

Oracle® Java Micro Edition Software Development Kit

Release Notes

Release 8 for Windows

E49310-01

April 2014

- ["What's New in This Release"](#)
- ["Installation Prerequisites"](#)
- ["Supported Platforms"](#)
- ["Installing Oracle Java ME SDK 8 Plugins"](#)
- ["Known Java ME SDK Bugs"](#)
- ["Known Java ME Embedded Runtime Bugs"](#)
- ["Other Known Bugs"](#)
- ["Documentation Accessibility"](#)
- ["Installation and Runtime Security Guidelines"](#)

What's New in This Release

The following items are new in the Oracle Java ME SDK 8 release:

- Support for Oracle Java ME Embedded 8.0, including improved integration with external devices, such as the Raspberry Pi and Qualcomm Internet of Everything (IoE) hardware platforms.
- Support for Device I/O API, which provides enhanced device controls and improved input/output (IO) for small embedded devices.
- Support for Connected Limited Device Configuration 8 (JSR 360)
- Support for Java ME Embedded Profile (JSR 361)
- Improved Virtual Machine (VM) configuration
- Improved memory monitoring and CPU profiling on supported external devices
- Terminated support for Mobile (CLDC) and CDC platforms.

Installation Prerequisites

The Oracle Java ME SDK 8 product has three distinct components:

- The Oracle Java ME SDK 8 base platform, which includes the runtimes (virtual machines), emulators, libraries, and more.
- A supported IDE, such as NetBeans 8.0 (installed separately).

- Oracle Java ME SDK 8 plugins for NetBeans IDE 8.0. The plugins extend NetBeans so that it can seamlessly access the Oracle Java ME SDK 8 features and utilities from the IDE.

Note: The NetBeans 8.0 IDE must be run with JDK 7u25 or higher in order to work with Oracle Java ME SDK 8 plugins.

Supported Platforms

The minimum system configuration for working with Oracle Java ME SDK 8 is:

- Microsoft Windows 7 (32-bit or 64-bit) with recent service packs.
- Java Platform, Standard Edition Software Development Kit (JDK) release 7 update 55 or higher.
- NetBeans IDE 8.0 with all the latest patches installed. You can download the latest version at <https://netbeans.org/downloads/>

Installing Oracle Java ME SDK 8 Plugins

Plugins make Oracle Java ME SDK 8 platform features available in NetBeans IDE 8.0. Plugins are delivered in two bundles:

- **Java ME SDK Tools:** This bundle is required.
- **Java ME SDK Demos:** This bundle is optional, but useful for getting started quickly. The documentation refers to the demos to illustrate features.

For more information on installing the Oracle Java ME SDK 8 plugins, see the *Oracle Java Micro Edition Software Development Kit Developer's Guide for Windows*.

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.

Known Java ME SDK Bugs

The following bugs are known to directly affect Oracle Java ME SDK 8:

Java ME SDK uninstaller does not kill all related processes

If NetBeans IDE or any of the Java ME SDK tools (such as, Custom Device Editor, Java ME SDK Update Center, and so on) are running when you uninstall Java ME SDK, some files may not be removed. In this case, you can manually remove the files that are left over from Java ME SDK. To make sure that all Java ME SDK files are removed, exit NetBeans IDE and any Java ME SDK standalone applications.

Not possible to uninstall Java ME SDK under a different user

Only the user that installed Java ME SDK can uninstall it. Trying to uninstall under a different user will not work. Make sure that you perform uninstallation under the same user that was used to install Java ME SDK.

Known Java ME Embedded Runtime Bugs

The following Java ME Embedded 8 bugs may affect users of Oracle Java ME SDK 8:

[Raspberry Pi] If Raspberry Pi with a running IMlet is powered off, when you power it back on, the IMlet appears as if it is running, and there is no way to stop or remove it
When you power on and connect Raspberry Pi back, the Device Manager does not read the list of suites installed in the device and the current IMlet statuses. This is why the IMlet appears to be in running state. However, since the actual task does not exist, attempting to stop it leads to an error message.

As a workaround, you should end the device-manager.exe process before powering Raspberry Pi back on.

IllegalStateException is thrown when calling AccessPoint(<ap>) if <ap> has already been connected by NetworkInterface.connect(<ap>)

An `IllegalStateException` may be thrown when calling the `AccessPoint()` method if the access point specified has already been connected using the `NetworkInterface.connect()` method. However, if the access point was previously connected by another call to `AccessPoint.of()`, no exception is thrown.

[Raspberry Pi] Device Manager does not detect the case when Raspberry Pi becomes unavailable

The Device Manager does not detect the case when Raspberry Pi becomes unavailable. It still reports that the board is connected at least 3 minutes after disconnecting.

SPIDevice.begin() or SPIDevice.end() does not cause an IllegalStateException, even if a previous transaction has already been started or ended

The `SPIDevice.begin()` and `SPIDevice.end()` methods do not cause an `IllegalStateException` to be thrown, even if a previous transaction has already been started or ended. Instead, an `IOException` is thrown.

The I2CDevice.write() method returns the wrong value

When calling the `I2CDevice.write(int subaddress, int subaddrSize, java.nio.ByteBuffer src)` method, it may return the wrong value.

[Raspberry Pi] The UART.read() method does not read the requested number of bytes. The ReceiveTimeout and TriggerLevel properties cannot be set.

The `UART.read(java.nio.ByteBuffer dst)` method returns bytes available in the internal buffer. Once timeout ends, the method will return bytes read during the timeout or the requested number of bytes.

Newly installed LIBlet remains in the storage if IMlet installation failed

In this case, you should try to install the IMlet again.

It is possible to install a LIBlet's service when another required LIBlet's service is absent

Currently, installation succeeds for a LIBlet service when another required LIBlet's service is not installed. This should not be allowed.

[Raspberry Pi] A Raspberry Pi board may not be automatically recognized by the Device Manager

If you start Java on a Raspberry Pi, then start the Device Manager, open the Device Connections Manager, and click **Add**, the board may not be present in the **IP Address** or **Host Name** drop-down list. You should enter the IP address or host name manually.

Other Known Bugs

The following bugs are not directly associated with the Java ME Embedded runtime, but may affect users of Oracle Java ME SDK 8:

Running the Java Card Platform Simulator (cref) with a port larger than 65535 results in a misleading error message

If you set the port number to N larger than 65535, the port number is calculated as $N - 65536$. If the result is larger than or equals to 0, but less than 1024, then `cref` produces an error message. If the result is larger than or equals to 1024, then `cref` is started on the resulting port number.

For example, setting the port number to 65536 results in the port number of 0 and `cref` exits with the following error message:

```
"-p 0" - port number is smaller than well-known port 1024
```

Setting the port number to 66560 starts `cref` on port number 1024.

You should avoid using port numbers larger than 65535 for the `cref` command.

Documentation Accessibility

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 8.

When using NetBeans IDE 8.0, see the About Accessibility help topic for a complete list of keyboard 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

Oracle Java ME SDK 8 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 Oracle Java ME SDK 8 components. It is important to note that the Oracle Java ME SDK 8 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 8 architecture can present an insecure operating environment to the Oracle Java ME SDK 8 installation file system itself, as well as its runtime environment, during execution. For this reason, it is important to observe the precautions outlined in the following security guidelines when installing and running the Oracle Java ME SDK 8.

To maintain optimum network security, Oracle Java ME SDK 8 can be installed and run in a *closed* network operating environment, meaning the Oracle Java ME SDK 8 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 8 does not require an Intranet connection that supports network connections to systems outside the Oracle Java ME SDK 8 architecture to intra-company resources.

An example of a requirement for an Internet connection is Oracle Java ME SDK 8 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 8. For example, some Java ME applications can use an HTTP connection. In any case, if the Oracle Java ME SDK 8 is open to any network access you should be aware of 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 Oracle Java ME SDK 8 behind a secure firewall that strictly limits unauthorized network access to the Oracle Java ME SDK 8 file system and services. Limit access privileges to those that are required for Oracle Java ME SDK 8 usage while allowing all the bidirectional local network communications that are necessary for the Oracle Java ME SDK 8 functionality. The firewall configuration must support these requirements to run Oracle Java ME SDK 8 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 Oracle Java ME SDK 8.
- Do not store any sensitive data on the same file system that is hosting Oracle Java ME SDK 8.
- To maintain the maximum level of security, make sure the operating system patches are up-to-date on the Oracle Java ME SDK 8 host machine.

Oracle® Java Micro Edition Software Development Kit, Release 8 for Windows
E49310-01

Copyright © 2009, 2014, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.