

Java Platform, Standard Edition

Java Mission Control User's Guide

Release 5.4

E61550-01

August 2014

Provides an overview of Java Mission Control. It includes information about the features, architecture, and accessibility of the product.

Copyright © 2001, 2014, 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 is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

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

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

Contents

1 About Java Mission Control

2 Using the Java Mission Control Client

2.1	Starting the Java Mission Control Client	2-1
2.2	Java Mission Control Client GUI	2-2
2.2.1	Tabs and Tab Groups	2-3
2.2.2	Panels	2-3

3 Accessibility in Java Mission Control

3.1	Screen Readers	3-1
3.2	Java Mission Control Accessibility Mode	3-1
3.3	Showing Text Labels on Buttons	3-2
3.4	Resizing Online Help Text	3-2
3.5	Known Issues and Workarounds	3-2

4 Logging Properties

5 Experimental Plug-ins

6 Frequently Asked Questions

Preface

Note: Java Mission Control requires a commercial license for use in production. To learn more about commercial features and how to enable them please visit

<http://www.oracle.com/technetwork/java/javaseproducts/>.

This document provides an overview of Java Mission Control. It includes information about the features, architecture, and accessibility of the product.

Audience

This document is intended for Java developers and support engineers who need an introduction about the functionality and architecture of Java Mission Control. It assumes that the reader has basic knowledge of the Java programming language.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at
<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit
<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit
<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documents

For information about using Java Mission Control, refer to the online help included with the product. You can view it by pressing F1 when working with the Java Mission Control client, or by clicking one of the question mark buttons located on most windows in the graphical user interface (GUI).

For information about Java Flight Recorder (JFR), see the *Java Flight Recorder Runtime Guide* at <http://docs.oracle.com/javacomponents/jmc.htm>.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>monospace</code>	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Abbreviations

The following abbreviations are used in this document:

Abbreviation	Meaning
JDK	Java Development Kit
JDP	Java Discovery Protocol
JFR	Java Flight Recorder
JMC	Java Mission Control
JMX	Java Management Extensions
JVM	Java Virtual Machine
MBean	Managed Bean (Java)
RCP	Rich Client Platform (Eclipse)
RMI	Remote Method Invocation (Java)

Support and Feedback

You can contact Oracle Support for technical assistance if you are entitled to the Oracle Support services for Java Mission Control. The following types of licenses include support for Java Mission Control:

- Oracle Java SE Advanced
- Oracle Java SE Suite
- WebLogic Server Enterprise Edition
- WebLogic Suite

If you have any suggestions about how to improve Java Mission Control or information on how it is most commonly used in your development environment, post it on the Java Mission Control forum at:

https://community.oracle.com/community/java/java_hotspot_virtual_machine/java_mission_control

All feedback is considered by the Java Mission Control development team to find ways for improving the product. Oracle's goal is to simplify the tasks for making your Java applications as fast and efficient as possible.

About Java Mission Control

Note: Java Mission Control requires a commercial license for use in production. To learn more about commercial features and how to enable them please visit

<http://www.oracle.com/technetwork/java/javaseproducts/>.

Java Mission Control enables you to monitor and manage Java applications without introducing the performance overhead normally associated with these types of tools. It uses data collected for normal adaptive dynamic optimization of the Java Virtual Machine (JVM). Besides minimizing the performance overhead, this approach eliminates the problem of the *observer effect*, which occurs when monitoring tools alter the execution characteristics of the system.

Java Mission Control consists of the client application (*JMC client*) and a number of plug-ins that run on it:

- *JVM Browser* shows the running Java applications and their JVMs. Each JVM instance is called a *JVM Connection*.
- *JMX Console* connects to a running JVM, collects and displays its characteristics in real time, and enables you to change some of the runtime properties through Managed Beans (MBeans). You can also create rules that trigger on certain events (for example, to send an e-mail if the CPU usage by the application reaches 90 percent).
- *Java Flight Recorder* (JFR) collects and saves detailed performance characteristics for historic analysis and profiling. It can be used as a standalone performance monitoring and profiling tool, but when used as a plug-in for the JMC client, it presents diagnostic information in logically grouped tables, charts, and dials. It enables you to select the range of time and level of detail necessary to focus on the problem.

Java Mission Control plug-ins connect to a JVM using the Java Management Extensions (JMX) agent. For more information about JMX, see the JMX Technology Home Page at

<http://www.oracle.com/technetwork/java/javase/tech/javamanagement-140525.html>

Using the Java Mission Control Client

This chapter describes how to start the JMC client and use its features.

Note: Java Mission Control requires a commercial license for use in production. To learn more about commercial features and how to enable them please visit <http://www.oracle.com/technetwork/java/javaseproducts/>.

The JMC client is the main application from which you connect to JVMs and start the tools included in Java Mission Control. You can run the JMC client either as a standalone application or as an Eclipse plug-in. Ensure that you run the JMC client in a secure environment, because it does not include or run with a security manager.

2.1 Starting the Java Mission Control Client

The JMC client executable file is located in the `bin` directory of the Java SE Development Kit (JDK) installation path (`JAVA_HOME`). If the `JAVA_HOME/bin` directory is in the `PATH` environment variable, you can start the JMC client by entering `jmc` at the command-line prompt (shell). Otherwise, you have to specify the full path to the JMC executable:

- `JAVA_HOME\bin\jmc.exe` (Windows)
- `JAVA_HOME/bin/jmc` (Linux, OS X)

Note: On Windows, a shortcut to the JMC client executable is added to the **Java Development Kit** folder under **All Programs** in the **Start** menu.

Note: On OS X, if you cannot find the JMC executable or locate JMC in Spotlight, you can start JMC from the command line using `/Library/Java/JavaVirtualMachines/jdk<version>.jdk/Contents/Home/bin/jmc`. Alternatively, use Finder to browse to `/Library/Java/JavaVirtualMachines/jdk<version>.jdk/Contents/Home/lib/missioncontrol` and double-click the Java Mission Control app.

Note: On OS X, when you launch the JMC executable, the Terminal window opens. When you close the JMC client, the Terminal is left behind with the text [Process completed]. To avoid this, open the Terminal **Preferences**, then open the **Settings** tab, select the scheme you use, then select the **Shell** tab, and change the **When the shell exits** option to **Close if the shell exited cleanly**.

Passing JVM Options To the JMC Launcher

JMC is a Java application, and the JMC client executable is a launcher for this application. JMC startup is controlled by options specified in the `jmc.ini` file, which is located in the `JAVA_HOME/bin` directory. Arguments to the `-vmargs` option in the `jmc.ini` file are options that are passed to the JVM running the JMC application. You can specify these options to control the way this JVM runs. If you do not want to modify the `jmc.ini` file, you can specify JVM options on the command line as arguments to the `-vmargs` option of the `jmc` command.

Note: If other options are specified for the `jmc` command, the `-vmargs` option must be specified last.

To start the JMC client with your own set of JVM options (overriding those specified in the `jmc.ini` file), run the following command (separate multiple arguments with spaces):

```
jmc -vmarg arguments
```

To start the JMC client with additional JVM options (appending them to those specified in the `jmc.ini` file), run the following command (separate multiple arguments with spaces):

```
jmc --launcher.appendVmargs -vmarg arguments
```

Using a Workspace Directory

If you want to copy your settings for the JMC client to another computer or another user, or use different predefined settings for different applications, add the `-data` command-line option and define a *workspace directory* when you start the JMC client:

```
jmc -data workspace-directory
```

2.2 Java Mission Control Client GUI

When you first launch the JMC client, the left part of the window contains two views by default:

- *JVM Browser* is the plug-in that shows all available JVM connections.
- *Event Types* contains a list of event types for a selected flight recording. For information about opening a flight recording, see the Java Flight Recorder plug-in section of the online help.

You can open the **Window** menu, select **Show View** and then **Other** to open a window that lists all available views. Select the necessary views and click **OK** to open them. The following views are available in addition to the default ones:

- *Properties* lists the properties of items that you select in tables, including hidden properties that are not displayed in the tables.
- *Progress View* displays the progress of running operations, for example, a flight recording.
- *Help* displays the Java Mission Control online help.

The other part of the JMC client window is reserved for the **Editor Area**, which is used to work with the plug-ins for available JVM connections. To launch a plug-in, right-click a JVM connection and select the plug-in (**Start JMX Console**, **Open Historical Data**, **Start Flight Recording**, etc.).

2.2.1 Tabs and Tab Groups

Views and editors are displayed as tabs united into tab groups. You can drag tabs and tab groups around the JMC client window to create a layout that works best for you. View tabs and tab groups can even be detached from the main window.

You can maximize a tab group to the size of the JMC client window or minimize it to make more room for other tab groups:

- To maximize a tab group, click the **Maximize** button in the top-right corner. The tab group expands almost to the size of the JMC client window and the **Maximize** button changes to the **Restore** button. The other tab groups are automatically minimized.
- To minimize a tab group, click the **Minimize** button in the top-right corner. The tab group shrinks and the **Minimize** button changes to the **Restore** button. The other tab groups fill the rest of the JMC client window proportionally.
- To restore the tab to its original state, click the **Restore** button.

To separate logically related information, editor tabs are divided into subtabs (or subtab groups), allowing you to drill down into specific data. Some of the subtabs have buttons to control the data displayed in them to the right of the subtab name.

2.2.2 Panels

Editor tabs are composed of panels with logically related information in the form of graphs, tables, lists, and so on. You can modify some of the parameters using various controls in the panels, such as buttons, text fields, drop-down lists, and so on.

Some panels can be collapsed to make more room for other panels in the tab (using the black triangle show and hide button next to the panel name). To the right of the panel name, there are buttons to add and remove components, refresh data, switch modes, change settings, get online help about the panel, and so on.

Most panels are unique, but some are similar for different tabs. Information for using each panel is provided in the description for the panel. To see this information, click the question mark button to the right of the panel name.

Accessibility in Java Mission Control

Note: Java Mission Control requires a commercial license for use in production. To learn more about commercial features and how to enable them please visit <http://www.oracle.com/technetwork/java/javaseproducts/>.

Oracle is dedicated to providing high quality information technology that is accessible to people with disabilities. To this end, Oracle has undertaken a substantial project to ensure the accessibility of Java Mission Control. Oracle is implementing these enhancements and will continue to address all accessibility issues that arise.

3.1 Screen Readers

Oracle supports a number of screen readers that translate screen-based information into spoken words to assist vision-impaired users.

Configuration options are currently available for the JAWS screen reader produced by Freedom Scientific, Inc. For information on configuring this product, please refer to the Freedom Scientific screen reader website at

<http://www.freedomscientific.com/documentation/screen-readers.asp>

Note: If you are using JAWS, be aware that switching tabs does not work as expected. For a workaround, see "[Switching Between Tabs](#)".

3.2 Java Mission Control Accessibility Mode

Java Mission Control displays performance data dials and graphs that provide a valuable graphical view of the data that can reveal trends and help identify minimum and maximum values for performance metrics. However, dials and graphs do not convey information in a manner that can be read by a screen reader. To remedy this problem, Java Mission Control provides a way to view dial and graph data in tabular format, which is more accessible for users with visual impairment. To toggle the accessibility mode, click the **Accessibility mode** button in the top-right corner of the relevant panel with dials or graphs.

To enable accessibility mode for all panels in Java Mission Control:

1. Open the **Window** menu and select **Preferences**.
2. Select **Java Mission Control** in the left pane.

3. Under **Accessibility Options**, select **Use accessibility mode**.
4. Click **Apply**.

3.3 Showing Text Labels on Buttons

Java Mission Control contains a lot of buttons with graphics and no text labels. To show the text labels that can be read by screen readers:

1. Open the **Window** menu and select **Preferences**.
2. Select **Java Mission Control** in the left pane.
3. Under **Accessibility Options**, select **Show text labels on buttons**.
4. Click **Apply**.

Restart any running plug-ins to see the text labels on buttons.

3.4 Resizing Online Help Text

Vision-impaired users might find it difficult to read the online help documents in the standalone (RCP) version of Java Mission Control unless the text size is increased. Due to a limitation in the Eclipse help viewer, currently you cannot resize its text. To change the font size, view the help in an external browser.

To change the default viewer for Java Mission Control Help:

1. Open the **Window** menu and select **Preferences**.
2. Select **Help** in the left pane.
3. Under **Open Modes**, select to open help contents in an external browser.
4. Click **Apply**.

Java Mission Control will use the default web browser specified in the operating system, and you cannot specify a different web browser from within Java Mission Control. How you specify a browser in the operating system depends on the operating system.

3.5 Known Issues and Workarounds

This section contains additional instructions for enhancing your experience with Java Mission Control's accessibility features.

Navigating in a Tree Table with Only One Row

When navigating in a tree table component containing only one row, press the Space key or the Shift key and use the Up and Down keys to get to the row.

Switching Between Tabs

When reaching a tab component, JAWS erroneously tells the user “to switch pages, press Ctrl+Tab”. The correct way to switch between tabs is to use the left or right arrow keys.

Reading Table Data with a Screen Reader

To read table data more efficiently with screen reading software, copy and paste the table data into a text editor and read it from there. To copy the data, select the rows

that you are interested in, right-click and select **Copy**. Then paste the data into a text editor.

Logging Properties

This chapter describes how to change the way Java Mission Control logs errors and warnings.

Note: Java Mission Control requires a commercial license for use in production. To learn more about commercial features and how to enable them please visit <http://www.oracle.com/technetwork/java/javaseproduct/s/>.

By default, the JMC client uses the `logging.properties` file located in the `com.jrockit.mc.ui` package (Java Archive or JAR file) to control the way logging is performed. For example, if you have a commercial license of the JDK installed in `JAVA_HOME`, the location of this JAR file should be `JAVA_HOME/lib/missioncontrol/plugins`.

You should not modify the original file. Instead, copy it to another location (for example, to the root of the JMC installation directory) and modify the copy. To point the JMC client to the copy of the `logging.properties` file:

1. Open the **Window** menu and select **Preferences**.
2. Select **Java Mission Control** from the left pane.
3. In the **Logging settings file** field, specify the path and name of the logging properties file that you want to use. You can click **Browse** to select the file in the file system explorer window.

Experimental Plug-ins

Note: Java Mission Control requires a commercial license for use in production. To learn more about commercial features and how to enable them please visit

<http://www.oracle.com/technetwork/java/javaseproducts/>.

The experimental plug-ins for Java Mission Control are a set of plug-ins designed to either extend Java Mission Control in various ways or assist in developing such extensions. An experimental plug-in can be one of the following:

- A technology preview that may later be incorporated into the main distribution, depending on the feedback.
- A plug-in extending Java Mission Control with functionality that is not directly related to Java SE, for example, the WebLogic Server plug-in for Java Flight Recorder.

Note: These plug-ins are currently not supported and are provided only for evaluation purposes. The experimental plug-ins are not included in the Java Mission Control product distribution.

You can download and install the experimental plug-ins from the Experimental Update Site using the Java Mission Control client. To install a plug-in:

1. Open the **Help** menu and select **Install New Software**.
2. Select the plug-ins that you want to install and follow the installation wizard.

Note: To use the installed plug-in, restart Java Mission Control.

Frequently Asked Questions

Note: Java Mission Control requires a commercial license for use in production. To learn more about commercial features and how to enable them please visit <http://www.oracle.com/technetwork/java/javaseproducts/>.

The following issues sometimes arise:

Issue 1 I cannot connect to the JVM. What could be the problem?

Consider the following:

- Are you using the correct protocol?

Ensure that you are using the same version of the JVM that you want to monitor as is being used by the JVM running the JMC client.

The format of the service URL is:

`service:jmx:rmi:///jndi/rmi://<hostname>/jmxrmi`

- Are the correct ports opened?

Note that running JMX over RMI requires two ports and that one of the ports will not be known beforehand.

- Is the communication caught in a firewall?

For more information, see "Java Mission Control Communications" in the *Introduction to Java Mission Control* on the Oracle Technology Network.

Issue 2 When attempting to connect to a JVM, I get a stack trace indicating that the JVM attempts to communicate with a strange IP or host name. What does this mean?

Sometimes RMI can have a problem determining which address to use. This can happen because of any of the following:

- There are access restrictions in the Security manager.
- The machine is multihomed and RMI is selecting the wrong interface.
- There is a mis-configured hosts file or a number of different network related configuration problems.

If all else fails, you can try specifying the `java.rmi.server.hostname` system property. Note that this can affect applications running on the JVM.

Issue 3 I am getting exceptions during startup about classes not being found. What is wrong?

Ensure that you are using `JAVA_HOME/bin/jmc` to start the JMC client.

Issue 4 Why does Java Mission Control fail to find any local JVMs?

Consider the following:

- Make sure that you are using `JAVA_HOME/bin/jmc` to start the JMC client.
- If you are running JMC from Eclipse, make sure that Eclipse is running on a JDK (not JRE).
- Make sure that there is a directory named `hsperfdata_username` in the system's `tmp` directory, that it is writable by the user running JMC, and that the file system supports access control lists (ACLs).

Issue 5 How do I connect to JBoss?

Before starting JMC, you need to add the `-Xbootclasspath/a` option to the `-vmargs` section of the `jmc.ini` file. The value for this option (after a colon) should be set to the extended classpath that is generated by JBoss's `jconsole.bat` file (for this, add `@echo on` to the file and run it).