

Note

Before using this information and the product it supports, read the information in “Notices” on page 15.

Second Edition (February 2008)

This edition applies to version 6.2 of the IBM WebSphere Everyplace Micro Environment and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright International Business Machines Corporation 2007. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Introduction	1
About this document	1
About the J9 runtime environment	1
About J9 class libraries	1
About installer packages	1
Package contents	2
 Tasks	 3
Installing the runtime	3
Uninstalling the runtime	3
 J9 command options	 5
Common options used with j9.exe	5
Advanced options used with j9.exe	7
J9 V2.4 GC command line options	10
 Further information	 13
 Notices	 15
Trademarks	17

Introduction

About this document

This document can help you install the Personal Profile (PPro) 1.1 configuration that you need to run a J9 VM executable on your Windows® Mobile 6.0 device.

Note: This document assumes a certain level of knowledge about working with Windows Mobile 6.0 devices. For example, if you need further information on how to install files on your device, please refer to the Windows Mobile 6.0 documentation.

About the J9 runtime environment

What is a J9 runtime?

The J9 virtual machine (VM), the core of IBM® WebSphere® Everyplace® Micro Environment (WEME), the IBM implementation of the Java™ Virtual Machine Specification, Version 1.4.2. A Java virtual machine executes machine instructions, known as bytecodes, typically compiled from Java language source code. For more on the Java Virtual Machine Specification, refer to <http://java.sun.com/docs/books/vmspec/>.

The J9 VM and Java Class Libraries (JCL) comprise the J9 runtime environment. The J9 runtime environment is Java 2 Platform, Micro Edition (J2ME™) compliant and contains PPro 1.1 based technologies.

IBM WebSphere Everyplace Micro Environment is a certified Java Powered product, developed under an agreement between IBM and Sun Microsystems.

About J9 class libraries

Which J9 class libraries are available for a Windows Mobile 6.0 devices?

The following J9 class library is available for your device:

jclfoun11_24 is an implementation of the J2ME Foundation Profile 1.1 specification (JSR 219). **iveppro11** is an implementation of the J2ME PPro specification (JSR 62).

About installer packages

What is an installer package?

The installer packages contain the components of the WEME PPro 1.1 runtime for your Windows Mobile 6.0 device. Executing the installer package file starts the InstallShield application and guides you through the install.

Type of Package	File name	Description	Space required to install
Runtime Package for Windows Mobile 6.0 CDC 1.1/FOUNDATION 1.1/PPRO 1.1:	ibm-wemewm60-arm-ppro11-6.2-YYYYMMDD-#####-###.exe	These runtime packages includes the: <ul style="list-style-type: none"> • J9 VM runtime • Foundation 1.1 class libraries • PPro 1.1 class libraries 	3 MB

Package contents

Each package includes the following files:

Found in Program Files\J9\PPR011\bin\

- iveppro11.dll
- j9vmall24.dll
- jclfoun11_24.dll
- j9jit24.dll
- j9.exe
- j9w.exe
- java.properties

Found in Program Files\J9\PPR011\lib

charconv.zip

Found in Program Files\J9\PPR011\lib\jclFoundation11

- classes.zip
- locale.zip

Found in Program Files\J9\PPR011\lib\jclFoundation11\ppro11

- ppro-ui.jar
- ppro-extras.jar

Found in Program Files\J9\PPR011\lib\security

- cacerts
- java.policy
- java.security

Tasks

Installing the runtime

Use the following steps to deploy this WEME runtime for Windows Mobile 6.0 Foundation 1.1/Personal Profile 1.1 on your target device.

1. Download the product installer from this Web site: <http://www.ibm.com/software/wireless/weme/>.
`ibm-wemewm60-arm-ppro11-6.2-YYYYMMDD-#####-###.exe`
2. Use the Extraction Wizard to extract the files. Extract the files to the default location indicated.
3. Select the **Explore** icon in **ActiveSync** to open a **File Explorer** window for your mobile device.
4. Use **File Explorer** to go to the root on your mobile device and copy the CAB file that was installed on your host computer to your mobile device.
5. Using **File Explorer**, select the CAB file that you copied to your mobile device and it will install the J9 executable to the device.
6. The default installation is to the folder `\Program Files\J9\PPR011`. The J9 can be in any directory as long as the `\bin` and `\lib` subdirectories remain intact. Note, however, that moving the J9 directory will most likely effect registry entries and shortcuts; so it isn't recommended.

Uninstalling the runtime

Follow these steps on your mobile device to uninstall the runtime:

1. From the **Start** button, select **Settings** tab, then select **System** tab, then select **Remove Programs** to run the uninstaller.
2. Select **Uninstall IBM wm60_arm_ppro11**.
3. Select the package you want to remove. A dialog box prompts you to confirm the removal.
4. Select **Yes** to proceed with the uninstall.

J9 command options

This section discusses the common and advanced J9 v2.4 command line options used with **j9.exe**.

Note: The Windows Mobile platform requires that you use fully qualified path names. Relative path names will not work.

Common options used with **j9.exe**

This section describes common J9 v2.4 command line options used with **j9.exe**.

Syntax:

j9 -jcl:ppro11 [*options...*] **-classpath** [*options*] *classname* [*arguments...*]

j9 -jcl:ppro11 [*options...*] **-jxe** *:jxe_file* [*arguments...*]

-? or **-help**

Displays help for J9 standard command options.

-classpath *path* or **-cp** *path*

Either **-classpath** *path* or **-cp** *path* can be used to set a class path for this invocation of J9. The final value of **-classpath** is determined:

- If the **-classpath** option is set, its value is used.
- If the **-classpath** option is not set, and the CLASSPATH environment variable is set, its value is used.
- If neither of the preceding are set, the current directory (.) is used.

If the class path includes:

- Multiple class path entries, separate them with a semicolon “;”.
- A JAR, ZIP or JXE file, add the full name of the file to the class path.
- CLASS files, specify the top-level directory of the CLASS file tree.

Example: **-classpath** \myclasses;\myjars\foo.jar

CAUTION: The J9 class libraries and the J9 VM are not compatible with other vendors' class libraries. Because it is possible to have more than one runtime environment installed on your host computer, make sure that you do not mismatch these libraries when specifying the class path. In particular, if your CLASSPATH environment variable is set, ensure that other vendors' libraries are not on it.

Note: **java** and **javax** class packages must be on **-Xbootclasspath**, not **-classpath**.

-jxe:jxe_file

Reads the specified JXE file, searching for the classes in this file. All classes found in the JXE file are placed at the end of the boot path.

Example: **-jxe:hello.jxe**

Note: When using the **-jxe** option, do not specify the **startup** class. Specify the **-jxe** option as the last option on the command line.

J9 command options

Note: It is recommended that you use **-classpath** (where applicable) or **-Xbootclasspath** if the JXE file contains boot classes. See the **-Xbootclasspath** option for details.

-Dprop=value

Sets the value of a system property.

Example: `-Dmy.property=some.value` Sets the value of `my.property` to `some.value`. If no value is given, **-Dprop** sets the value to null. To set values for multiple system properties, repeat the option statement, using a space to separate statements.

Example: `j9 -Dprop1=val1 -Dprop2=val2 -Dprop3=val3`

Note: Spacing is important in this option's syntax. There is no space between the initial **-D**, its property argument, the equal sign, or the value argument.

Example: `-Dname=John_Smith`

Note: If *value* contains spaces, enclose the option in double quotes.

Example: `"-Dmy.property=value with space"`

-debug:options

Enables debug, Java Debug Wire Protocol (JDWP) standard options.

-jcl[: extralib1 : extralib2]

When necessary, this command option is used to specify which JCL shared library will provide JNI natives for the class library Java code. For example, if both Foundation 1.1 and PPro 1.1 class libraries are available, to use the PPro 1.1 class libraries you must specify `-jcl:ppro11`.

With J9 2.4 the `-jcl` command line option is extended with the `extralib` parameters. The `extralib` parameters are used to specify directories off of the core library. Anything in the specified directory is added to the bootstrap classpath.

Example: `-jcl:foun11:ppro11`

Loads the JCL Foundation 1.1 library as normal and also includes `jclFoundation11/ppro11/*` on the bootstrap classpath.

A new system property set by the VM, `com.ibm.util.extralibs.properties` contains the value of the extra parameters, in this case: `com.ibm.util.extralibs.properties="ppro11"`.

If the `-jcl` option is used without indicating a `-Xbootclasspath:path`, the value for `path` is assumed to be `$JAVAHOME/lib/jclLibraryName/classes.zip`. However, if the class libraries are stored in a non-default location, you must include the `-Xbootclasspath:path` to direct the VM to the `classes.zip` file. See the **-Xbootclasspath** option for details.

Note: If the **-Xbootclasspath** and the `-jcl` VM options are mismatched, the VM generates an Incompatible class library error.

-verbose[:class, gc, stack, sizes]

Enables verbose output. Parameters are:

- **class** displays each fully-qualified class name as it is loaded (that is, enable verbose class loading). This is the default value.
- **gc** displays garbage collection information.
- **stack** displays stack information.

- **sizes** displays default VM sizes.

-verify

Enables class file (bytecode) verification.

Note: The **-verify** option is true by default. To disable bytecode verification, specify **-noverify**.

-version

Each VM build is identified by a version string of the form:

YYYYMMDD_####_flags Example: 20051103_03795_1EdFGq The first 8 digits indicate the date the VM was built. The next 5 digits indicate the build ID. The flags indicate the configuration:

1st letter:

- l: little endian
- b: big endian
- L: 64-bit little endian
- B: 64-bit big endian

2nd letter:

- E: emulated Floating Point Unit (FPU)
- H: hardware FPU

3rd letter:

- s: static linkage
- d: dynamic linkage

4th letter:

- C: CLDC
- F: Foundation
- S: J2SE

5th letter:

- M: Desktop GC
- m: Tiny GC
- G: Embedded GC

6th letter:

- i: no JIT
- a: AOT only
- r: large JIT
- q: small JIT
- V: MicroJIT
- R: large JIT + MicroJIT
- Q: small JIT + MicroJIT
- A: AOT + MicroJIT

-X Prints help for non-standard (advanced) options.

Advanced options used with j9.exe

This section describes advanced J9 v2.4 command line options used with **j9.exe**. These command line options are non-standard and subject to change without notice.

J9 command options

-Xbootclasspath:*path*

Sets the bootstrap class path to *path*.

Example: `-jcl:ppro11 -Xbootclasspath:%JAVAHOME%\lib\jclFoundation11\classes.zip`

Note: When using this command line option, the `-jcl:LibraryName` option must also be used, as shown in the above example, to indicate which class library natives the application should use.

-Xbootclasspath/p:*path*

Prepends the classes in *path* to the bootstrap class *path*. This option is useful for applying temporary fixes to application classes and/or adding to the bootstrap classpath.

-Xbootclasspath/a:*path*

Appends the classes in *path* to the bootstrap class path. This option is useful for applying temporary fixes to application classes and/or adding to the bootstrap classpath.

-Xdbg:*options*

Enables standard JDWP debug options.

-Xdbginfo:*symbol_file_path*

Enables the debug info server.

-Xrdbginfo:*host:port*

Enables the remote debug info server.

-Xrunjdw:*options*

Enables standard JDWP debug options and starts a JDWP server.

-Xfuture

Turns on strict class file format checks. These checks enforce closer conformance to the class file format specification.

-Xss*x* Sets the maximum Java thread stack size to *x*.

-Xmso*x*

Sets the operating system thread stack size to *x*.

-Xint Runs interpreted only.

-Xjit[:count=*x*, code=*x*]

With no parameters, **-Xjit** enables the JIT. Useful parameters are: **count**=*x*, where *x* is the upper limit of the number of times a method is invoked before it is compiled.

Example: `-Xjit:count=0`, forces the JIT to compile everything on first execution.

code=*x*, where *x* sets the size of the JIT code cache, in kilobytes.

Example: `-Xjit:code=1024`, sets the size of the JIT code cache to 1MB.

The code cache will grow dynamically if required.

-Xoptionsfile=filename StartupClass [*arguments...*]

J9 VM Version 2.4 on all platforms supports an option file for the purpose of reducing the length of the command line.

Example: `%JAVAHOME%\bin\j9 -Xoptionsfile=vm.options com.ibm.myapps.MyApp1`

- An option file is a text file with one option per line.

- Lines starting with # are ignored and can be used for comments.
- The \ character can be used as a continuation so that a single option can span multiple lines.
- The following command line options must be converted into their J9 internal form when used in the options file:

Command line option	J9 internal form
-analyze	-Xanalyze:NULL
-analyze:	-Xanalyze:
-classpath path	-Djava.class.path=path
-dbginfo:	-Xdbginfo:
-debug:	-Xdbg:
-jcl:config	-Xjcl:jclconfig_24
-noverify	-Xverify:none
-rdbginfo:	-Xrdbginfo:
-verify	-Xverify
-verify:	-Xverify:
-Xrunjdpw:	-Xdbg:

- The following options are ignored when listed in the options file and must be entered on the command line:

```

-jar
-jxe
-jxe:
-jxspace:
-Xoptionsfile=

```

Note: Embedded options files are not supported.

- Undocumented options are ignored when listed in the options file. To be used, they must be added to the command line.
- All other options, such as **-D**, **-Xint**, **-Xmx** etc. are the same on the command line and in the options file.

Environment variables are not supported in the options file. For example the following works on the command line but not in the options file :

```
-Dmy.property=some.value
```

Options listed in the option file override options on the command line regardless of their position.

Example: vm.options file:

```

-Xint
-Xanalyze:st=true,ia=192.168.1.100,ms=100000
-Djava.security.manager
-Djava.security.policy=my.policy
#-Djava.class.path=my.jar

```

-Xrundll[:options]

Loads helper libraries, such as those used with JVMPI.

J9 V2.4 GC command line options

The Garbage Collector for WEME runtime for Windows Mobile 6.0 Foundation 1.1/PPro 1.1 provides global GC with compaction. The following related command line options are provided:

Option parameter key

x integer value in bytes, or append with 'k' or 'M' for large values

-Xmxx

Sets memory object heap memory size to *x*.

Xmx >= **NewSpace** size plus **OldSpace** size

Scavenger enabled: minimum size 1536 bytes on 32-bit architectures, 6072 bytes on 64-bit architectures

Scavenger disabled: minimum size 512 bytes on 32-bit architectures, 2048 bytes on 64-bit architectures

-Xmsx Sets the initial memory size to *x*.

Scavenger enabled: minimum size 4 kilobytes on 32-bit architectures, 8 kilobytes on 64-bit architectures **Xms** >= **Xmn** + **Xmo**

Scavenger disabled: minimum size 4 kilobytes on 32-bit architectures, 8 kilobytes on 64-bit architectures **Xms** >= **Xmos**

-Xmosx

Sets the initial **OldSpace** size to *x*.

Note: Minimum size 512 bytes on 32-bit architectures, 2048 bytes on 64-bit architectures

-Xmoxx

Sets the maximum **OldSpace** size to *x*.

-Xmox

Sets the initial and maximum **OldSpace** size to *x*.

Note: Attempts to set **-Xmo** and **-Xmos**, or **-Xmo** and **-Xmox** are rejected

-Xmcax

Sets the RAM class segment increment to *x*.

RAM class segments contain the portion of the Java classes that needs to be modified at runtime, such as the pointers to the Class Loader, super classes, implemented interfaces, statics, first instance and so forth. There is at least one RAM class segment per Class Loader. If more space for the RAM classes is needed, the J9 VM allocates a new segment in the same size.

-Xmcox

Sets the ROM class segment increment to *x*.

The code of Java classes loaded from class files is stored in a special ROM class segment type called Dynamically Loaded Classes, whose size is determined by this parameter. There is at least one ROM class segment per Class Loader. If more space for the dynamically loaded classes is needed, the J9 VM allocates a new segment in the size determined by the Class Loader.

Note: This option does not apply to classes loaded from a JXE file. The `rom.classes` entry in the JXE file is mapped directly to a ROM class segment.

-Xmoix

Sets the **OldSpace** increment to *x*. This value is used to expand the **OldSpace**. A value of 0 means no expansion is allowed. If **-Xmoi** is not specified, there are no restrictions on the expansion size of **OldSpace**.

-Xmnsx

Sets the initial **NewSpace** size to *x*.

On 32-bit architectures the minimum size = 1024 bytes; on 64-bit architectures the minimum size = 4096 bytes.

-Xmnxx

Sets the maximum **NewSpace** size to *x*.

-Xmnx

Sets both the initial **NewSpace** size and maximum **OldSpace** size to *x*.

Note: Attempts to set **-Xmn** and **-Xmns**, or **-Xmn** and **-Xmnx** are rejected.

J9 command options

Further information

If you need more information or have questions about our product:

In the United States and Canada, call 1-800-IBM-CALL (1-800-426-2255)

In all other countries, you can submit your questions on the web at:
<http://www.ibm.com/software/pervasive/support/questions.shtml>

You might find helpful information at the following websites or newsgroup:

- <http://www.ibm.com/software/wireless/weme/>
- <http://www.ibm.com/software/wireless/wece/>
- <http://www.ibm.com/embedded>
- newsgroup: ibm.software.websphere.studio.device-developer

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Notices

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Department LZKS
11400 Burnet Road
Austin, TX 78758
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, or other countries, or both:

IBM
IBM logo
Everyplace
WebSphere

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Windows[®] is a trademark of Microsoft Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.