

# Oracle® JRockit JDK

Upgrade Guide

R27.6

June 2008

ORACLE®

Copyright © 2007, 2008, 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 software or related documentation 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 RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software 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 which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software 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.

# Contents

## Upgrading Oracle JRockit JDK

Supported Configurations .....	1
Upgrading from an Earlier Release .....	1
Java Standard Edition (Java SE) Dependencies .....	2
J2SE 1.4.2 .....	2
J2SE 5.0 .....	2
Java SE 6 .....	3
Upgrading from Oracle JRockit JDK 5.0 R26 .....	3
Upgrading from Oracle JRockit JDK 1.4.2 R26 .....	3
Upgrading from Oracle JRockit JDK 1.4.2 R24 .....	3
Upgrading from Oracle JRockit JDK 8.1 .....	3
Upgrading from Oracle JRockit JDK 7.0 SP6 R23 .....	3
Special Note on Upgrading to JRockit JVM R26 and R27 .....	4
Practical Upgrading .....	4
The JVM Version .....	4
Startup Options .....	5
Testing Your Application in a Staging Environment .....	5
Release Notes .....	5
Changes to Intermediate Releases .....	5
Noticeable Changes .....	6
Updated Command-line Options .....	6
Support for Larger Heaps .....	8

Garbage Collection . . . . .	8
References API . . . . .	8
Client Application Deployment Technologies . . . . .	8
Tools — JRockit Mission Control. . . . .	8

# Upgrading Oracle JRockit JDK

This document contains information that will help you upgrade to the latest version of the Oracle JRockit JDK. It contains information on the following subjects:

- [Supported Configurations](#)
- [Upgrading from an Earlier Release](#)
- [Special Note on Upgrading to JRockit JVM R26 and R27](#)
- [Practical Upgrading](#)
- [Release Notes](#)

## Supported Configurations

Please make sure that the Oracle JRockit JVM is deployed on a supported configuration. If it isn't, you will have to upgrade your configuration. For details on all supported configurations, please refer to Oracle JRockit JDK [Supported Configurations](#) at:

[http://edocs.bea.com/jrockit/jrdocs/suppPlat/supp\\_plat.html](http://edocs.bea.com/jrockit/jrdocs/suppPlat/supp_plat.html)

## Upgrading from an Earlier Release

This section contains information critical to successfully upgrading your version of the JRockit JDK to version R27. It includes information on the following subjects:

- [Java Standard Edition \(Java SE\) Dependencies](#)

- [Upgrading from Oracle JRockit JDK 5.0 R26](#)
- [Upgrading from Oracle JRockit JDK 1.4.2 R26](#)
- [Upgrading from Oracle JRockit JDK 1.4.2 R24](#)
- [Upgrading from Oracle JRockit JDK 8.1](#)
- [Upgrading from Oracle JRockit JDK 7.0 SP6 R23](#)

## Java Standard Edition (Java SE) Dependencies

J2SE dependencies vary, depending upon the J2SE version you are running:

- [J2SE 1.4.2](#)
- [J2SE 5.0](#)
- [Java SE 6](#)

### J2SE 1.4.2

Oracle JRockit JDK 1.4.2 R26 and R27 are certified against J2SE 1.4.2 and it includes the Sun J2SE class libraries for J2SE 1.4.2. Please make sure that your application is compliant with J2SE 1.4.2. For verification, please refer to:

<http://java.sun.com/j2se/1.4.2/docs/api/>

Please refer to [Compatibility Between Releases](#) for general implications at:

<http://edocs.bea.com/jrockit/jrdocs/suppPlat/prodsupp.html#wp999010>

### J2SE 5.0

Oracle JRockit JDK 5.0 R26 and R27 are certified against J2SE 5.0 and it includes the Sun J2SE class libraries for J2SE 5.0. Please make sure that your application is compliant with J2SE 5.0. For verification, please refer to:

<http://java.sun.com/j2se/1.5.0/docs/api/>

Please refer to [Compatibility Between Releases](#) for general implications at:

<http://edocs.bea.com/jrockit/jrdocs/suppPlat/prodsupp.html#wp999010>

## Java SE 6

Oracle JRockit JDK 6 R27 is certified against Java SE 6 and it includes the Sun Java SE class libraries for Java SE 6. Please make sure that your application is compliant with Java SE 6. For verification, please refer to:

<http://java.sun.com/javase/6/docs/api/>

Please refer to [Compatibility Between Releases](#) for general implications at:

<http://edocs.bea.com/jrockit/jrdocs/suppPlat/prodsupp.html#wp999010>

## Upgrading from Oracle JRockit JDK 5.0 R26

No additional procedures are required to upgrade to Oracle JRockit JDK 5.0 R27 from Oracle JRockit JDK 5.0 R26.

## Upgrading from Oracle JRockit JDK 1.4.2 R26

No additional procedures are required to upgrade to Oracle JRockit JDK 1.4.2 R27 from Oracle JRockit JDK 1.4.2 R26.

Note that Oracle JRockit JDK 1.4.2 R27 supports Solaris.

## Upgrading from Oracle JRockit JDK 1.4.2 R24

Oracle JRockit JDK 1.4.2 R24 was certified against J2SE 1.4.2, which is the same J2SE version that Oracle JRockit JDK 1.4.2 R27 has been certified against. Your application should already be compatible with J2SE 1.4.2.

## Upgrading from Oracle JRockit JDK 8.1

Oracle JRockit JDK 8.1 was certified against J2SE 1.4.1. Make sure that your application is compatible with J2SE 1.4.2. Please refer to the section [Java Standard Edition \(Java SE\) Dependencies](#) for further details.

## Upgrading from Oracle JRockit JDK 7.0 SP6 R23

Oracle JRockit JDK 7.0 SP6 R23 was certified against J2SE 1.3.1. Make sure that your application is compatible with J2SE 1.4.2.

Refer to [Java Standard Edition \(Java SE\) Dependencies](#) for further details.

## Special Note on Upgrading to JRockit JVM R26 and R27

JRockit JVM R26 and R27 represent new, major JVM releases when you upgrade to it from any of these releases:

- Oracle JRockit JDK 1.4.2 R24.5.0
- Oracle JRockit JDK 8.1
- Oracle JRockit JDK 7.0

By simply replacing your existing JVM with this version and making no or minimal changes to your application, JRockit JVM R26 and R27 will deliver:

- Improved performance
- Improved startup time
- Much improved `-xdebug` mode
- Access to Oracle JRockit Mission Control, providing stabilized and fully-supported feature-rich monitoring and management tools

## Practical Upgrading

This section includes information that you will find helpful when upgrading to JRockit JVM R27. It includes information on these subjects:

- [The JVM Version](#)
- [Startup Options](#)
- [Testing Your Application in a Staging Environment](#)

### The JVM Version

To verify the installation, check the version string by issuing the command `java -version`. The JRockit JDK version information, for example, Oracle JRockit JDK 1.4.2\_15 R27.4.0 is comprised of two elements

- The Java 2 Standard Edition (J2SE) major, minor, micro, and update version; for example, 1.4.2\_15
- The specific JRockit JVM version; for example, R27.4.0

## Startup Options

As a rule of thumb, consider as few startup options as possible. Please make sure that all startup options are well understood and required.

Some `-xx` options might have changed. For more information of how options can be changed, please refer to [Compatibility Between Releases](#) at:

<http://edocs.bea.com/jrockit/jrdocs/suppPlat/prodsupp.html#wp999010>

## Testing Your Application in a Staging Environment

Please make sure to test your application in a controlled staging environment before upgrading your production environment. The staging environment should be as similar to your production environment as possible. This includes hardware, software, usage patterns and load.

Applications are sometimes sensitive to timing. A JRockit JVM upgrade can sometimes expose errors in your application; for example, dependencies on unspecified behavior or APIs. Timing differences introduced when upgrading the JVM may for instance expose erroneous synchronization.

## Release Notes

This section contains important information regarding release information that might have some impact on upgrading from one version of the JRockit JVM to another. It contains the following subjects:

- [Changes to Intermediate Releases](#)
- [Noticeable Changes](#)

## Changes to Intermediate Releases

Be careful when reading the recent changes in the release notes; be sure to read about the changes happening in intermediate version upgrades up to R27. For a list of the changes in intermediate versions, please refer to the release notes for each version:

- [Oracle JRockit JDK 7.0 Release Notes](#)
- [Oracle JRockit JDK 1.4.2 Release Notes](#)
- [Oracle JRockit JDK 5.0 Release Notes](#)

- [Oracle JRockit JDK R26 Release Notes](#)
- [Oracle JRockit JDK R27 Release Notes](#)

## Noticeable Changes

Following is a list of some of the changes that might indicate that your application could behave differently after upgrading to R27. These changes include:

- [Updated Command-line Options](#)
- [Support for Larger Heaps](#)
- [Garbage Collection](#)
- [References API](#)
- [Client Application Deployment Technologies](#)
- [Tools — JRockit Mission Control](#)

Please also visit the R27 [Developers' FAQ](#) at:

[http://edocs.bea.com/jrockit/webdocs/dev\\_faq.html](http://edocs.bea.com/jrockit/webdocs/dev_faq.html)

## Updated Command-line Options

The rules for how command-line parameters are parsed have been updated to avoid user confusion. Incompatible command-line combinations now cause JRockit JVM to print out an error message and terminate. Please refer to the specific option in the Oracle JRockit JVM *Command-Line Reference* (by clicking on the option name in [Table 1](#)) for a description of the new behavior.

**Table 1 Updated Command-line Options**

<b>-X options</b>	<b>-XX options</b>
<a href="#">-Xgc</a>	<a href="#">-XXallocClearChunks</a>
<a href="#">-XgcPrio</a>	<a href="#">-XXallocClearChunkSize</a>
<a href="#">-Xms</a>	<a href="#">-XXallocPrefetch</a>
<a href="#">-Xmx</a>	<a href="#">-XXallocRedoPrefetch</a>

**Table 1 Updated Command-line Options**

<b>-X options</b>	<b>-XX options</b>
<code>-Xns</code>	<code>-XXcompactSetLimit</code>
<code>-XpauseTarget</code>	<code>-XXcompactSetLimitPerObject</code>
	<code>-XXcompactRatio</code>
	<code>-XXcompressedRefs</code>
	<code>-XXdisableFatSpin</code>
	<code>-XXdisableGcHeuristics</code>
	<code>-XXexternalCompactRatio</code>
	<code>-XXfullCompaction</code>
	<code>-XXfullSystemGc</code>
	<code>-XXinternalCompactRatio</code>
	<code>-XXlargeObjectLimit</code>
	<code>-XXminBlockSize</code>
	<code>-XXnoCompaction</code>
	<code>-XXnoSystemGc</code>
	<code>-XXpointerMatrixLinearSeekDistance</code>
	<code>-XXsetGC</code>
	<code>-XXstaticCompaction</code>
	<code>-XXthroughputCompaction</code>
	<code>-XXtlaSize</code>
	<code>-XXusePointerMatrix</code>
	<code>-XX:(+ -)useNewHashFunction</code>
	<code>-XX:(+ -)useThreadPriorities</code>

## Support for Larger Heaps

It is now possible to make use of a larger part of the process memory for the java heap as the demand for continuous heap has been removed.

## Garbage Collection

The new default garbage collection strategy is `-Xgcprio:throughput`. A new feature `-XpauseTarget` is implemented to specify desired maximum pause time. The Generational Copy collector, started with `-Xgc:gencopy` has been removed. For more information on garbage collection in the Oracle JRockit JVM, please refer to [Memory Management Basics](#) at:

[http://edocs.bea.com/jrockit/geninfo/diagnos/garbage\\_collect.html](http://edocs.bea.com/jrockit/geninfo/diagnos/garbage_collect.html)

## References API

New implementation of how softly reachable references are collected. The `-Xverbose:references` output has been enhanced. For more information, please refer to the `-Xverbose` section in the Oracle JRockit JVM [Command-Line Reference](#) at

<http://edocs.bea.com/jrockit/jrdocs/refman/optionX.html#wp999543>

## Client Application Deployment Technologies

The features of Java Web Start and Java Plug-in have been removed.

## Tools — JRockit Mission Control

The monitoring and management tools now require a license. A free development license is available to test the tools in a development environment and limited to work only during the first hour of any application run; that is, restarting the application will allow another hour of tools testing. For more information on JRockit Mission Control licensing, please refer to [JRockit Mission Control documentation](#) at:

<http://edocs.bea.com/jrockit/tools/index.html>

Oracle JRockit Mission Control includes the Management Console, JRockit Runtime Analyzer and the Memory Leak Detector. The Management Console can be used without a license but is then not eligible for support.

For more information on JRockit Mission Control, please refer to the [JRockit Mission Control page on dev2dev](#), at:

<http://dev2dev.bea.com/jrockit/tools.html>

and the JRockit Mission Control [documentation set](#) at:

<http://edocs.bea.com/jrockit/tools/index.html>

