emulation.
- Support for the latest version of NetBeans IDE (version 7.3.1) and Eclipse IDE (version 4.3).

## Release Highlights

This section summarizes the release highlights.

- **Implementation and support for the Oracle® Java ME Embedded 3.3 runtime for the IMP-NG platform and the Oracle® Java Wireless Client 3.2 runtime for the CLDC platform.**
- **Support for device logging of virtual and physical devices (external devices) in the emulator.**
- **Support for dragging and dropping of JAD and JAR files to run MIDlet suites in the emulator.**
- **Support for installing, running, stopping, and removing applications in the emulator's Application Management System (AMS) user interface.**
- **MIDP emulator now has screen capture capability.**
- **The external events generator now features an ADC tab**.
- **The Eclipse version now supports these features.**
  - Memory Monitor
  - Network Monitor and Network Monitor snapshot

- Custom Device Editor

- **Device addresses can be managed and registered devices can be viewed.** Right-click on the Device Manager icon in the Windows system tray, and choose Managed Device Addresses or Registered devices.
- **Bug fixes and enhancements**.

## Installation Prerequisites

The Oracle Java ME SDK product has three distinct components:

- The Java ME SDK base platform, which includes the runtimes (virtual machines), emulators, libraries, and more.
- A separately-installable, supported IDE, such as NetBeans or Eclipse
- A Java ME SDK plugin for each supported IDE. The plugin extends the IDE so that it can seamlessly access the Java ME SDK features and utilities from the IDE.

---

**Note:** To work with the Oracle Java ME SDK, NetBeans users must apply NetBeans 7.3 Updates. Select Tools --> Plugins in the NetBeans interface.

---

### Supported Platforms

For working with Oracle Java ME SDK, a minimal system configuration is:

- Microsoft Windows XP 32-bit or Windows 7 32-bit and 64-bit with recent service packs.
- Java Platform, Standard Edition Software Development Kit (JDK) version 1.7 with latest updates, or higher. The JDK must be on your Windows path.
- A supported IDE:
  - A recent version of Eclipse Indigo (Eclipse Classic 3.7.2) or Eclipse Juno (Eclipse Classic 4.2.1).
  - NetBeans version 7.1.2, NetBeans 7.2 and 7.3. Download the "All" bundle (contains all components). The Mobility pack must be installed and active.

---

**Note:** To work with the Java ME SDK plugins, you must run NetBeans using JDK version 1.7.

---

To download the latest IDE, go to: `https://netbeans.org`

If you are using an earlier version than 7.1.2, upgrading NetBeans will not work. Please install a clean version of the version you choose.

Alternatively, you can download the Java SE bundle from the NetBeans download site and then install Java EE, Java ME, and the NetBeans plugins separately. This method enables you to download only what you need.

---

**Note:** CPU monitoring and memory monitoring are not supported in NetBeans 7.2. If you need these features use 7.2.1 or higher.

---

### Installing Java ME Plugins

Plugins make Java ME SDK platform features available to a supported IDE. Plugins are delivered in two bundles.

- **Java ME SDK Tools**. Required.
- **Java ME SDK Demos**. Optional, but useful for getting started quickly. The documentation refers to these demos to illustrate features.

> **Note:** The samples do not implement security measures. The "Installation and Runtime Security Guidelines" suggest how to maintain an environment in which sample code can be run safely.

For more information on installing and working with Java ME SDK plugins, see the *NetBeans Developer's Guide* and the *Eclipse Developer's Guide*.

## Known Problems

- **On Device Tooling (ODT) MIDlet loses contact with the Qualcomm Orion IoE board during network profiling**. To work around this, in network mode, set `JC_SOCKET=4`. Or, alternatively, run network profiling in serial mode.
- **Drivers for the Qualcomm Orion IoE embedded device are not signed**. To install the drivers, Windows should be switched to Test mode. For more information, see the *Oracle Java ME Embedded 3.4 Getting Started Guide for the Reference Platform (Qualcomm Orion IoE)*.
- **The Java ME SDK 3.2 plugins do not work with Java ME SDK 3.3.** Although Java ME SDK version 3.3 supports the Oracle® Java Wireless Client 3.2 runtime you must use the version 3.3 plugins for the CLDC platform.
- **Certificates used to sign MIDlets for the Mobile emulator must be generated by JDK 6**. Certificates generated using `keytool` in JDK 7 will not work. (This is not a problem signing IMlets for the IMP-NG emulator. Certificates generated with JDK 7 work with IMP-NG.)
- **Unpredictable emulator behavior if debugger is started twice**. The user must not attempt to reinstall or run an application, if the application has previously been started in debug mode, but the debugger not attached. This scenario is not handled properly and can cause unpredictable emulator behavior.
- **Debugging behavior is not the same on CLDC and CDC**. The main class stops differently for CLDC and CDC when given a new breakpoint with the following parameters:

  Breakpoint Type: Class; Class Name: *; Suspend: Breakpoint thread

  In CLDC execution stops only once, while in CDC it stops every time the base class loads.
- **On the MIDP Emulator and external devices the Output Console Window does not filter Standard Output and Error Output messages**. All messages are shown. Because messages cannot be filtered, error text is not colored red; all message text is black.
- **You cannot watch static variables during an Eclipse debugging session, and sometimes the Variable view cannot show data.**

In the source code, move the mouse over the required variable to inspect the variable value.

- **Running `cref` with a port larger than 65535 results in a misleading error message:** p 1" - port number is smaller than well-known port 1024. The port should be less than 65535.
- **When creating a CDC sample project the New Sample Project wizard Default Platform Selection page does not present a CDC platform**. To work around this problem, choose the sample project, specify a Name and Location, and click Finish (do not attempt to choose the default platform). The sample project is created, but NetBeans indicates errors. In the Projects view, right-click on the project and Select Properties. Select the Platform category and ensure that the platform type is CDC. Choose an appropriate Device and click OK. The project errors are resolved and it is ready to use.
- **On-device profiling and memory monitoring do not work on an external device**.

  Perform these operations with your emulator first, and then test your application on the device.
- **Suspending the Emulator during a Memory Monitor session hangs the emulator.**

  Do not use the Suspend option (F5) while the Memory Monitor is running. If the emulator is hung, open the Windows task manager and stop the emulator process (javaw).

  To switch to another application while the Memory Monitor is running, choose Application > AMS Home (F4), and select a different application.
- **Eclipse plugin for CLDC/MIDP does not support** CDC.
- **Eclipse Juno can't run the Oracle Java ME SDK Online Help if Jetty has been disabled as directed in the Juno installation procedure**. Please refer to the PDF or HTML versions of the Developer's Guide for Eclipse on Windows instead.
- **Eclipse configuration (visible in the Application Descriptor) can display the same device twice.** This is known to happen with external devices. If a device has been already been registered under a given IP address and it is re-registered (because the device or the device manager restarts) multiple instances can appear on the device list. This is a bug in MTJ.
- **Eclipse MTJ loses TrueZIP.jar**. This issue prevents NetBeans project import using File > Import > Java ME > NetBeans project. Restart Eclipse and wait a few minutes for Eclipse to acquire TrueZIP.jar and Eclipse will work correctly.
- **Analog to Digital and Digital to Analog Conversion is not supported on the Raspberry Pi external device**.

## Documentation Accessibility

This release includes PDF documentation in the binary download. For an HTML version of the documentation that is suitable for screen readers, please visit the Java ME Developer Tools documentation page at `http://docs.oracle.com/javame/developer.html` and choose this version of Oracle Java ME SDK.

When using NetBeans, see the online help topic “About Accessibility” for a complete list of keyboard shortcuts. Java ME SDK online help in keyboard shortcuts are summarized in the online help topic “Help Viewer Shortcuts”.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

#### Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

## Installation and Runtime Security Guidelines

The Oracle Java ME SDK requires an execution model that makes certain networked resources available for emulator execution. These required resources might include, but are not limited to, a variety of communication capabilities between the Java ME SDK components. It is extremely important to note that the Oracle Java ME SDK installation and runtime system is fundamentally a developer system that is not specifically designed to guard against any malicious attacks from outside intruders. Given this, the Oracle Java ME SDK architecture can present an insecure operating environment to the Oracle Java ME SDK installation file system itself, as well as its runtime environment, during execution. For this reason, it is critically important to observe the precautions outlined in the following security guidelines when installing and running the Oracle Java ME SDK.

To maintain optimum network security, Oracle Java ME SDK can be installed and run in a “closed” network operating environment, meaning the Oracle Java ME SDK system is not connected directly to the Internet, or to a company Intranet environment that could introduce unwanted exposure to malicious intrusion. This is the ideal secure operating environment when it is possible. Oracle Java ME SDK does not require an “Intranet” connection that supports network connections to systems outside the Oracle Java ME SDK architecture to intra-company resources.

An example of a requirement for an Internet connection is Oracle Java ME SDK running wireless functionality that requires a connection to the Internet to support the communications with the wireless network infrastructure that is part of the Java ME application execution process. Whether or not an Internet connection is required depends on the particular Java ME application running on Oracle Java ME SDK. For example, some Java ME applications can use an HTTP connection. In any case, if the Oracle Java ME SDK is open to any network access you must observe the following precautions to protect valuable resources from malicious intrusion:

- Installing the Java ME SDK Demos plugin is optional. Some sample projects use network access and open ports. Because the sample code does not include protection against malicious intrusion, you must ensure your environment is secure if you choose to install and run the sample projects.
- Install the Oracle Java ME SDK behind a secure firewall that strictly limits unauthorized network access to the Oracle Java ME SDK file system and services. Limit access privileges to those that are required for Oracle Java ME SDK usage while allowing all the bidirectional local network communications that are necessary for Oracle Java ME SDK functionality. The firewall configuration must

support these requirements to run the Oracle Java ME SDK while also addressing them from a security standpoint.

- Follow the principle of "least privilege" by assigning the minimum set of system access permissions required for installation and execution of the Oracle Java ME SDK.
- Do not store any data sensitive information on the same file system that is hosting the Oracle Java ME SDK.
- To maintain the maximum level of security, make sure the operating system patches are up-to-date on the Oracle Java ME SDK host machine.

---

Oracle® Java Micro Edition Software Development Kit, Release 3.4 for Windows
E25309-08