



BEA Oracle JRockit Mission Control™®

Monitoring MBeans from the JRockit Management Console

JRockit Mission Control 3.0.2
Document Revised: June, 2008

Contents

Introduction to Java MBeans

Architectural Overview	1-1
JMX Specification	1-2
Platform MBeans	1-2
Oracle JRockit JDK MBeans	1-3
MBean Attributes	1-4

Using the Management Console MBean Browser

Getting Familiar with the MBean Browser Tab	2-1
MBean Icons Explained	2-3
Viewing MBean Attributes	2-3
Editing MBean Attributes	2-4
Copying Attribute Data to Other Applications	2-6
Invoking Operations	2-7
Viewing Notifications	2-9

Introduction to Java MBeans

An MBean is a managed object that follows the design patterns conforming to the JMX specification. An MBean can represent a device, an application, or any resource that needs to be managed. The management interface of an MBean comprises a set of readable and/or writable attributes, operations, and notifications.

This topic serves as a brief introduction to MBeans and how they are accessed by Oracle JRockit Mission Control. It includes information on the following subjects:

- [Architectural Overview](#)
- [JMX Specification](#)
- [Platform MBeans](#)
- [Oracle JRockit JDK MBeans](#)
- [MBean Attributes](#)

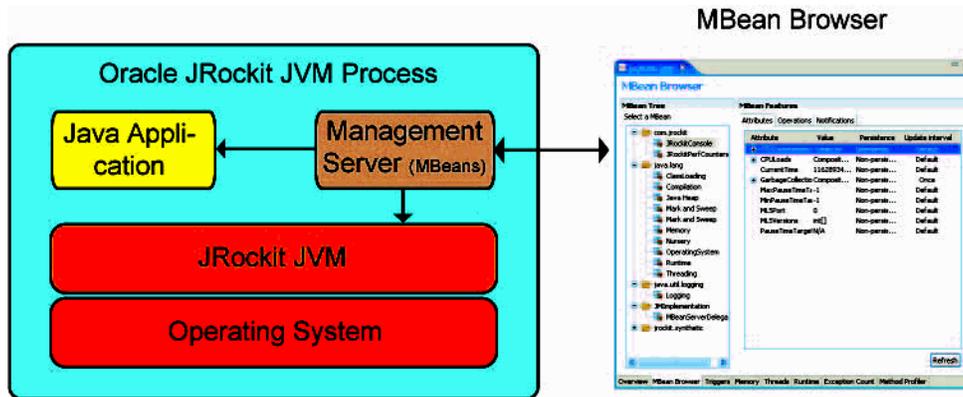
Architectural Overview

[Figure 1-1](#) shows the architecture of the monitoring and management support. The Oracle JRockit JVM is highly equipped for monitoring and management. The platform equipment provides information on performance, resource consumption, and the JVM and logging settings of applications running on the Java platform.

JMX provides a standard way to connect to the Java runtime environment and applications; the JMX Remote API allows that environment to be accessed remotely. The environment is

accessible through the JMX managed bean (MBean) interfaces, which are registered in the management server. Applications can also create their own MBeans and register them in the management server, which can serve as a single point for remote access. A JMX-compliant client, such as the Management Console, can connect to the platform management server and manage the application (as well as the Java platform) using JMX technology.

Figure 1-1 Architectural Overview of the JRockit Monitoring and Management Support



JMX Specification

For a complete reference on the JMX standard, please see:

<http://java.sun.com/j2se/1.5.0/docs/api/javax/management/package-summary.html>

Platform MBeans

The Java platform provides a set of platform MBeans for monitoring and managing the Java virtual machine and the logging facility. Table 1-1 describes the MBeans included in this release:

Table 1-1 MBeans Included in JRockit Mission Control

Platform MBean	Description
<code>java.lang.management.ClassLoadingMXBean</code>	Class loading system of the Java virtual machine.
<code>java.lang.management.CompilationMXBean</code>	Compilation system of the Java virtual machine.
<code>java.lang.management.MemoryMXBean</code>	Memory system of the Java virtual machine.

Table 1-1 MBeans Included in JRockit Mission Control

Platform MBean	Description
<code>java.lang.management.MemoryManagerMXBean</code>	Memory manager in the Java virtual machine.
<code>java.lang.management.MemoryPoolMXBean</code>	Memory pool in the Java virtual machine.
<code>java.lang.management.GarbageCollectorMXBean</code>	Garbage collector in the Java virtual machine.
<code>java.lang.management.ThreadMXBean</code>	Threading system of the Java virtual machine.
<code>java.lang.management.RuntimeMXBean</code>	Runtime system of the Java virtual machine.
<code>java.lang.management.OperatingSystemMXBean</code>	Operating system on which the Java virtual machine is running.
<code>java.util.logging.LoggingMXBean</code>	Logging facility.

Oracle JRockit JDK MBeans

The Oracle JRockit JDK has its own set of MBeans. For the JRockit JVM R26, these MBeans inherit the functionality of the platform MBeans described in [Table 1-1](#). The inherited MBeans are using the same domain and names as the platform MBeans, but expose additional attributes, operations and notifications.

For Oracle JRockit JVM R27.1, the JRockit JVM-specific MBeans have been relocated to its own domain: `bea.jrockit.management`.

Note: These MBeans are proprietary and subject to change at any time. While you can access these MBeans directly using other JMX based tools, such direct access is not supported by Oracle.

An MBean is a managed object that follows the design patterns conforming to the JMX specification. An MBean can represent a device, an application, or any resource that needs to be managed.

Each platform MBean has a set of attributes and operations such as memory usage, thread CPU usage, garbage collection statistics, and so on. Some might also emit notifications.

The Management Console can connect to the Management Server, which in turn allows you to monitor your application and view all the JMX objects. You need to be connected to the management server to view the objects.

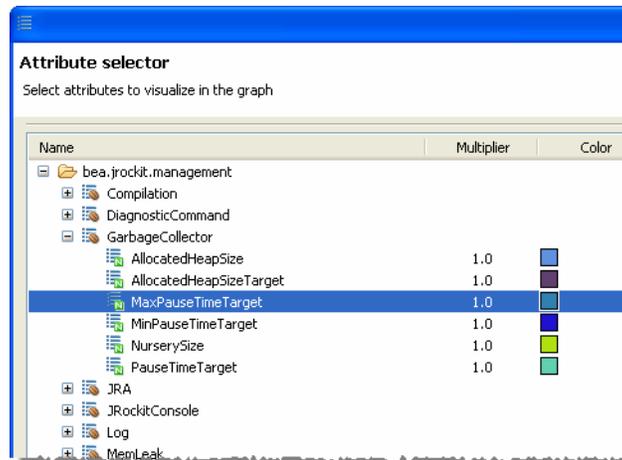
MBean Attributes

Each MBean has its own set of attribute subscriptions that can be used to monitor the different aspects of your application and connection (see [To edit an attribute value](#) for how to change attribute values).

By subscribing to the attributes, you have a powerful way to monitor your application. You can subscribe to the attributes from many places in the Management Console; for example, you can add subscription attributes to the graphs on the **Overview** tab.

[Figure 1-2](#) shows the **Attribute selector** that is shown when you want to add an attribute to a graph on the **Overview** tab of the Management Console.

Figure 1-2 Attribute Subscription Selector



Using the Management Console MBean Browser

Oracle JRockit Mission Control uses MBeans to provide a unified and consistent interface for monitoring and managing application performance. To monitor the performance of these MBeans, JRockit Mission Control contains an MBean Browser,

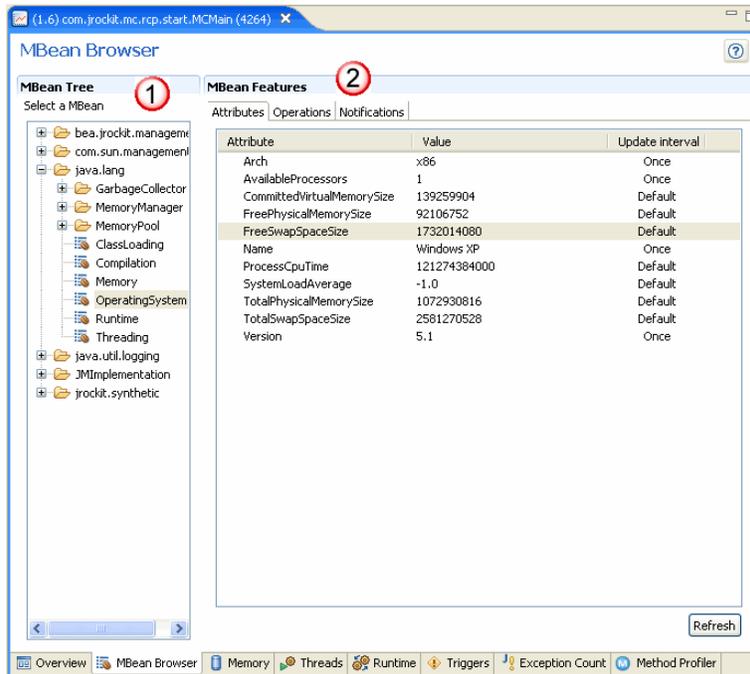
This section includes the following topics:

- [Getting Familiar with the MBean Browser Tab](#)
- [MBean Icons Explained](#)
- [Viewing MBean Attributes](#)
- [Editing MBean Attributes](#)
- [Copying Attribute Data to Other Applications](#)
- [Invoking Operations](#)
- [Viewing Notifications](#)

Getting Familiar with the MBean Browser Tab

This section introduces the **MBean Browser** tab ([Figure 2-1](#)).

Figure 2-1 MBean Browser



The **MBean Browser** tab is divided into the following sections:

1. **MBean Tree**—is a list of all registered MBeans (listed by Java package).
2. **MBean Features**—a multi-tabbed interface that contains the following tabs:
 - **Attributes**, which lists the attributes for each MBean.
 - **Operations**, which lists the operations you can invoke from the selected MBean.
 - **Notifications**, which lists any notifications prompted by the MBean during runtime.

MBean Icons Explained

[Table 2](#) describes the available icons for MBeans.

Table 2 Description of MBean Icons

Icon	Description
	The MBean itself.
	A composite attribute. Open the folder and you display all attribute subscriptions for the composite attribute.
	Indicates that the attribute subscription is numerical.
	Indicates that the attribute subscription is a synthetic numerical. A synthetic attribute subscription does not exist on the server. It is either calculated on the client or derived from other attribute subscriptions.

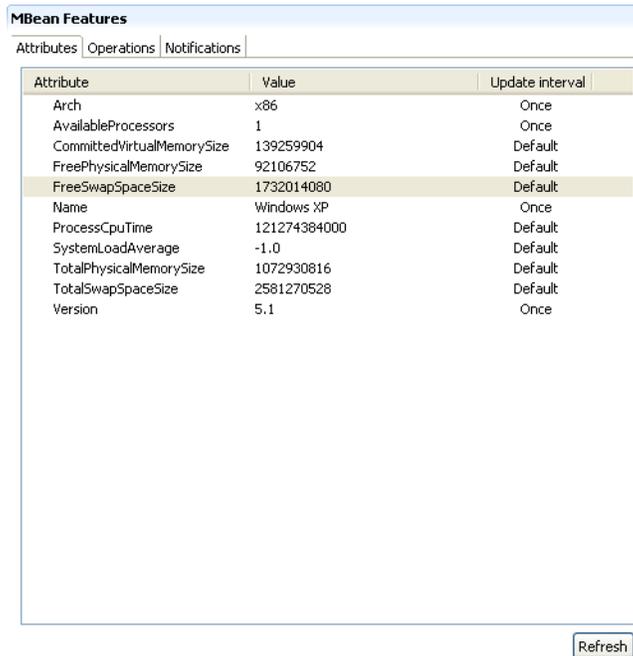
Viewing MBean Attributes

MBean attributes represent the current state of data and information within a managed object. Attributes can be read-only for the purposes of simply revealing state or read-write for providing the ability to update it (writable fields are **bold** in the interface).

To view MBean attributes

- Click the **Attributes** tab to open the tab ([Figure 2-3](#)).

Figure 2-3 Attributes Tab



The screenshot shows a window titled "MBean Features" with three tabs: "Attributes", "Operations", and "Notifications". The "Attributes" tab is active, displaying a table with three columns: "Attribute", "Value", and "Update interval". The table lists various system attributes such as "Arch", "AvailableProcessors", "CommittedVirtualMemorySize", "FreePhysicalMemorySize", "FreeSwapSpaceSize", "Name", "ProcessCpuTime", "SystemLoadAverage", "TotalPhysicalMemorySize", "TotalSwapSpaceSize", and "Version". A "Refresh" button is located at the bottom right of the table area.

Attribute	Value	Update interval
Arch	x86	Once
AvailableProcessors	1	Once
CommittedVirtualMemorySize	139259904	Default
FreePhysicalMemorySize	92106752	Default
FreeSwapSpaceSize	1732014080	Default
Name	Windows XP	Once
ProcessCpuTime	121274384000	Default
SystemLoadAverage	-1.0	Default
TotalPhysicalMemorySize	1072930816	Default
TotalSwapSpaceSize	2581270528	Default
Version	5.1	Once

The attributes are arrayed on a table that displays the following information:

- **Attribute**—the name of the attribute. This value cannot be updated.
- **Value**—the attribute’s value. This value is editable, as described in [To edit an attribute value](#).
- **Update interval**—how often during runtime the attribute is updated. This value is editable, as described in [To edit the update interval](#).

Editing MBean Attributes

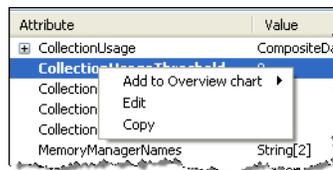
For some MBean attributes, you can change the properties of: Value and Update interval. This section describes how:

- [To edit an attribute value](#)
- [To edit the update interval](#)

To edit an attribute value

1. Right-click the **Value** column for that attribute you want to edit.
A context menu appears (Figure 2-4).

Figure 2-4 Value Column with Edit Button



2. Select **Edit**.
3. Depending on the type of attribute you are trying to edit, you will get a matching Edit dialog.
For a True/False value, the Edit dialog looks like Figure 2-5.

Figure 2-5 True/False Value



4. Select or type in a different value in the dialog that appears.
5. Click **OK**.

To edit the update interval

1. Click on the **Attribute** you wish to change.
2. Click in the **Update interval** column for that attribute.

The field opens for updating and a drop-down control appears (Figure 2-6)

Figure 2-6 Update Interval Open for Edit

CurrentThreadCpuTimeSupported	true	Non-persistent	Default
CurrentThreadUserTime	360518400	Non-persistent	Default
DaemonThreadCount	14	Non-persistent	2000
NumberOfMonitorDeadlocksThreshold	0	Non-persistent	Default

3. Click the drop-down control to open the Update interval menu (Figure 2-7).

Figure 2-7 Update Interval Menu



4. Select the update interval you want to use:
 - **Once**, to have the attribute once per application run.
 - **Default**, to use the default value.
 - **Custom...**, to select a custom update interval.

If you select either **Once** or **Default**, the menu closes and the new update interval appears in the **Update interval** field. If you select **Custom...**, the Enter update interval dialog box appears (Figure 2-8).

Figure 2-8 Enter update interval Dialog Box



5. Enter the new update interval (in milliseconds) and click **OK**.

The new update interval will appear in the **Update interval** field.

Copying Attribute Data to Other Applications

You can copy the information about an attribute displayed in the Attribute Features panel of the MBean Browser's Attributes tab to other applications, such as Notepad, Microsoft Excel, or even an email program. This is a useful feature if you want to preserve certain data for later reference or for troubleshooting when your application is no longer running or otherwise available.

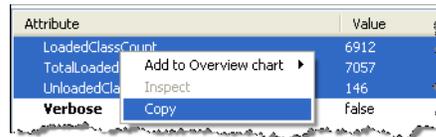
To copy attribute data to another application

1. Open the Attributes tab and locate the attribute whose data you want to copy.

The features of that attribute appear in the Attribute Features panel.

2. Select the features you want to copy. To select multiple features, hold down the Shift key (for a range of contiguous features) or the Ctrl key (for non-contiguous features) and select the features. Right click your selection to open a context menu and select Copy ([Figure 2-9](#)),

Figure 2-9 Selecting the Copy command



3. You can now paste the information into another application; for example, [Figure 2-10](#) shows what it might look like if copied attribute features were pasted into Microsoft Excel.

Figure 2-10 Attribute features copied into Excel

	A	B	C
1	MemoryPoolNames	String[2]	Default
2	LastGcInfo	null	Default
3	Name	Static Young Collector	Default

Note that columnar integrity of the Attribute Features table is preserved.

Invoking Operations

Operations are a mechanism that cause the execution of code on the JMX server within the Managed Object. These operations are basically a remote method invocation and can take parameters and return a result. JRockit Mission Control supports the execution of these operations, along with passing parameters and displaying the results.

The Operations tab lists the operations associated with each MBean and allows you to invoke any of those operations. For each operation, the tab shows the operation name, its parameters, and its return value.

This topic shows you how:

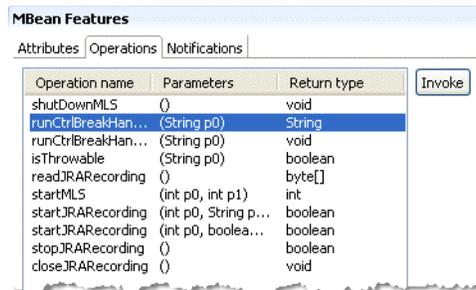
- [To open the operations tab](#)
- [To invoke an operation](#)

To open the operations tab

- Click the **Operations** tab on the **MBean Browser** tab.

The **Operations** tab appears (Figure 2-11).

Figure 2-11 Operations Tab



To invoke an operation

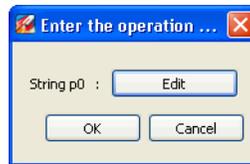
1. Highlight the operation name you want to invoke.

Note: To be able to invoke a value, the Operation name needs to have a Parameter that is editable, i.e. there is a value within the parentheses in the **Parameters** column.

2. Click **Invoke**.

Depending on the value of the parameter, different control windows open. In this example, a parameter with a string value is shown (Figure 2-12).

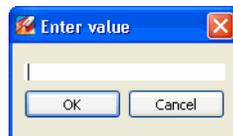
Figure 2-12 Operations with String Parameter Value



3. Click **Edit**.

The **Enter value** dialog opens (Figure 2-13).

Figure 2-13 Enter Value Dialog



4. Type in a value.

5. Click **OK**.
6. Click **OK**.

Viewing Notifications

Notifications provide information about events on the server. Notifications are exposed through an MBean and must be “listened for” before you will see the event. You can view Notifications on the **MBean Browser/Notifications** tab.

This topic shows you how:

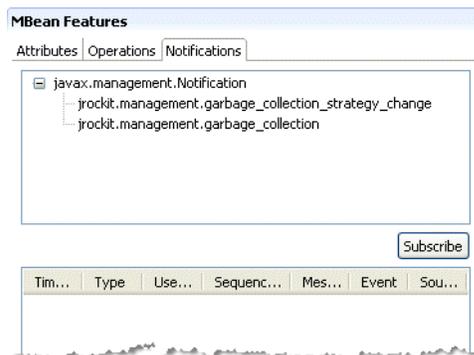
- [To open the Notifications tab](#)
- [To subscribe to a notification](#)

To open the Notifications tab

- Click the **Notifications** tab on the **MBean Browser** tab.

The **Notifications** tab appears ([Figure 2-14](#)).

Figure 2-14 Notifications Tab



To subscribe to a notification

1. Select an MBean
2. Click the **Notifications** tab on the **MBean Browser** tab.
3. Click **Subscribe**.

Using the Management Console MBean Browser