WebSphere® Studio Device Developer

J9 VM Runtime Guide
For Windows 2000 / Windows XP Professional

First Edition (June 2004)

© Copyright International Business Machines Corporation 2004. All rights reserved.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

© Copyright IBM Corp. 2004

Chapter 1. Introduction

About this document

This document will help you customize your environment to run a J9 VM executable. It will also provide assistance on how to complete runtime image and execute the desired class library configuration on the target, using **Microsoft Windows 2000 or Windows XP Professional**.

About the J9 runtime environment

The J9 VM, the core of **WebSphere Everyplace Micro Environment** (WEME), is IBM's implementation of the **Java Virtual Machine Specification, Version 1.3**. 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 **Connected Limited Device Configuration** (CLDC) and **Connected Device Configuration** (CDC) based technologies. In addition, the **WebSphere Everyplace Custom Environment** (WECE) is a combination of the J9 VM and IBM custom libraries.

The WEME product is supported on a variety of:

- Operating systems (including Microsoft® Windows, Linux, PalmOS, OSE, Rex, VxWorks, PocketPC, Symbian, QNX and Nucleus)
- Hardware architectures (including Intel x86, xScale. ARM, MIPS, SH4, and PowerPC)

WebSphere Everyplace Micro Environment is a certified Java Powered product, developed under an agreement between IBM and Sun Microsystems. Deployment of applications or devices with Workplace Client Technology requires an appropriate deployment license from IBM or one of IBM's partners.

© Copyright IBM Corp. 2004

Chapter 2. Tasks

The instructions provided in this section will help a developer create and deploy a complete runtime image and execution environment-which includes the J9 VM and desired class library configuration(s)-on the target.

J9 runtimes included

In order to execute the J9 VM on the target, one of the runtimes for a Microsoft Windows target is required. For **Windows 2000 or Windows XP Professional**, the runtime package is found:

J2ME Profile	Location
CDC/Foundation 1.0	C:\Program Files\IBM\DeviceDeveloper\wsdd5.0\ive-2.2\runtimes\win32\x86\Foundation10
CLDC 1.1/MIDP 2.0	C:\Program Files\IBM\DeviceDeveloper\wsdd5.0\ive-2.2\runtimes\win32\x86\Midp20
Personal Profile 1.0	C:\Program Files\IBM\DeviceDeveloper\wsdd5.0\ive-2.2\runtimes\win32\x86\PPro10

Updating the software

Additional libraries can be installed as features, using the WebSphere® Studio Device Developer Update Manager.

Customizing the J9 runtime

The J9 runtime is a feature-rich VM that can be customized to fit the needs of a specific application. Depending on which runtime is being used, all of the files may or may not be in the directory structure.

Before loading the runtime files onto the target, some of the files can be stripped out from the runtime image in order to minimize its size.

Removing Unused Runtime Files

Not all of the files and subdirectories are required to run the J9 VM. If trying to minimize the size of the runtime, many of the files included in this package can be removed.

Development Files

The first components that should not be loaded to the target devices are the following subdirectories, which are used for development purposes only:

- `/exe`
- `/include`
- `/lib`

© Copyright IBM Corp. 2004

Unused DLL Files

As a general rule, any component for a shared library, or DLL file, that is not being used can be removed from the target image.

For instance, if debugger and MicroAnalyzer support are no longer needed on the target, then the corresponding shared library files (for example, `j9dbg22.dll` and `j9prf22.dll`) can be removed from the runtime image.

Unnecessary Class Libraries

Another simple way to reduce the runtime is class library native files. The application likely only uses one class library at a time; therefore, only the specific supporting native shared library files are needed on the target.

For instance, if the application uses the Foundation 1.0 library, only the `j9foun10_22.dll` file is required on the target, among all of the Java™ Class Library native files.

This example runtime allows for the execution of a very simple Java application, the ″HelloWorld″ sample program, on a target using the Foundation 1.0 class library configuration.

Note: The ″HelloWorld″ sample program is included with the WebSphere Studio Device Developer workbench. Examples are accessed by WebSphere Studio Device Developer plug-in.

The minimized target runtime in its entirety consists of the following files:

`j9.exe` J9 executable program

`j9dyn22.dll` Dynamic class loader.

`j9gc22.dll` Garbage Collector.

`j9hook.dll` Link library for hookable components

`j9prt22.dll`: J9 port library.

`j9thr22.dll`: Thread library native.

`j9vm22.dll`: J9 VM library.

`j9zlib22.dll`: Zlib data compression library.

`jclfoun10_22.dll`: Foundation 1.0 natives

`jsig.dll`: Signal Handler

As can be seen, removing unused or unnecessary files obviously minimizes the size of the runtime. Of course, as features or capabilities are added, more libraries must be added to the target to support them.

Chapter 3. J9 command options

Common options

The following table contains common J9 v2.2 command line options.

```
J9 - VM for the Java(TM) platform, Version 2.2
(c) Copyright IBM Corp. 1991, 2004 All Rights Reserved
Target: {Encoded Build Number} (Windows XP 5.1 build 2600 Service Pack 1 x86)
IBM is a registered trademark of IBM Corp.
Java and all Java-based marks and logos are trademarks or registered
trademarks of Sun Microsystems, Inc.
Usage: j9 [options] classname [args...]
Usage: j9 [options] -jxe:<jxeFile> [args...]
[options]
    -classpath <path>
    -cp <path>       set classpath to <path>.
    -jxe:<jxeFile>   run the named jxe file.
    -D<prop>=<val>   set the value of a system property.
    -debug:<options> enable debug, JDWP standard <options>.
    -jcl:<config>    specify which JCL DLL to use (e.g. cdc, cldc, ...).
    -verbose         [:class,gc,stack,sizes]
                     enable verbose output(default=class).
    -verify          enable class file verification.
    -X               print help on non-standard options.
```

Refer to the following table for a more detailed description of each of these options:

Syntax	Description
-? or -help	This command option displays help

© Copyright IBM Corp. 2004

Syntax	Description
-classpath	This command option sets the class path for this invocation of J9. The final value of `-classpath` is determined by: 1. If the `-classpath` option is indicated, its value is used. 2. Otherwise, if the CLASSPATH environment variable is set, its value is used. 3. If both are unspecified, the value ″.″ is used. If the class path includes: • More than one class path entry, you should separate them with your operating system’s path separators (for example, on Windows, use semicolons.) • A JAR, ZIP or JXE file, you should add the full name of the file to the class path. • CLASS files, you should specify the top-level directory of the CLASS file tree. **Example: (on windows)** `-classpath c:\ive\lib\classes.zip; c:\myclasses;c:\myjars\foo.jar` **CAUTION:** **The J9 class libraries (classes.zip) and the J9 VM are not compatible with other vendors’ class libraries. It is possible that you might have more than one runtime environment installed on your host computer. You must make sure that you do not mismatch these libraries when specifying the class path.**
-cp<path>	Set classpath to <path> This is equivalent to -**classpath**.
-jxe:<jxeFile>	This command option reads the specified JXE file, looking for the classes in this file. All classes found in the JXE are placed at the end of the ″boot path.″ **Note:** When using the **-jxe** option, do not specify the startup class. **Example:** -jxe:hello.jxe **Note:** The **-jxe** option must be the last option on the command line. **CAUTION:** **The -jxe option is not the preferred way to run an application. It is best to use -classpath (where applicable) or -Xbootclasspath: (if the .jxe contains boot classes).**
-Dprop=<val>	This command option sets the value of a system property. For example, `-Dmy.property=some.value` sets the value of `my.property` to `some.value`. `-Dprop` sets the value to null. You can use multiple instances of this option by repeating the option statement separated with a space. **Example:** `j9 -Dprop1=val1 -Dprop2=val2 -Dprop3=val3` **Note:** Spacing is important in this option’s syntax. There is never a space between the initial -D, its property argument, the equals sign, or the value argument. **Example:** -Dname=″John Smith″

Syntax	Description
-debug:<options>	This command enables debug, **Java Debug Wire Protocol** (JDWP) standard <options>
-jcl:<config>	This command option specifies which JCL DLL to use (which JNI natives are used by the class library Java code.) If you use the **-jcl:<config>** without indicating a **-Xbootclasspath: <path>**, then the value for the **-Xbootclasspath: <path>** is assumed (`%JAVAHOME%/lib/jclLibraryName/classes.zip`). However, if the class libraries are stored in a non-default location, then you must include the **-Xbootclasspath: <path>** in order to direct the VM to the **classes.zip** file. **Note:** If the **-Xbootclasspath** and the -jcl VM options are mismatched, the VM will generate an ″Incompatible class library″ error. The possible library arguments are: • -jcl:foun10 (jclFoundation10 class library) • -jcl:ppro10 (jclPPro10 class library) • -jcl:midp20 (jclMidp2.0 class library) • -jcl:max (jclMax class library) • -jcl:rm (jclRM class library) **Note:** Not all these options are available for all platforms.
-verbose[:class, gc, stack, sizes]	This command option turns one of the following: • `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	This command option enables bytecode verification. The `-verify` option is on by default. To disable bytecode verification specify `-noverify`.
-X	This command option prints help on non-standard options.

Advanced options

The following options are non-standard and subject to change without notice:

```
-Xbootclasspath:<path>     set bootstrap classpath to <path>
-Xbootclasspath/p:<path>   prepend <path> to bootstrap classpath
-Xbootclasspath/a:<path>   append <path> to bootstrap classpath
-Xrun<dll>[:options]       Load helper libraries, such as those used with JVMPI
-Xint             run interpreted only
-Xnoaot           do not run precompiled code
-Xfuture          enable strictest checks, anticipating future default

Arguments to the following options are expressed in bytes.
Values suffixed with "k" (kilo) or "m" (mega) will be factored accordingly.
-Xmca<x>          set RAM class segment increment to <x>
-Xmco<x>          set ROM class segment increment to <x>
-Xmn<x>           set initial/maximum new space size to <x>
-Xmns<x>          set initial new space size to <x>
-Xmnx<x>          set maximum new space size to <x>
-Xmo<x>           set initial/maximum old space size to <x>
-Xmos<x>          set initial old space size to <x>
-Xmox<x>          set maximum old space size to <x>
-Xmoi<x>          set old space increment to <x>
-Xms<x>           set initial memory size to <x>
-Xmx<x>           set memory maximum to <x>
-Xmr<x>           set remembered set size to <x>
-Xmrx<x>          set maximum size of remembered set to <x>
-Xmso<x>          set OS thread stack size to <x>
-Xiss<x>          set initial java thread stack size to <x>
-Xss<x>           set maximum java thread stack size to <x>
-Xmine<x>         set minimum size for heap expansion to <x>
-Xmaxe<x>         set maximum size for heap expansion to <x>

Arguments to the following options are expressed as a decimal from 0 to 1.
A value of 0.3 represents a request of 30%
-Xminf<x>         minimum percentage of heap free after GC
-Xmaxf<x>         maximum percentage of heap free after GC

Arguments to the following options are expressed as decimal numbers.
-Xgcthreads<x>          set number of GC threads
-Xnoclassgc             disable dynamic class unloading
-Xclassgc               enable dynamic class unloading
-Xalwaysclassgc         enable dynamic class unloading on every GC
-Xnocompactexplicitgc   disable compaction on a system GC
-Xcompactexplicitgc     enable compaction on a system GC
-Xcompactgc             enable compaction
-Xnocompactgc           disable compaction
-Xlp                    enable large page support

-Xdbg:<options>               enable debug, JDWP standard options
-Xrunjdwp:<options>           enable debug, JDWP standard options
-Xdbginfo:<symbol file path>  enable debug info server
-Xrdbginfo:<host><port>       enable remote debug info server
```

Refer to the following tables for a more detailed description of each of these options:

Syntax	Description
-Xbootclasspath:<path>	This command option sets the bootstrap classpath to <path>. **Note:** When using this command line option, the **-jcl:LibraryName** option must be used to indicate which class library natives the application should use: `-Xbootclasspath:c:\ive\lib\jclCldc\classes.zip` If you use the **-jcl:<path>** without indicating a -**Xbootclasspath: <path>**, then the value for the **Xbootclasspath: <path>** is assumed (`%JAVAHOME%/lib/jclLibraryName/classes.zip`). However, if the class libraries are stored in a non-default location, then you must include the **-Xbootclasspath: <path>** in order to direct the VM to the **classes.zip** file.
-Xbootclasspath/p:<path>	This command option prepends <path> to the bootstrap classpath. **Note:** This option is useful for applying temporary fixes and/or adding to the bootstrap classpath.
-Xbootclasspath/a:<path>	This command option appends <path> to the bootstrap classpath. **Note:** This option is useful for applying temporary fixes to application classes and/or adding to the bootstrap classpath.
-Xrun<dll>[:options]	Load helper libraries, such as those used with JVMPI.
-Xint	This command option runs interpreted only. This disables both the JIT and AOT support.
-Xjit:code=z	This command is used to set the size of the JIT code cache, in kilobytes (K). In other words, code=1024 would result in a 1 MB cache. On x86 platforms, the cache will grow dynamically if required.
-Xnoaot	This command option prevents running the precompiled code.
-Xgcpolicy:optthruput	This command option sets no scavenger and no concurrent mark. This is the default. This garbage collection policy delivers very high throughput to applications, but at the cost of occasional pauses, which can vary from a few milliseconds to many seconds, depending on the size of the heap and the quantity of garbage. Garbage collection cycles introduce occasional unexpected pauses in the execution of application code. Because applications grow in size and complexity, and heaps become correspondingly larger, this garbage collection pause time tends to grow in size and significance. This garbage collection policy is the default.

Syntax	Description
-Xgcpolicy:optavgpause	This policy enables two garbage collection technologies whose aim is to minimize pause times; namely scavenger and concurrent mark. Provided the weak hypothesis that most objects die young holds true then the scavenger helps reduce pause times by concentrating the garbage collection effort on the nursery since it is there that most recyclable space will be found. Rather than occasional but lengthy pause times to collect the entire heap, the nursery is collected more frequently and provided the nursery is small enough, pause times will be comparatively short. However, over time the tenure area may become full if too many objects live too long. So in order to minimize the pause time when a collection of the tenure area is necessary, concurrent mark is enabled to perform some garbage collection activities concurrently with normal program execution. The optavgpause option substantially reduces the time that is spent in these garbage collection pauses, in addition to limiting the effect of increasing heap size on the length of the garbage collection pause. This option is particularly relevant to configurations that have large heaps. However, with the reduced pause time, you might experience some reduction of application throughput, which varies from application to application. This command option sets scavenger and concurrent mark.
-Xfuture	This command option enables the strictest checks, anticipating a future default.

Syntax	Description
-Xmca<x>	This command line option 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, like the pointers to the class loader, super classes, implemented interfaces, statics, first instance and so forth. There is at least on RAM class segment per Class Loader. If more space for the RAM classes is needed, then the J9 VM allocates a new segment in the same size.
-Xmco<x>	This command line option sets the ROM class segment increment to <x>. The code of Java classes loaded from .class files is stored in special ROM Classes segment types called Dynamically Loaded Classes whose size are determined by this parameter. There is at least on ROM class segment per Class Loader. If more space for the dynamically loaded classes is needed, then the J9 VM allocates a new segment in the size determined by the class loader. **Note:** This parameter does not apply for classes loaded from a **Jxe**. The ″rom.classes″ entry in the Jxe is mapped directly to a ROM class segment.

Syntax	**Description**
-Xmn<x>	This command line option sets the initial and maximum **NewSpace** size to <x>. **Note:** This sets the size of the **NewSpace** and **SurvivorSpace**. This command is only relevant if **-Xgcpolicy:optavgpause** is set.
-Xmns<x>	This command line option sets the initial **NewSpace** size to <x>.
-Xmnx<x>	This command line option sets the maximum **NewSpace** size to <x>.
-Xmo<x>	This command line option sets the initial and maximum **OldSpace** size to <x>.
-Xmos<x>	This command line option sets the initial **OldSpace** size to <x>.
-Xmox<x>	This command line option sets the maximum **OldSpace** size to <x>.
-Xmjit<x>	This command line option enables the micro JIT. **Note:** The micro JIT is not enabled by default. You must explicitly enable the micro JIT on platforms that support it or J9 will run in interpreted mode only.
-Xmoi<x>	This command line option sets the **OldSpace** increment to <x>. This value used to expand the **OldSpace**. A value of 0 means no expansion is allowed. If -Xmoi is not specified then there are no restrictions on the expansion size of **OldSpace**.
-Xms<x>	This command line option sets the initial memory size to <x>. **Note:** This option is the same as **-Xmos<x> plus -Xmnx<x>**.
-Xmx<x>	Memory object heap memory size -**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
-Xmr<x>	This command line option sets the remembered set size to <x>.
-Xmrx<x>	This command line option sets the maximum size of remembered set to <x>.
-Xmso<x>	This command line option sets the OS thread stack size to <x>.
-Xiss<x>	This command line option sets the initial Java thread stack size to <x>.
-Xss<x>	This command line option sets the maximum Java thread to <x>.
-Xmine<x>	This command line option sets the minimum size for heap expansion to <x>.
-Xmaxe<x>	This command line option sets the maximum size for heap expansion to <x>.

Syntax	Description
-Xmdx<x>	This command line option sets the default memory space maximum size. **Xmdx <=Xmx**. Defaults to -**Xmx/8. -Xmx - -Xmdx** is the space available for user memory spaces created with the **com.ibm.oti.vm.MemorySpace** class provided in **jclRM**. **Note:** This is ignored if *-Xresman* is also used.
-Xresman<x>	This command line option enables resource managed support with the **com.ibm.oti.vm.MemorySpace** class in **jclRM**.

Syntax	Description
-Xminf<x>	This command line option sets the minimum percentage of heap free after GC.
-Xmaxf<x>	This command line option sets the maximum percentage of heap free after GC.

Syntax	Description
-Xgcthreads<x>	This command line option sets the number of GC threads
-Xnoclassgc	This command line option disables dynamic class unloading
-Xclassgc	This command line option enables dynamic class unloading only on class loader changes (default)
-Xalwaysclassgc	This command line option enables dynamic class unloading during global collection
-Xnocompactexplicitgc	This command line option will disable compaction on a system GC.
-Xcompactexplicitgc	This command line option will enable compaction on every system GC.
-Xcompactgc	This command line option will enable compaction.
-Xnocompactgc	This command line option will disable compaction.
-Xlp	This command line option will enable large page support.

Syntax	Description
-Xdbg:<options>	This command line option enables standard Java Debug Wire Protocol (JDWP) debug options.
-Xrunjdwp:<options>	This command line option enables standard JDWP debug options. **Note:** Start a JDWP server. For more information on the JDWP options, see the **Connection and Invocation Details** web site.
-Xdbginfo:<symbol file path>	This command line option enables the debug info server.
-Xrdbginfo:<host>:<port>	This command line option enables the remote debug info server.

Appendix. Additional information

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/wme/
- http://www.ibm.com/software/wireless/wce/
- 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® might 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 might have patents or pending patent applications covering subject matter 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, 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

© Copyright IBM Corp. 2004

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 information. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this information 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.

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 document 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 measurement 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.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

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.

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. *2004* 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:

Microsoft, Windows, Windows NT, Windows Mobile and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

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