BEAJRockit

Release Notes

Contents

BEA JRockit 1.4.2 SDK Release Notes

Version 1.4.2_08 R24.5.0

This document contains important details for this version of BEA JRockit 1.4.2 SDK. It contains information on the following subjects:

License Agreement

The BEA JRockit 1.4.2 SDK is subject to the terms and conditions of the BEA JRockit Binary License Agreement.

Platform Support

This release is certified on the platforms listed on the Supported Platforms page.

Installation

BEA JRockit 1.4.2 SDK is available as part of the WebLogic Platform 8.1 suite (Service Pack 3 and higher) and as a standalone application. For instructions on installing the standalone JRockit application, please see Installing BEA JRockit 1.4.2.

BEA JRockit Version Numbering

This version of the BEA JRockit SDK includes the BEA JRockit JVM Version R24.5.0 compliant with Java 2 Platform, Standard Edition, v 1.4.2_08. This is represented by the BEA JRockit version number 1.4.2_08 R24.5.0.

Documentation

This release of BEA JRockit SDK includes a complete documentation set comprised of these documents:

- Release Notes
- Introduction to BEA JRockit 1.4.2 SDK
- Using BEA JRockit 1.4.2 SDK
- Tuning BEA JRockit JVM
- Developing Applications

Additionally, the following web-only documents are available:

- Supported Platforms, which lists all platforms supported by this version of BEA JRockit SDK.
- Installing BEA JRockit 1.4.2 SDK, which contains acquisition and installation instructions for this specific release.
- Developers FAQ, which contains frequently-asked questions (and answers) about BEA JRockit.

Copies of all BEA JRockit 1.4.2 SDK documents can be found at:

http://edocs.bea.com/wljrockit/docs142/index.html

New Features

For details on these new features and other features, please refer to New Features in BEA JRockit 1.4.2 SDK in Introduction to BEA JRockit 1.4.2 SDK.

New Features in 1.4.2_08 R24.5.0

No new features added.

New Features in 1.4.2_05 R24.4.0

- Memory Leak Tool available in the BEA JRockit Management Console.

New Features in 1.4.2_04 R24.3.0

- Improved 32-bit and 64-bit runtime performance
- Improved startup performance
- Support for Native POSIX Thread Library on Linux
- Dynamic garbage collection
- Dynamic heap resizing

Most Recent Changes

This section describes the most recent changes of BEA JRockit.

Changes in 1.4.2_08 R24.5.0

Table 1 lists the changes have been made for this release.

Table 1 Changes in 1.4.2_08 R24.5.0

Change Request ID	Description
CR188112, CR203172, CR210772, CR212228, CR223211	Property differences between BEA JRockit and Sun Hotspot have been eliminated. See System Property Changes for a list and description of these changes.
CR192114	With earlier versions of BEA JRockit, you were unable to use multiple `-Xmanagement:class` options as startup parameters. As of this service pack, the product has been enhanced to make this possible.
CR193488	Calling `java.util.TimeZone.getDefault()` in the constructor of a `java.util.logging.LogManager` could cause a `"java/lang/UnsatisfiedLinkError: getSystemTimeZoneID"`.
CR195355	The implementation of the mmap syscall on RedHat Enterprise Linux 3.0 was mapping the lowest page in the virtual address space (starting at 0x00000000), breaking anything trying to access to this page to generate a SIGSEGV; for example JRockit null pointer exceptions or a forced dumping. This issue hs been resolved by marking the null pointer memory area inaccessible before it can be compromised.
CR196799	JRockit crashes when calling `Class.forName` with initialize set to `true` if it has previously been set to `false`. This has now been fixed.
CR196953	When specifying `-Xmx4g` on the command line on 32 bit architectures, the actual heap size was silently set lower than specified. With this release, a warning is printed when that situation occurs..
CR199462	When BEA JRockit crashed when running on an IA64 Linux 2.6-based systems, the stack trace that printed in the crash dump would be incomplete. The product has been corrrected so that the stack trace is now complete.
CR201407	Improved handling of dumping in SIGBUS situations.
CR201837	When running WLS 81 SP2 and JRockit with OptimizeIt's Code Coverage, null pointer exceptions were thrown in the security providers. This has now been fixed.

Table 1 Changes in 1.4.2_08 R24.5.0

Change Request ID	Description
CR204907	Applications that over-allocate direct NIO buffers (e.g. `ByteBuffer.allocateDirect()`) caused occasional crashes because of the native out of memory situation that this can cause.
	A new option, `-XXMaxDirectMemorySize = <bytes>`, has been added to protect against direct overallocation, which gives a more orderly `java.lang.OutOfMemoryError: Direct buffer memory`. Performance will be affected negatively when reaching the `-XXMaxDirectMemorySize` limit.
CR204924	`FileChannel.read(ByteBuffer dst, long position)` doens't return -1 at end of file. This has now been fixed.
CR205415	A variable memory leak in gencon garbage collector has been fixed.
CR206249	FileInputStream.skip with relative negative offset now valid.
CR206352	When running with `-Xmanagement`, there was a problem measuring per-thread CPU usage for SLES9. This has now been fixed.
CR206974	When BEA JRockit crashed when running on a Linux 2.6-based system, crash dumps were cut off whenever the crash dumping code tried to read from read-protected memory. The crash dumping code can now handle read-protected memory and is more likely to produce complete crash dumps.
CR207466	`OutOfMemoryError` thrown when deploying a war in WLS that has an empty jar file. Now, when trying to open an empty zip file, an `IOException: Zip file is empty` is thrown.
CR207941	WLS 8.1 SP3 - Jrockit crashes when calling `socket.setNetworkInterface(netInterface)`. This has now been fixed.
CR208575	BEA JRockit no longer fails to start on Windows 2003 SP1 on Itanium.
CR208909	`MulticastSocket.joinGroup(SocketAddress, NetworkInterface)` now works.
CR209140	Some customers, when trying to start their server would occasionally experience server hang-ups during the code generation phase of startup. This issue has been resolved by employing a reprocess to pre-resolve classes before code generation.

Table 1 Changes in 1.4.2_08 R24.5.0

Change Request ID	Description
CR211041	Running the Memory Leak tool did not display "points-to" information when using the singlecon garbage collector. This has been fixed.
CR211941	JNI frames were faulty for methods that either returned an object or were synchronized. This has now been fixed.
CR212375	There was a crash in the garbage collector (mmGetObjectSize) that has been fixed.
CR214100	Conversions to Strings given byteArrays greather than 16Mb gives `java.nio.BufferOverflowException`. This has now been fixed.
CR214493, CR216259	A Java leak detected with ObjectMonitors has been fixed.
CR214904	Some customers, when running BEA JRockit 1.4.2_04 with a high load on Linux AS 3.0 using native threads and a parallel garbage collector would encounter socket exceptions. These exceptions corrected in this service pack by adding a missing synchronization to socket read/writes.
CR215827	More `-X` options have been added to `Xusage.txt` (`java -X` output).
CR215830	The `-Xrs` option has been added and will be interpreted as `-Xnohup`.
CR216628	When trying to run WebLogic Server and 15,000 jsp pages on BEA JRockit, a customer was running out of native memory. This issue was resolved.
CR220413	JRockit now prevents that someone alters the parent field in `java.lang.ClassLoader`, as JRockit is depending on it for native structure recycling.
CR223299	BEA JRockit was crashing when an internal ZipFileP, despite being freed upon zip closing, was still referenced and used by ZipInputStream. This issue was resolved by ensuring that the input stream no longer holds native data.
CR223967	Application program no longer gets `NullPointerException` in `Thread.wait()` at high concurrency levels.
CR224233	Dumping behavior has been improved to better ensure that a crash file is generated whenever a crash occurs.
CR224673	Changed all references from BEA Weblogic JRockit(R) to BEA JRockit(R). (See also System Property Changes.)

Table 1 Changes in 1.4.2_08 R24.5.0

Change Request ID	Description
CR224844	Accessing a non-NULL Java array with index value of 0 no longer gives a `NullPointerException`.
CR225142	Changes to `java.vendor*` system properties have been reverted. (See also System Property Changes.)
CR225405	JRockit no longer accepts shortened versions of command line switches.
CR228112	A crash in caFPUPushExtendedDouble in linux environments is fixed.
CR229176	A rare error when using apache-xalan xml/xslt transforms is fixed. Xalan uses the swap bytecode (seldom used by javac) which triggered bad behaviour within BEA JRockit.
	The issue was tracked here as well: http://issues.apache.org/jira/browse/XALANJ-1778
CR229549	JRockit no longer crashes on zip-files with unusual extra data.
CR229722	JRockit no longer core dumps on stack overflow on Windows 2003 SP1 on Itanium."
CR231193	An optimization recursion issue could cause stack overflow, e.g. crash dumps with a recursive call stack in `computeDomFrontiers`. This has now been fixed.
CR232187	ObjectStreamClass.lookup() could hang if a class explicitly defines its serial version UID in a static field and also looks up its serial version UID in its static initializer. SUN bug 4803747 fixed.
CR234462	With earlier versions of BEA JRockit that supported the JRockit Runtime Analyzer, if you had an expired `jrockit.license` file and tried to start a JRA recording, the JRA recording would fail silently. Now, when this happens, an informative error message is printed.
CR234711	JRockit can now be linked with C-programs that also make use of pthread_create.
CR235032	Certain options that were available in BEA JRockit 1.4.1 are not available in BEA JRockit 1.4.2. Please refer to Changes Between BEA JRockit 1.4.1 and 1.4.2 Releases for a list of these options.

Table 1 Changes in 1.4.2_08 R24.5.0

Change Request ID	Description
CR235315	When synchronizing on a thread and trying to perform `System.exit`, the system could hang. We now removed unnecessary locks during shutdown sequence to ensure that we can perform shutdown properly.
CR235819	After some bounds checking, a `memmove()` of the source array to the destination array occurs. If that `memmove()` happens while the garbage collector or heap compaction is moving objects around, BEA JRockit might copy a pointer after the object it points to has moved, but before the pointer was updated. This would result in a bogus pointer being copied and the VM crashing. This condition has been fixed.
CR236917	Differences between the Sun JDK implementations of both the `getMethod()` and `getMethods()` of and JRockit's implementations of those methods have been resolved.

System Property Changes

This section describes, by associated platform and Change Request, changes to system properties in this service pack. The purpose of these changes was to remove any differences between BEA JRockit system properties and those used by the Sun JDK.

All Platforms (Windows 32- and 64-bit; Linux 32- and 64-bit)

This section describes changes to system properties which are valid on all platforms.

Changes Based Upon CR203172

- **Change:** Added missing JRockit-specific system property:

  `java.vm.vendor.url.bug = "http://support.bea.com";;`

- **Previous Behavior:** Did not exist

- **Current Behavior:** `"http://support.bea.com";;`

Changes Based Upon CR224673

- **Change:** Changed all references from "BEA Weblogic JRockit(R)" to "BEA JRockit(R)":

  `java.vm.name = "BEA JRockit(R) 1.4.2_08 JVM R24.5.0-46"`

- **Previous Behavior:** `BEA WebLogic JRockit(TM) 1.4.2_05 JVM R24.4.0-1"`

- **Current Behavior:** `BEA JRockit(R) 1.4.2_08 JVM R24.5.0-46"`

Changes Based Upon CR223211 and CR203172

- **Change:** Syncronized locale initialization code with Sun 1.4.2

  `user.country`

  `user.language**user.variant`

  `file.encoding`

  `file.encoding.pkg`

- **Previous Behavior:** Could be wrongly initialized in some locales

- **Current Behavior:** Initialized in the same way as Sun J2SDK 1.4.2

Windows platforms (Windows 32- and 64-bit)

This section lists and describes system property changes available only on Windows 32- and 64-bit platforms.

Changes Based Upon CR203172

- **Change:** Changed initialization rules for `user.home` property

- **Previous Behavior:** Only initialized by environment variable `"USERPROFILE"`

- **Current Behavior:** Initialized by Windows registry; fallback on environment variable `"USERPROFILE"`

Changes Based Upon CR203172

- **Change:** Changed initialization rules for `user.name` property

- **Previous Behavior:** Only initialized by system call

- **Current Behavior:** Initialized by environment variable `"USERNAME"`; fallback on system call

Linux Platforms (linux32, linux64)

This version of BEA JRockit contains no Linux-specific system property changes.

Changes in 1.4.2_05 R24.4.0

Table 2 lists the changes made for BEA JRockit 1.4.2_05 R24.4.0.

Table 2 Changes in 1.4.2_05 R24.4.0

Change Request ID	Description
N/A	The non-standard tuning option to disable fat lock spinning has been changed to `-XXdisableFatSpin`.
CR125647	Dumping thread list through java plugin within a browser should not make the browser die.
CR128853	System properties differs compared to Hotspot.
CR129229	Java plugin with Netscape 7.1 and Mozilla 1.5 requires libgcc_so.1.so.
CR131705	BEA installer on windows fails to check for sufficient launch conditions on startup.
CR134874	`ArrayStoreException` when adding a PortalFramework Book to a Page array. Fixed by changing a < to a > in an array store check.
CR172057	Deadlock causing assert when running with hprof and sampling enabled.
CR172574	Include missing JNI support, i.e. jawt and jvm includes and libs.
CR173561	The amount of free memory in the nursery was sometimes incorrectly reported, causing a JRA recording to fail at point of parsing.
CR174135	`IOException` on windows caused by Win32 pipe.readv/writev not keeping contract about NEVER reading writing to/from IOVEC entry n+1 if entry n was not emptied/filled completely.
CR174583	Optimizer incorrectly removes empty loop.
CR175303	Minor usability issues in Console. On the Select Trigger page in the Notification Rule Wizard it is not clear what unit the limits of different triggers is expressed in and on the Memory tab, the text about suggesting heap size is in a different font than the rest of the text on the page.
CR175304	PipeLeftOverInputStream will lose data if original stream raises `IOException`.
CR175608	Floats get mixed up by asynchronous exceptions.
CR175612	Ctrlhandler memdump reports incorrect results on win64.

Table 2 Changes in 1.4.2_05 R24.4.0

Change Request ID	Description
CR175773	Mystic whitespace diffs in Xusage.txt for Linux builds.
CR175779	JRA does not display GC-backtrace for parallel collector.
CR176464	Print address of context struct in crash dumps.
CR176476	OOME on suse64 with -Xgc:gencon in garbage collection test.
CR176729	Default nursery size for client mode differed in documentation from code implementation.
CR176741	Multibyte characters are broken in `java.io.FileNotFoundException` message.
CR177194	BEA installers for Linux do not clean-up all files after aborted installation.
CR177326	FileNativeIO.open on directories will cause a file handle leak.
CR177515	Missing error code led to assertion failure when running JCK tests.
CR178207	SysconfCrash: Crashing during dumping when printing CPU count on linux.
CR178359	Stack dependency problems due to lack of dependency checks.
CR178567	JRockit crashes when unable to access /proc.
CR178772	BEA jre installer (silent mode) fails to install with space character in path.
CR178774	BEA sdk installer on win2003_ia64 isn't able to open jar-files.
CR178796	In the Management Console CPU load of 100% is not plotted in Processor chart.
CR179005	Overwriting spilled floating point variables.
CR179100	Some lines in Xusage.txt are longer than 80 characters.
CR179217	Multiple threads can call abort() during a crash causing incomplete dumps.
CR179223	Should be able to start with an 8k nursery.
CR179435	During parallel marking we crash on illegal class block when forcing an evacuation.
CR180681	Anti-aliased graphs can't be used over remote X connections.

Table 2 Changes in 1.4.2_05 R24.4.0

Change Request ID	Description
CR180864	Leaking guard pages from the thread stacks when they die causing us to try to throw an `OutofMemoryException`. When trying to generate code for the exception throwing we crash. Several different types of crashes but they all seem to be the result of the JVM running out of memory to malloc.
CR181302	Broken stack float parameters in invoked methods causing inconsistency.
CR181771	Class.getMethod sometimes throws exception incorrectly for interfaces. Problem finding clone-methods in interfaces.
CR182008	The java lock code for contended locks on SMP machines have been re-written to enhance performance on heavily contended locks. Please note that the unsupported `-XX` locking flags have changed.
CR182014	JNI - FindClass(NULL) causes dump
CR182214	Environment variable _JAVA_OPTIONS works with Sun's JVM but not with BEA's JVM
CR182835	Null pointer exception in native code due to writing outside bit sets.
CR183047	Codecache reads wrong idx-file on Posix.and hence does not load any data.
CR183981	JRockit doesn't honor socket connect time-out value.
CR184314	`-Xmanagement` does not work with Resin 3.0.7.
CR184513	Adding useful information to be displayed at fatal exits.
CR185170	Using BEA installer with -Djava.util.prefs.SystemRoot.location=system set the .systemPrefs directory is still placed in JAVAHOME.
CR185273	Codepatching of undoing of devirtualization is unsafe.
CR186166	Licensing system for protected JRockit features, e.g. JRA and Memory Leak.
CR187216	Typo in Direct-X-Buffer.put(..)
CR187927	JRockit 81SPx - JNI code fails with JRockit.
CR188046	Replace JFree libraries from JRockit console & JRA.
CR188132	Random hangs with threads stuck in garbage collection. This is caused by not clearing exceptions thrown during garbage collection.

Table 2 Changes in 1.4.2_05 R24.4.0

Change Request ID	Description
CR188225	Print glibc identification information in JRockit text dump.
CR188226	Discovered non-removed blank lines in Xusage.txt on windows.
CR188272	JVM crash in codegen code: if a method is being generated and the exception structure needs to be reallocated at the same time an other thread has access to the old exception pointer, the second thread could crash while reading from freed memory.
CR188932	Update the server-side of the console comm protocoll to support memleak.
CR189369	Upgrade to 1.4.2_05-fcs.
CR189490	JRockit: Error stopping Windows service. JRockit returns an NT dialog to the effect that "beasvc.exe encountered a problem and needs to close". The console window prints the error "System error 109 has occurred. The pipe has been ended." But the dialog hangs.
CR189969	Using Method.invoke() with more than 8 arguments on IA64 occasionally resulted in some of the passed arguments values being incorrect. More precisely, the 9+ arguments of elementary type and size less than 64bits.
CR190677	We sometimes miss that somebody has interrupted the process due to a thread race.
CR190791	JRA GC graph should cover most of chart area.
CR190877	Running with beanshell works with Hotspot but not with JRockit.
CR190946	`org.codehaus.groovy.classgen.RuntimeIncompleteClass-Exception` when trying to run Codehaus Groovy scripts.
CR191000	Various small lir2native issues on ia32.
CR191750	Buffer overruns and possibly erroneous results in HIR2MIR value numbering.
CR191917	`-XXpreopt` fails on arithmetic right shifts with rhs too long.
CR192446	`NewStringUTF()` should create a `java.lang.String` object from an array of UTF8 characters. It does not specify how to handle the extensions described by encodings above 0x7f (0x80 - 0xff). These characters, which happen to coincide with latin-1, slip through without harm.

Table 2 Changes in 1.4.2_05 R24.4.0

Change Request ID	Description
CR192918	Management Console persistence doesn't work with headless mode. Now it does, so that one may log data and review it later without running a graphical console with a display.
CR193722	Dump when oomdiagnostics=true due to conflict with setting up the memory leak detecting system.
CR193894	Running the Management Console in headless mode on a system without a display required also setting `-Djava.awt.headless=true`. Now there is no dependency on AWT when running in headless mode.
CR194163	Random deadlocks in java.util.Properties when using the Management Console.
CR194182	JRockit crashes when NPE is thrown during debugging.
CR194298	Unable to increase java heap size beyond 1574304kb with /3GB parameter in boot.ini on IA-32 Windows. The java.dll and verify.dll files were based differently between 1.4.1 and 1.4.2, and loaded at different times. Their position in process memory constrained the heap size, since the heap demands consecutive memory. The files have been re-based to a position where they will have minimum influence on heap allocation and heap size
CR194539	Unable to acquire more than 2.9 GB java heap with hugemem kernel on RHEL 3.0
CR194953	The previous release of JRockit did not ship a working jvm.lib.
CR195293	Null pointer exception in native code at jchars_eq_istr.
CR195903	Slow memory allocation in code generation.
CR195905	JRockit dumping on Red Hat AS3.0 (upgrade 2). Set the eip to point to the first valid java location when throwing `ArrayOutOfBoundsException` in `-Xdebug` mode. **Note:** The `-Xdebug` option should not be used in a production environment.
CR196789	(*env)->FindClass() from JNI_OnUnload fails to find class.
CR197011	Make sure that hir2mir in can't elongate live ranges.
CR197200	Can't run the management console due to a bug in value numbering.
CR197368	Crash dumping uses makecontext. Makecontext may only setup args of type int.

Table 2 Changes in 1.4.2_05 R24.4.0

Change Request ID	Description
CR197592	JRockit core dump on RH3.0 (upgrade 2) /IA64 for integration bvt test.
CR198330	Time in GC does not get registered in JRockit management console properly.
CR198550	Crash when re-virtualizing method-calls when newly loaded classes re-implement methods of other loaded classes from a common super class.
CR199161	Two graphics libraries (JCommon and JFreeChart) distributed under the terms of the GNU Lesser General Public License have and used by the JRockit Management Console have been replaced with GreyChart, an open source graphics library distributed under the terms of a BSD-style license.
CR199200	Licensing system fails to find license file.
CR199650	Inform about JRockit behavior and suggest proper action when memory message is printed due to JRockit trying to allocate more memory (outside the Java heap) and failing
CR199651	JMAPI fails to deliver GC event notifications.
CR199682	Set explicit file permissions for payload files not copied directly from Sun JDK.
CR200247	JRockit process crashes with JVMPI profilers (optimizeit & jprobe).
CR200303	SerialVersionUID incorrect for classes with fields with international character names.
CR200313	Crashes on Linux 64. Using mprotect instead of mmap/munmap when possible stopped the crashes and prevented us from leaking guard pages.
CR200320	Win64 crash in javax/imageio/stream/MemoryCache.disposeBefore due to either compiler change or compiler setting change. Some native (vm) code now uses NAT-bits, and we never had handling of this before.
CR200475	Remove errorneously included man page man/man1/javaws.1 from linux64 shipments.
CR200484	Remove JRA encryption.
CR200487	Add additional Management Console launcher 'jrcc' in addition to 'console'.

Table 2 Changes in 1.4.2_05 R24.4.0

Change Request ID	Description
CR200703	i18n-problems with javaws, zh_CN.GB2312 and Red Flag, i.e. asian characters are not displayed correctly on Red Flag due to font problems. Adding the correct font file fixed the problem.
CR200727	Create one off patch: SerialVersionUID incorrect for classes with fields with international character names
CR200731	libnative_chmod.so should be executable (755).
CR200746	Same problem as 200703, but on SuSE Linux.
CR201175	Scanning of objects during mark phase (during garbage collection) cannot handle uninitialized classes.
CR201418	JRockit does not build or install properly on SuSE Linux ES8 with obsolete RPM version 3.0.6.
CR202425	Crash during code optimization of methods when removal of a lock region was incomplete.
CR203604	Regression error in new Java lock code.
CR205071	Change JRockit EULA (end user licence agreement) displayed in the installer. The change in the JRockit EULA is to allow redistribution.
CR205354	"promotion failed" displayed as reason for garbage collection for several old space collections in a row. This should not be possible. We miss to unset the signal "promotion failed".

Changes in 1.4.2_04 R24.3.0

Table 3 lists the changes made for BEA JRockit 1.4.2_04 R24.3.0.

Table 3 Changes in 1.4.2_04 R24.3.0

Change Request ID	Description
CR121049	JRockit may, under certain circumstances, crash when throwing `java.lang.LinkageError`.
CR124597	The `Exception.printStackTrace()` for NullPointer, Arithmetic (divide by zero) and ArrayIndexOutOfBounds does not produce stack trace information.
CR125239	JRockit Fails to Load net.dl `javax.naming.directory` API may fail with `java.lang.UnsatisfiedLinkError: no net in java.library.path`.
CR125387	Call java.nio.channels.DatagramChannel.receive(ByteBuffer) doesn't block even though blocking has been configured with DatagramChannel.configureBlocking(true).
CR129563	JRockit may crash when generating native code for obfuscated byte code.
CR132438	Process reaping broken on Linux (Sun Bug 4482614)
CR137057	JRockit can crash when HotSpot-sampling very large methods for optimization.
CR136236	JRockit may crash during compilation of very large JSPs.
CR136462, CR173177, CR180853	JRockit may crash when the garbage collector is inspecting the stack of a thread that is simultaneously returning from a JNI functions with object parameters.
CR172229	Cannot execute code in Eclipse scrapbook.
CR177403	The implementation of the debugging interface JVMDI has been improved.
N/A	Start-up time with `-Xdebug` reduced.

Table 3 Changes in 1.4.2_04 R24.3.0

Change Request ID	Description
N/A	The userRoot and systemRoot location in the Windows registry for User and System Preferences is now the same as that used by the Sun SDK/JRE. Any existing user or system preferences created by JRockit 1.4.2_03 can be moved to the common Sun/BEA JRockit userRoot and systemRoot by using the java.util.prefs.Preferences API to export/import them to/from an intermediate XML file.
N/A	Code generation optimizations can be disabled by specifying so in the JRockit Monitoring and Management API (JMAPI).
N/A	The gencopy (-Xgc:gencopy option) garbage collection algorithm has been removed. For an alternative garbage collection method, please see the Using BEA JRockit 1.4.2 SDK document.
N/A	In client mode (-client), 64 MB is no longer the limit for maximum heap size (-Xmx), instead the same rules that apply to the server mode (-server) apply to -client. For additional information, please refer to Setting the Maximum Heap Size.

Changes Between BEA JRockit 1.4.1 and 1.4.2 Releases

The following options were supported in BEA JRockit 1.4.1 but not in BEA JRockit 1.4.2.

- Xnativethreads
- Xthinthreads
- Xallocationtype
- Xalloctype
- Xgc:gencopy
- XXcodecoverage
- XXnonparallelgencon
- XXjradelay
- XXjrarecordingtime
- XXjrafilename
- XXjranonativesamples
- XXfatlockreads

- XXfatlockyields

- XXfatlockspins

- XXenablefatspin

- XXspinsbeforeinflation

- XXspinsbeforeyield

- XXspinsbeforeread

- XXspinsbeforeread

Known Issues

The following issues are known in the 1.4.2_05 and 1.4.2_08 release:

Change Request ID	Description
CR199664	JRockit can't mmap more than 2GB on SuSE
	The Java heap is limited to approximately 2 GB on SUSE Linux 8 and 9. JRockit needs contiguous virtual memory for the Java Heap. These operating systems load libraries on the 1 GB boundary.
CR213685	Java stack traces are limited to 32 frames. This may make identifying the cause of a java.lang.StackOverflowError difficult.
CR244389	RHEL 2.1 cannot produce core files larger that 2 GB even though the process size may be up to 3 GB. Core files are created by the OS when JRockit dumps because of a bug in JRockit or in third-party JNI code. The core files are used by BEA support and engineering to find the reason of a crash. Therefore, BEA cannot support customers with a JRockit process size exceeding 2 GB.
	BEA recomends upgrading to a later version of RHEL.
CR248558	When starting WebLogic Server 8.1 SP5 as a Windows Service, with the bundled BEA JRockit 1.4.2_08 (R24.5.0), JRockit crashes during startup. Contact BEA Support for a patch.
N/A	Some JVM options are case sensitive. This is particularly noteworthy since some non-standard options are written in a new form, for example, -XXdisableFatSpin.

Change Request ID	Description
A signal being sent to wrong thread on Linux 2.6 kernels on Itanium (2.6.11 and previous Itanium kernels) might cause random hanging or crashing JVM.	BEA JRockit is known to hang or crash on 2.6 kernels on Itanium, due to a bug in the Linux 2.6.11 (and previous) kernels. The bug is in the kernel `sigprocmask()` syscall and will be fixed in the 2.6.12 kernel.
	We cannot estimate how often the bug actually occurs, but under different circumstances it has been seen as often as once per hour or as seldom as once every two days.
	BEA believes that this issue affects SMP systems more than single CPU systems.
	Affected Linux versions are: SuSE Linux Enterprise Server 9.0 and Red Hat Enterprise Linux 4.0 on Itanium running on 2.6.11 (or previous) kernels. (CR230226 and CR218035).
	SuSE has confirmed that this bug will be fixed in their upcoming SLES 9.0 SP2 release. We are working with Red Hat to get this fix into RHEL 4.0 Update 2.
	You can see the committed patch at:
	http://kernel.org/git/?p=linux/kernel/git/ torvalds/linux-2.6.git;a=commit; h=a2a64769d0d3cc0380b4b6ecdcb781a2f790a69e
	Customers running SMP systems **must** patch their kernel to include the kernel fix or upgrade to a newer kernel. As of this writing (June 17, 2005) no officially updated kernels are available. If you are running on SLES 9.0 Itanium, you should upgrade to SP2 once it is made available.
	The Red Hat internal reference number is 74397. For SuSE the internal reference number is 78084.
N/A	JRockit 1.4.2_05 R24.4.1 on Red Hat Enterprise Linux 4.0 x86_64 is suffering from Red Hat Enterprise Linux 4.0 kernel bugs. You must use the Red Hat Enterprise Linux 4.0 Update 1.
N/A	When using the JRockit 1.4.2 on Linux, you should first make sure that the environment variable 'LD_ASSUME_KERNEL' is *not* defined. If 'LD_ASSUME_KERNEL' is defined, JRockit will use an older and slower threading implementation for no gain.
N/A	Do NOT install i386 versions of glibc on Red Hat or SuSE Linux since JRockit will randomly crash or hang. JRockit only supports the glibc versions built for the i686 architecture on Red Hat IA-32.

Change Request ID	Description
N/A	When running the Java BEA JRockit Management Console over a remote X connection, turn the anti-aliasing option off, since it severely slows down the console.
N/A	When starting BEA JRockit with the `-Xverbose:memory` option, the system message will only display the maximum heap size if that value has been set by using `-Xmx`.
CR175446	When using Linux on Hyper Threading-enabled Intel processors, be aware of the following: • To make sure that JRockit can detect that Hyper Threading is enabled you need to make the devices in `/dev/cpu/X/cpuid` (where X is the cpu number) readable for any process. This is not a security problem, since almost all of this information is available through /proc/cpuinfo anyway, and it is read only. • For the device to be loaded automatically you also need to add the line alias char-major-203 cpuid to your `/etc/modules.conf` file. JRockit will still work on Hyper Threading enabled system even if it doesn't detect that Hyper Threading is enabled, but Hyper Threading specific optimizations will not be performed.
CR129229	Java Plug-in with Netscape 7.1 and Mozilla 1.5 on Linux requires different libgcc files. There are two different compilations of the Java Plug-in library for Netscape 6: one "normal" and one compiled with gcc 3.2. As noted in the Sun Java Plug-in installation guide for Linux at: http://java.sun.com/j2se/1.4.2/manual_install_linux.html#note Newer browser versions, specifically Mozilla 1.5 and Netscape 7.1, require the gcc-3.2-compiled version of libjavaplugin.so in `<JRE>/plugin/i386/ns610-gcc32/libjavaplugin_oji.so`, whereas older versions, specifically Netscape 7.02, require the "normal" one in `<JRE>/plugin/i386/ns610/libjavaplugin_oji.so` The newer ns610-gcc32 version requires the library libgcc_s.1.so, which is not installed by default on Red Hat Linux Advanced Server 2.1. Therefore, if you are going to use one of these browsers on Red Hat Linux Advanced Server 2.1, we recommend that you install `libgcc-3.2-7.i386.rpm` to work around this problem. **Note:** This is a non-standard library with the remote possibility of having an adverse effect on your system.

Change Request ID	Description
CR185163	JRockit might under some circumstances report the wrong line number in the stack trace of a thrown java.lang.ClassCastException. However, the stack trace will report the correct method from which the exception was thrown.
CR135725	JRockit may deadlock when forking. This may occur when compiling JSPs. There is a known issue in Red Hat Advanced Server 2.1, see: https://bugzilla.redhat.com/bugzilla/show_bug.cgi?id=115429 for more information