



M6-JMX 600.001

# AutoPilot M6 for JMX Installation and User's Guide Version 6.1

**CONFIDENTIALITY STATEMENT:** THE INFORMATION WITHIN THIS MEDIA IS PROPRIETARY IN NATURE AND IS THE SOLE PROPERTY OF NASTEL TECHNOLOGIES, INC. ALL PRODUCTS AND INFORMATION DEVELOPED BY NASTEL ARE INTENDED FOR LIMITED DISTRIBUTION TO AUTHORIZED NASTEL EMPLOYEES, LICENSED CLIENTS, AND AUTHORIZED USERS. THIS INFORMATION (INCLUDING SOFTWARE, ELECTRONIC AND PRINTED MEDIA) IS NOT TO BE COPIED OR DISTRIBUTED IN ANY FORM WITHOUT THE EXPRESSED WRITTEN PERMISSION FROM NASTEL TECHNOLOGIES, INC.



**PUBLISHED BY:**

RESEARCH & DEVELOPMENT  
NASTEL TECHNOLOGIES, INC.  
48 SOUTH SERVICE ROAD, SUITE 205  
MELVILLE, NY 11747

COPYRIGHT © 2001-2006. ALL RIGHTS RESERVED. NO PART OF THE CONTENTS OF THIS DOCUMENT MAY BE PRODUCED OR TRANSMITTED IN ANY FORM, OR BY ANY MEANS WITHOUT THE WRITTEN PERMISSION OF NASTEL TECHNOLOGIES.

DOCUMENT TITLE: **AUTOPILOT M6 FOR JMX**

VERSION: **VERSION 6.1**

DOCUMENT RELEASE DATE: **OCTOBER 2008**

NASTEL DOCUMENT NUMBER: **M6-JMX 600.001**

**CONFIDENTIALITY STATEMENT:** THE INFORMATION WITHIN THIS MEDIA IS PROPRIETARY IN NATURE AND IS THE SOLE PROPERTY OF NASTEL TECHNOLOGIES, INC. ALL PRODUCTS AND INFORMATION DEVELOPED BY NASTEL ARE INTENDED FOR LIMITED DISTRIBUTION TO AUTHORIZED NASTEL EMPLOYEES, LICENSED CLIENTS, AND AUTHORIZED USERS. THIS INFORMATION (INCLUDING SOFTWARE, ELECTRONIC AND PRINTED MEDIA) IS NOT TO BE COPIED OR DISTRIBUTED IN ANY FORM WITHOUT THE EXPRESSED WRITTEN PERMISSION FROM NASTEL TECHNOLOGIES, INC.

**ACKNOWLEDGEMENTS:**

THE FOLLOWING TERMS ARE TRADEMARKS OF NASTEL TECHNOLOGIES CORPORATION IN THE UNITED STATES OR OTHER COUNTRIES OR BOTH: NASTEL AUTOPILOT, AUTOPILOT M6, M6 WEB SERVER, M6 WEB CONSOLE, M6 FOR WMQ, MQCONTROL.

THE FOLLOWING TERMS ARE TRADEMARKS OF THE IBM CORPORATION IN THE UNITED STATES OR OTHER COUNTRIES OR BOTH: IBM, MQ, WEBSPHERE MQ, WIN-OS/2, AS/400, OS/2, DB2, INFORMIX, AIX, AND z/OS.

JAVA, J2EE, AND THE JAVA LOGOS ARE TRADEMARKS OF SUN MICROSYSTEMS INC. IN THE UNITED STATES OR OTHER COUNTRIES, OR BOTH.

INSTALLANYWHERE IS A REGISTERED TRADEMARK OF ZEROG SOFTWARE IN THE UNITED STATES OR OTHER COUNTRIES, OR BOTH.

THIS PRODUCT INCLUDES SOFTWARE DEVELOPED BY THE APACHE SOFTWARE FOUNDATION ([HTTP://WWW.APACHE.ORG/](http://www.apache.org/)); INCLUDING DERBY DATABASE SERVER. THE "JAKARTA PROJECT" AND "TOMCAT" AND THE ASSOCIATED LOGOS ARE REGISTERED TRADEMARKS OF THE APACHE SOFTWARE FOUNDATION.

INTEL, PENTIUM AND INTEL486 ARE TRADEMARKS OR REGISTERED TRADEMARKS OF INTEL CORPORATION IN THE UNITED STATES, OR OTHER COUNTRIES, OR BOTH.

MICROSOFT, WINDOWS, WINDOWS NT, WINDOWS XP, THE WINDOWS LOGOS, MS SQL SERVER, AND VISUAL SOURCESAFE ARE REGISTERED TRADEMARKS OF THE MICROSOFT CORPORATION.

UNIX IS A REGISTERED TRADEMARK IN THE UNITED STATES AND OTHER COUNTRIES LICENSED EXCLUSIVELY THROUGH X/OPEN COMPANY LIMITED.

MAC, MAC OS, AND MACINTOSH ARE TRADEMARKS OF APPLE COMPUTER, INC., REGISTERED IN THE U.S. AND OTHER COUNTRIES.

"LINUX" AND THE LINUX LOGOS ARE REGISTERED TRADEMARKS OF LINUS TORVALDS, THE ORIGINAL AUTHOR OF THE LINUX KERNEL. ALL OTHER TITLES, APPLICATIONS, PRODUCTS, AND SO FORTH ARE COPYRIGHTED AND/OR TRADEMARKED BY THEIR RESPECTIVE AUTHORS.

JASPERREPORTS IS A PRODUCT OF JASPERSOFT CORPORATION.

ORACLE IS A REGISTERED TRADEMARK OF ORACLE CORPORATION AND/OR ITS AFFILIATES.

SYBASE IS A TRADEMARK OF SYBASE, INC. © INDICATES REGISTRATION IN THE UNITED STATES OF AMERICA.

MYSQL IS A REGISTERED TRADEMARK OF MYSQL AB IN THE UNITED STATES, THE EUROPEAN UNION AND OTHER COUNTRIES.

OPENREPORTS IS DISTRIBUTED UNDER THE GNU GENERAL PUBLIC LICENSE.

OTHER COMPANY, PRODUCT, AND SERVICE NAMES, MAY BE TRADEMARKS OR SERVICE MARKS OF OTHERS.

# Table of Contents

|   |           |
|---|-----------|
| <b>CHAPTER 1: INTRODUCTION.....</b>                       | <b>1</b>  |
| 1.1 HOW THIS GUIDE IS ORGANIZED .....                     | 1         |
| 1.2 HISTORY OF THIS DOCUMENT.....                         | 1         |
| 1.2.1 <i>User Feedback</i> .....                          | 1         |
| 1.3 RELATED DOCUMENTS .....                               | 1         |
| 1.4 RELEASE NOTES .....                                   | 1         |
| 1.5 INTENDED AUDIENCE.....                                | 1         |
| 1.6 SYSTEM REQUIREMENTS.....                              | 2         |
| 1.7 TERMS AND ABBREVIATIONS.....                          | 2         |
| 1.8 TECHNICAL SUPPORT.....                                | 2         |
| 1.9 CONVENTIONS .....                                     | 2         |
| <b>CHAPTER 2: ABOUT AUTOPILOT M6 FOR JMX.....</b>         | <b>3</b>  |
| 2.1 AUTOPILOT M6 FOR JMX.....                             | 3         |
| <b>CHAPTER 3: AUTOPILOT M6 FOR JMX INSTALLATION .....</b> | <b>5</b>  |
| 3.1 BEFORE INSTALLATION.....                              | 5         |
| 3.1.1 <i>Technical Documents</i> .....                    | 5         |
| 3.1.2 <i>Installation Recommendations</i> .....           | 5         |
| 3.1.3 <i>Download JMX</i> .....                           | 5         |
| 3.2 INSTALLING M6 JMX .....                               | 5         |
| 3.3 JAVA APPLICATION INSTRUMENTATION.....                 | 6         |
| 3.4 JMX INSTRUMENTATION .....                             | 6         |
| 3.5 MANAGED BEAN PACKAGING AND DEPLOYMENT .....           | 8         |
| 3.6 MANAGED BEAN INSTANTIATION AND LOADING.....           | 8         |
| <b>CHAPTER 4: USING AUTOPILOT M6 FOR JMX .....</b>        | <b>9</b>  |
| 4.1 DEPLOY JMX AGENT AND JMX MONITOR .....                | 9         |
| <b>CHAPTER 5: AUTOPILOT JMX METRICS .....</b>             | <b>15</b> |
| 5.1 AUTOPILOT M6 JMX MONITOR METRICS .....                | 16        |
| <b>APPENDIX A: REFERENCES.....</b>                        | <b>17</b> |
| A.1 NASTEL DOCUMENTATION.....                             | 17        |
| A.2 JAVA™.....  | 17        |
| <b>APPENDIX B: CONVENTIONS.....</b>                       | <b>19</b> |
| B.1 TYPOGRAPHICAL CONVENTIONS.....                        | 19        |
| <b>GLOSSARY .....</b>                                     | <b>21</b> |

## Figures

---

|  |    |
|--|----|
| FIGURE 3-1. LAX.CLASS.PATH .....                       | 6  |
| FIGURE 3-2. RUNNING THE APJMX UTILITY.....             | 7  |
| FIGURE 3-3. JMX_MBEANS\MBEANS.JMA SAMPLE FILE .....    | 8  |
| FIGURE 4-1. DEPLOY JMX AGENT .....                     | 9  |
| FIGURE 4-2. JMX AGENT GENERAL PROPERTIES .....         | 10 |
| FIGURE 4-3. JMX AGENT OPTIONS PROPERTIES.....          | 11 |
| FIGURE 4-4. DEPLOY JMX MONITOR .....                   | 12 |
| FIGURE 4-5. JMX MONITOR GENERAL PROPERTIES .....       | 12 |
| FIGURE 4-6. JMX MONITOR JMX OPTIONS.....               | 13 |
| FIGURE 5-1. JMX METRICS COLLECTED BY JMX EXPERTS ..... | 15 |

## Tables

---

|   |    |
|---|----|
| 1-1. DOCUMENT HISTORY .....                         | 1  |
| 4-1. JMX AGENT: GENERAL.....                        | 10 |
| 4-2. JMX AGENT: JMX OPTIONS .....                   | 11 |
| 4-3. JMX MONITOR: GENERAL.....                      | 12 |
| 4-4. JMX MONITOR: JMX OPTIONS .....                 | 13 |
| 5-1. AUTOPILOT M6 JMX FACTS .....                   | 16 |
| 5-2. AUTOPILOT M6 JMX FACTS: DEFAULT SERVICES ..... | 16 |
| B-1 TYPOGRAPHICAL CONVENTIONS.....                  | 19 |



# Chapter 1: Introduction

Welcome to the AutoPilot M6 for JMX Guide. This guide explains how to update and perform new installations. It is designed to work with AutoPilot M6, its components and other plug-ins, and run simultaneously without interference or performance degradation. Please review this guide carefully before installing the product.

## 1.1 How This Guide is Organized

*Chapter 1:* Identifies the users and history of the document, as well as additional and alternate documents. System requirements are outlined in addition to supplying support and reference information. All other system and platform information is listed in the AutoPilot M6 Installation and User's Guides.

*Chapter 2:* Contains a brief functional description of AutoPilot M6 JMX.

*Chapter 3:* Provides instructions for new installations of AutoPilot M6 JMX.

*Chapter 4:* Provides post-installation set-up and configuration instructions.

*Appendix A:* Provides a detailed list of all reference information required for the installation of AutoPilot M6 JMX.

*Appendix B:* Contains conventions used in this document.

*Glossary:* Contains a listing of unique and common acronyms and words and their definition.

## 1.2 History of This Document

| 1-1. Document History |                 |                       |                 |
|-----------------------|-----------------|-----------------------|-----------------|
| Release Date:         | Document Number | For AutoPilot Version | Summary         |
| June 2004             | AP/JMX 430.001  | AP 4.0 and higher     | Initial release |
| October 2008          | M6/JMX 600.001  | M6                    | Update for M6   |

### 1.2.1 User Feedback

Nastel encourages all Users and Administrators of AutoPilot to submit comments, suggestions, corrections and recommendations for improvement for all AutoPilot documentation. Please send your comments via Post/Mail, or by e-mail. Send messages to: [support@nastel.com](mailto:support@nastel.com). You will receive a written response, along with status of any proposed change, update, or correction.

## 1.3 Related Documents

The complete listing of related and referenced documents is listed in Appendix A of this guide.

## 1.4 Release Notes

See README.HTM files on installation media or AutoPilot installation directory.

## 1.5 Intended Audience

This document is intended for personnel installing and customizing AutoPilot. The installer should be familiar with:

- Java Run Time Environment 1.5.1 (JRE 1.5.1) or higher (included in AutoPilot M6 for Windows, Solaris, AIX, HP-UX and Linux).
- Target operating system environment.
- The installed may need administrative privileges for the target platform.
- Procedures for installing software on the target platform such as Windows, UNIX, Mac OS etc.

## 1.6 System Requirements

AutoPilot M6 JMX 6.1 requires the following:

- AutoPilot M6 or higher
- AutoPilot M6 CEP Server installed on the same machine as JMX
- JDK 1.5.1 or higher compiler
- Approximately 510k of disk space

## 1.7 Terms and Abbreviations

A list of Terms and Abbreviation used in this document is located in the Glossary.

## 1.8 Technical Support

If you need technical support, you can contact Nastel Technologies by telephone or by e-mail. To contact Nastel technical support by telephone, call  [\(800\) 963-9822 ext. 1](tel:(800)963-9822), if you are calling from outside the United States dial [001-631-761-9190](tel:001-631-761-9190). To contact Nastel technical support by e-mail, send a message to <mailto:support@nastel.com>. To access the Nastel automated support system (user id and Password required), go to: <http://support.nastel.com/>. Contact your local AutoPilot M6 Administrator for further information.

## 1.9 Conventions

Refer to Appendix B for typographical and naming conventions used in all AutoPilot documentation.

## Chapter 2: About AutoPilot M6 for JMX

---

This chapter describes JMX and its application in AutoPilot M6.

### 2.1 AutoPilot M6 for JMX

**AutoPilot M6 for JMX** is designed to monitor JMX enabled applications. JMX is Java Management Extension interface supported by a wide variety of applications including BEA WebLogic, IBM WebSphere Application Server as well many other applications. AutoPilot M6 JMX provides the required facilities to instrument, monitor, and host JMX MBeans.

#### Features

- JMX Agent capable of hosting custom JMX MBeans (Management Beans)
- Monitor and control JMX enabled applications
- Integration with AutoPilot M6 rules and notification engine
- Generate custom events, alerts and actions
- Instrument Java applications with JMX using GUI wizard
- Access to Java application attributes and actions

#### Benefits

- Monitor and control user written Java applications
- Monitor and control any JMX enabled applications without coding
- Manage enable your Java applications within minutes

**AutoPilot M6 JMX** provides facilities for instrumentation; deployment and monitoring of JMX enabled applications. It contains the following components:

- **APJMX:** JMX MBean instrumentation wizard. Utility is located in [*AUTOPILOT\_HOME*]\bin directory. This utility can be used to instrument java classes with JMX dynamic managed bean wrapper.
- **JMX Agent:** hosts and instantiates managed beans (MBeans). Nasted JMX agent provides bean instantiation and persistence services as well as connectors for monitoring and communication.
- **JMX Monitor:** connects to JMX agent, gathers, and publishes metrics exposed via JMX interface. JMX Monitor uses RMI connector to communicate with JMX agents.

JMX agent and monitor components are installed as part of AutoPilot M6 JMX package. The package should be installed on the CEP server where mbeans will be installed and hosted.

**This Page Intentionally Left Blank**

# Chapter 3: AutoPilot M6 for JMX Installation

This chapter covers the installation and setup requirements for AutoPilot M6 JMX.

## 3.1 Before Installation

### 3.1.1 Technical Documents

Prior to installation, you should review all text files and installation procedures on the installation media or files provided. You should print all of the installation related materials to give yourself quick access to any required information during any installation or migration procedures.

Additional sets of documents are available from your Nastel representative or Nastel Support.

### 3.1.2 Installation Recommendations

Nastel recommends that you observe the following update sequence when updating AutoPilot M6 with patches, updates and service packs:

- Machines running the Domain Server
- CEP Servers

### 3.1.3 Download JMX

Download JMX from the Nastel Resource Center or copy from your installation media.

## 3.2 Installing M6 JMX

1. Save your work and logoff AutoPilot M6.

|   |              |  |
|---|--------------|--|
|  | <b>NOTE:</b> | There are no specific logoff procedures required to exit AutoPilot M6 Console. |
|---|--------------|--|

2. Stop the CEP servers and/or domain servers that will be updated as specified in the *AutoPilot M6 Installation Guide*.
3. Copy `AP_JMX-6.0.1.pkg` into the `[AUTOPILOT_HOME]\updates` directory.
4. At the command prompt run:  
`[AUTOPILOT_HOME]\bin\pkgman ..\updates\AP_JMX-6.0.1.pkg`

|   |              |  |
|---|--------------|--|
|  | <b>NOTE:</b> | Make sure there are no errors posted at the bottom of the screen. If errors occur re-run pkgman. |
|---|--------------|--|

5. Verify plug-in installation: `[AUTOPILOT_HOME]\bin\pkgman -info [AUTOPILOT_HOME]\bin\pkgman -verify [PackageName]`

|   |              |   |
|---|--------------|---|
|  | <b>NOTE:</b> | Make sure there are no errors posted at the bottom of the screen. |
|---|--------------|---|

## 3.3 Java Application Instrumentation

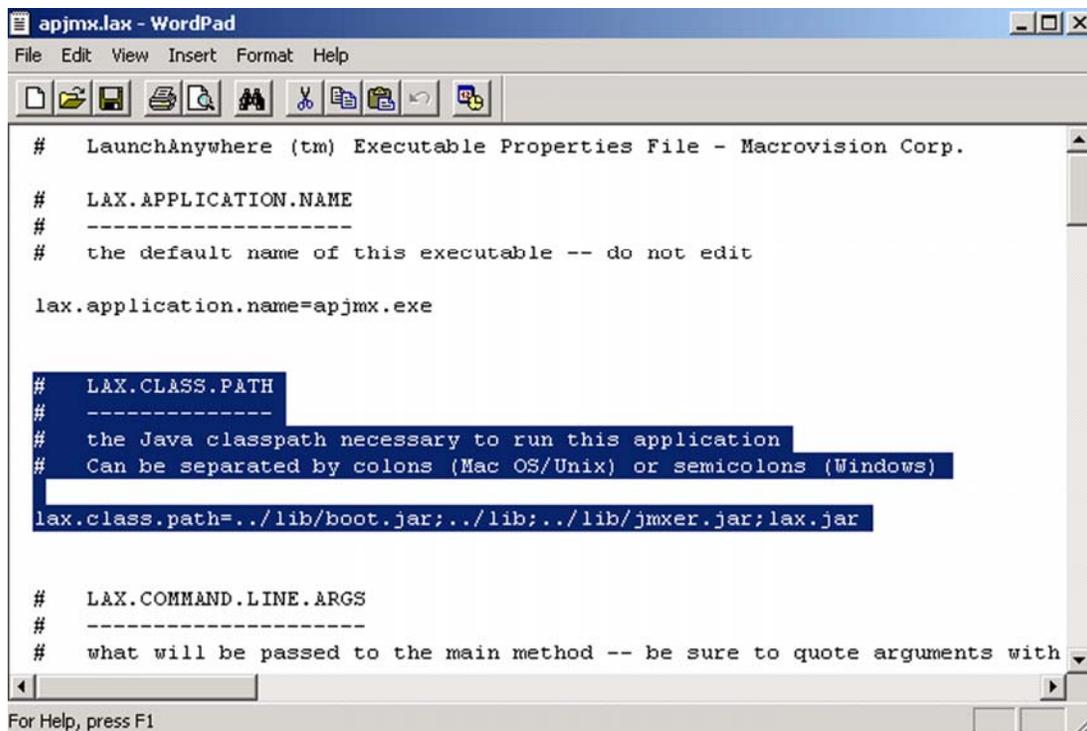
Any java application can be instrumented using JMX. AutoPilot M6 JMX is a GUI wizard designed to automate and simplify the instrumentation process. This process is fully JMX compliant and does not use any of the Nastel proprietary classes or interfaces. JMX instrumentation process requires the following steps:

1. JMX instrumentation of applications
2. Packaging mbeans into a deployable jar file
3. Deployment of JMX agent and monitor

## 3.4 JMX Instrumentation

The following configuration steps are required before running apjmx utility:

1. Edit [ *AUTOPILOT\_HOME* ]\bin\apjmx.lax file to specify class path to the application class that will be instrumented with JMX. AutoPilot M6 JMX uses class introspection to determine class variables, methods and exceptions. Add your jar file or class path to the lax.class.path variable.



```

apjmx.lax - WordPad
File Edit View Insert Format Help

# LaunchAnywhere (tm) Executable Properties File - Macrovision Corp.

# LAX.APPLICATION.NAME
# -----
# the default name of this executable -- do not edit

lax.application.name=apjmx.exe

# LAX.CLASS.PATH
# -----
# the Java classpath necessary to run this application
# Can be separated by colons (Mac OS/Unix) or semicolons (Windows)
lax.class.path=../lib/boot.jar;../lib;../lib/jmxer.jar;lax.jar

# LAX.COMMAND.LINE.ARGS
# -----
# what will be passed to the main method -- be sure to quote arguments with
  
```

Figure 3-1. LAX.CLASS.PATH

- Run the apjmx utility and specify fully qualified name of your application class, as well as the name of the target JMX wrapper class. The wizard will walk you through several steps during which you specify which variables and methods to expose via JMX interface.



Figure 3-2. Running the APJMX Utility

- Once the instrumentation is complete, AutoPilot M6 JMX will create a target mbean java source file under `[AUTOPILOT_HOME]\bin` directory. This file may be under subdirectories that reflect original application package name structure. The resulting file is a subclass of the original application with JMX extensions added on.
- Compile your mbean application. Make sure your class path includes JMX runtime classes supplied with AutoPilot installation:

```
CLASSPATH=<aphome>\lib\jmxri.jar;<aphome>\lib\jmx_remoting.jar;<your_classes>
```

|   |              |   |
|---|--------------|---|
|  | <b>NOTE:</b> | <b>MBean Persistence:</b> The JMX agent provides a persistence service that saves currently registered MBeans for auto-loading when the agent restarts. In order to qualify for auto-loading, an MBean must provide a default constructor that takes no parameters. However, note that MBeans registered in the "DefaultServices" domain do not qualify for auto-loading. |
|---|--------------|---|

## 3.5 Managed Bean Packaging and Deployment

Compiled JMX mbeans should be packed into a JAR file using the jar utility.

**Example:** `jar cvf mybeans.jar com\nastel\jmx\*.class` packages all classes under `com\nastel\jmx` directory into `mybeans.jar` file.

The resulting jar file should be copied into the `[AUTOPILOT_HOME]\jmx_mbeans` directory.



### IMPORTANT:

This folder only exists on CEP Servers which have AutoPilot M6 JMX package installed.

## 3.6 Managed Bean Instantiation and Loading

AutoPilot JMX agent will automatically loads mbeans specified in:

`[AUTOPILOT_HOME]\jmx_mbeans\mbeans.jma` file.

This file contains name and description of all persistent JMX mbeans. See example below:

```

mbeans.jma - Notepad
File Edit Format Help
<mlet name="Test:name=Frame" code=MFrameBean archive="mbeans.jar"></mlet>
<mlet name="Test:name=RandomNumber" code=RandomNumber archive="mbeans.jar"></mlet>

```

*Figure 3-3. jmx\_mbeans\mbeans.jma Sample File*

The general format is as follows:

```
<mlet name="domain_name:name=MyAppl" code=class_name archive="jarfile">
```

## Chapter 4: Using AutoPilot M6 for JMX

### 4.1 Deploy JMX Agent and JMX Monitor

1. Open your AutoPilot Console.
2. Right-click on the CEP server that has AutoPilot M6 JMX agent installed.
3. To deploy JMX Agent, click **Deploy Expert>JMX Monitors>JMX Agent**.

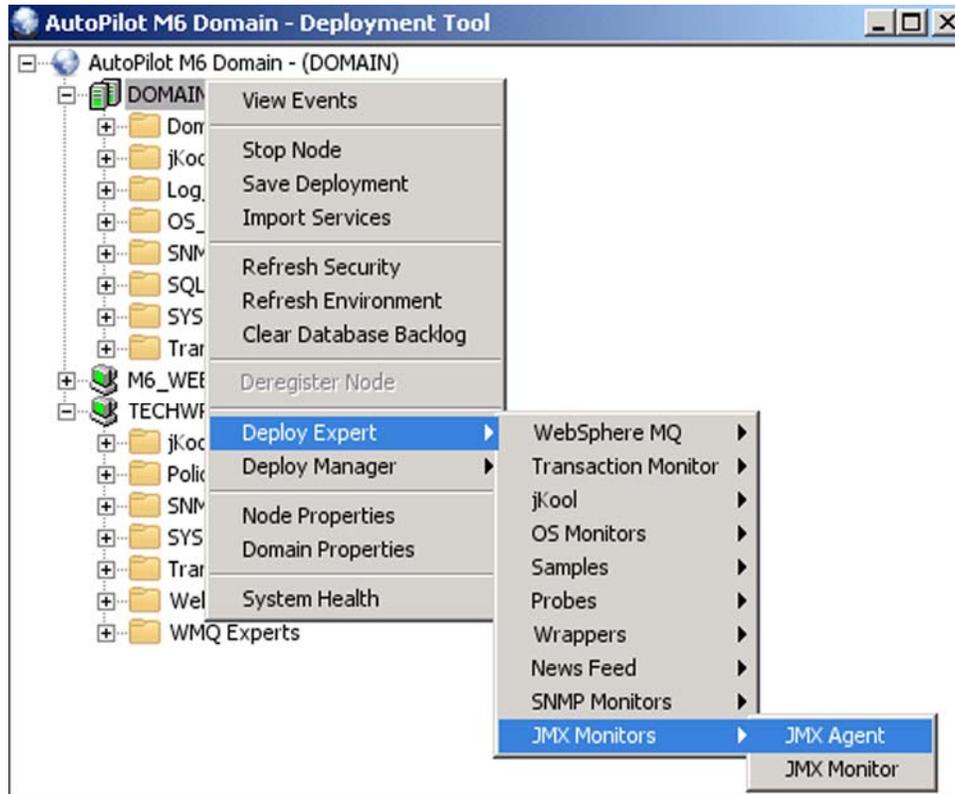


Figure 4-1. Deploy JMX Agent

4. It is recommended that you update all three properties to define you agent. At a minimum assign a unique name to your agent.

| <b>4-1. JMX Agent: General</b> |  |
|--------------------------------|--|
| <b>Property</b>                | <b>Description</b>   |
| <b>Brief description</b>       | A short, user-defined, description of the service.   |
| <b>Context</b>                 | User defined category that will be registered in the Domain Server. Context is displayed as folder icon under each CEP Server. Default is: JMX Experts.                  |
| <b>MBean sampling filter</b>   | Only MBeans whose names match the filter will be sampled. Default is all MBeans.   |
| <b>MBean sampling rate</b>     | MBean sampling rate in seconds. Default is 30 seconds.   |
| <b>MBean sampling</b>          | Enable/disable MBean attributes sampling and publishing as facts. Default is enabled.  |
| <b>Name</b>                    | Name that uniquely identifies the service in the domain. The default name system assigned with the word service and twelve random digits (Example: service_123456789012) |

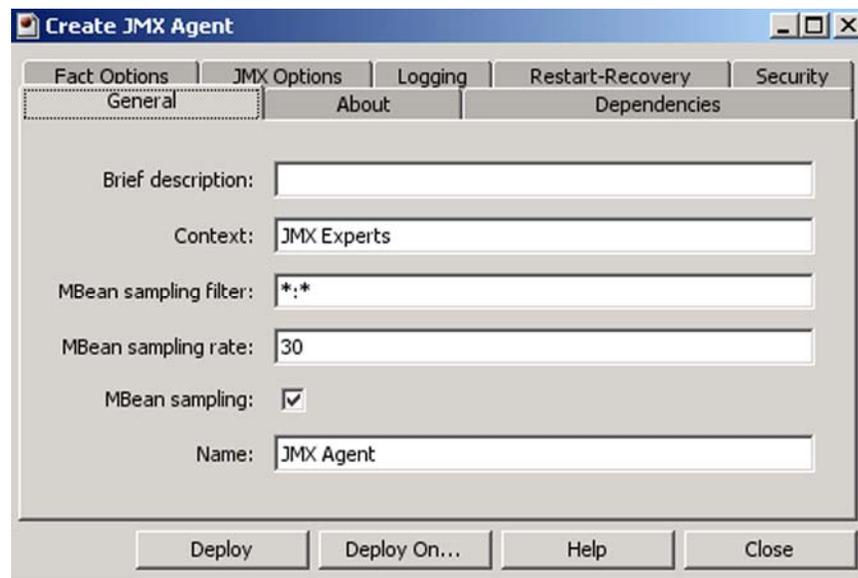


Figure 4-2. JMX Agent General Properties

5. Review the **JMX Options** tab, where the JMX mbean directory and RMI communication details are specified. It is recommended that the system-defined default settings not be changed.

| 4-2. JMX Agent: JMX Options |  |
|-----------------------------|--|
| Property                    | Description  |
| Agent working directory     | Working directory of this JMX agent.                       |
| Enable RMI                  | Enable/disable RMI connector server.                       |
| RMI port                    | The TCP/IP port number to use for the RMI Connector Server |
| RMI service name            | The service name to use for the RMI Connector Server       |
| Short name keys             | Comma separated keys for shortening MBean names.           |

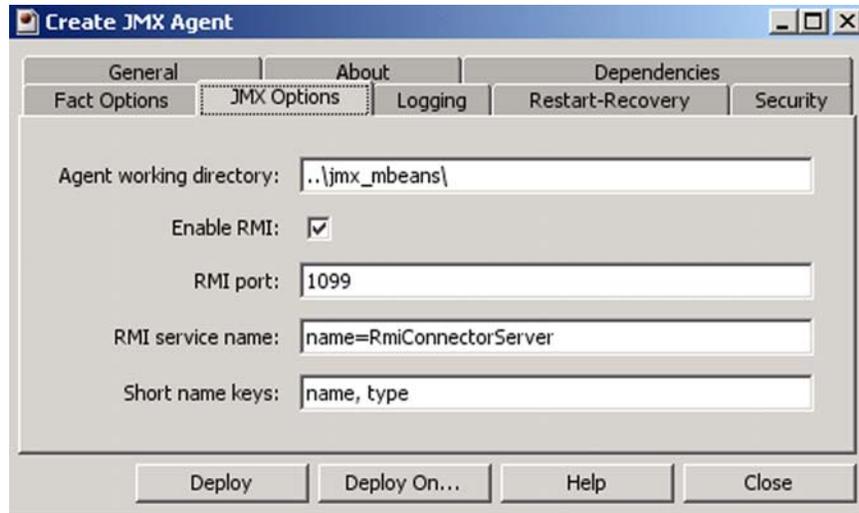


Figure 4-3. JMX Agent Options Properties

- 6 Right-click on the CEP server that has AutoPilot M6 JMX monitor installed.
- 7 To deploy JMX Monitor, click **Deploy Expert>JMX Monitors>JMX Monitor**.

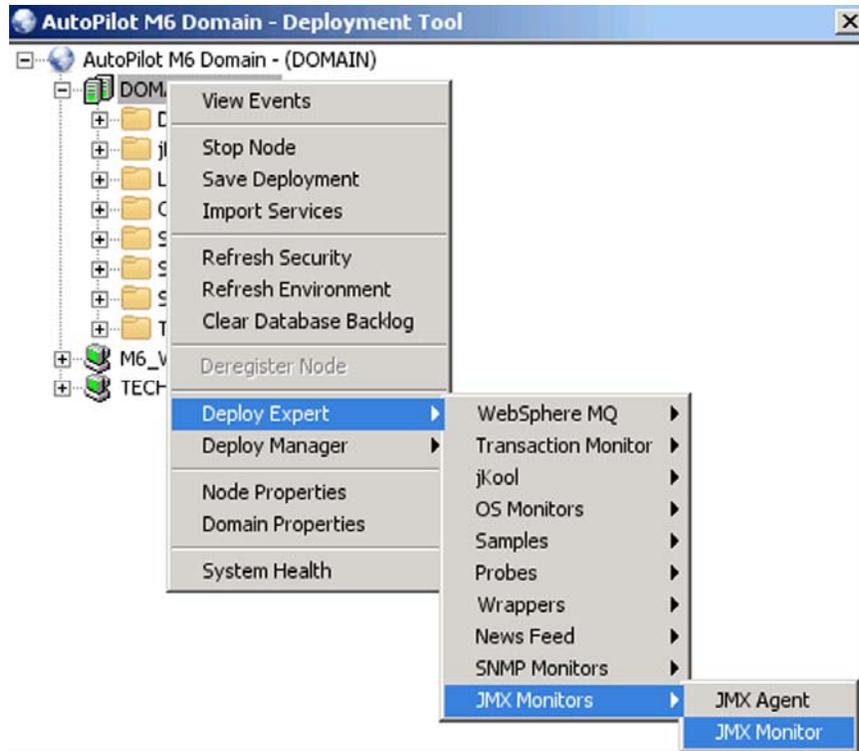


Figure 4-4. Deploy JMX Monitor

8. Specify the monitor name as well as JMX options. The JMX monitor can connect to any JMX agent, on any TCP/IP server. Leave defaults if both are running on the same server.
9. It is recommended that you update all three properties to define your agent. At a minimum assign a unique name to your monitor.

| 4-3. JMX Monitor: General |  |
|---------------------------|--|
| Property                  | Description  |
| <b>Brief description</b>  | A short, user defined, description of the service  |
| <b>Context</b>            | A user define category that will be registered with the domain server. The default is: JMX Experts   |
| <b>Name</b>               | Name that uniquely identifies the service in the domain. The default name system assigned with the word service and twelve random digits (Example: service_123456789012) |

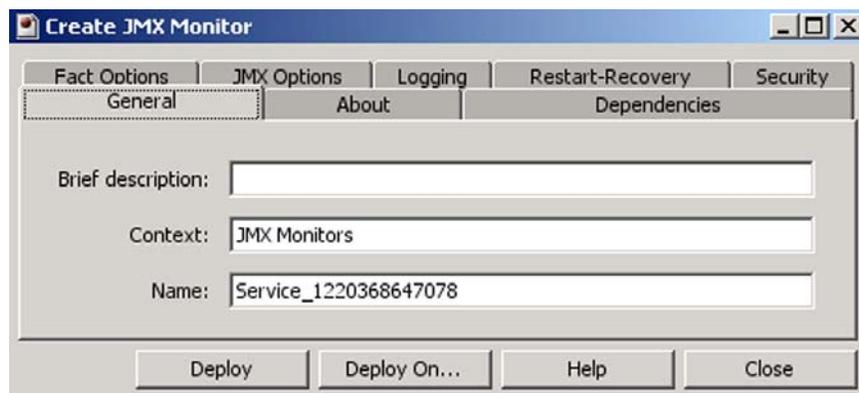


Figure 4-5. JMX Monitor General Properties

10. Review the **JMX Options** tab, where the JMX mbean directory and RMI communication details are specified. It is recommended that the system-defined default settings not be changed.

| 4-4. JMX Monitor: JMX Options |  |
|-------------------------------|--|
| Property                      | Description  |
| Connector host name           | Host name or IP address of the computer running the JMX agent and connector. Default is localhost. |
| Connector port                | TCP/IP port number of the connector on the JMX agent.  |
| MBean filter                  | Only MBeans whose names match the filter will be displayed. Default is all.                        |
| Sample time threshold (sec.)  | MBean sample time threshold in seconds. Triggers reconnect once crossed.                           |
| Sampling interval (sec.)      | Sampling frequency in seconds. Default is 300 seconds.   |
| Short name keys               | Comma separated keys for shortening MBean names. Default is name, type.                            |

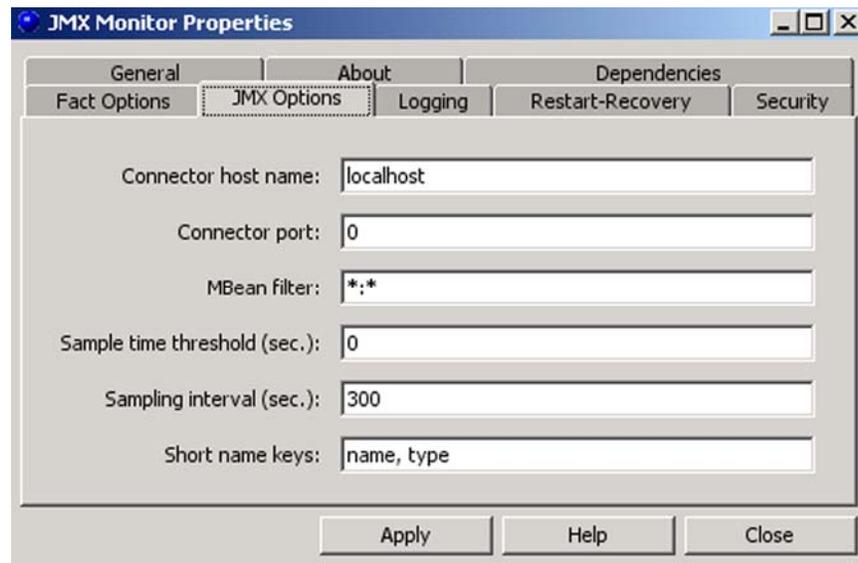


Figure 4-6. JMX Monitor JMX Options

Once the JMX agent and monitor are deployed, JMX monitor will connect and gather metrics from all mbeans under the specified JMX agent at a specified interval. That interval is configured in “JMX Options” of the JMX monitor instance. JMX monitor will also publish facts regarding each mbean.

These facts about your application can now be folded into AutoPilot business views for business rules, notifications and actions.

**This Page Intentionally Left Blank**

## Chapter 5: AutoPilot JMX Metrics

This section describes all AutoPilot JMX metrics collected by WebSphere MQ monitors. They are published as facts and are available under each expert as shown below:



Figure 5-1. JMX Metrics Collected by JMX Experts

Once published, these facts can be included in one or more business views for validation, automation, notifications, logging and alerts.

JMX experts publish facts in the following format:

```
Facts\<fact_category>\<fact_subcategory>\<qmgr_name>\<fact_name>\<value>
```

## 5.1 AutoPilot M6 JMX Monitor Metrics

| 5-1. AutoPilot M6 JMX Facts  |   |
|------------------------------|---|
| Metric                       | Description   |
| <b>DefaultServices</b>       | JMX domain that contains all default MBeans instantiated by the JMX Agent. Default services are required for proper MBean operation, deployment, and persistence. |
| <b>JMImplementation</b>      | JMX domain for JMX server implementation.   |
| <b>Samples/StaticSamples</b> | Example MBeans that are delivered as part of the JMXAgent. For demonstration purposes only.   |

| 5-2. AutoPilot M6 JMX Facts: Default Services |  |
|---|--|
| Metric  | Description  |
| <b>PersistenceService</b>                     | JMX Persistence Service for MBean persistence                                |
| <b>RelationService</b>                        | JMX Relation Service MBean   |
| <b>RMIConnectorService</b>                    | RMIConnectorService MBean for managing RMI connections to the JMX agent      |
| <b>MBeanLoader</b>                            | An internal JMX system MBean that comes as part of every JMX implementation. |

---

## Appendix A: References

---

### A.1 Nastel Documentation

M6/USR 600.007 *Nastel AutoPilot® M6 User's Guide*

M6/INS 600.007 *Nastel AutoPilot® M6 Installation Guide*

M6WMQ/INS 600.002 *Nastel AutoPilot® M6 for WebSphere MQ Installation Guide*

M6WMQ/ADM 600.002 *Nastel AutoPilot® M6 for WebSphere MQ Administrator's Guide*

M6WMQ/SM 600.002 *Nastel AutoPilot® M6 for WebSphere MQ Security Manager User's Guide*

M6/WMQ 600.001 *Nastel AutoPilot® M6 Plug-in for WebSphere MQ*

### A.2 Java™

<http://java.sun.com/products/JavaManagement/reference/docs/index.html>

<http://www.hp.com/products1/unix/java/infolibrary/index.html>

<http://developer.java.sun.com/developer/technicalArticles/Servlets/corba/>

**This Page Intentionally Left Blank**

# Appendix B: Conventions

## B.1 Typographical Conventions

| B-1 Typographical Conventions   |  |
|---------------------------------|--|
| Convention                      | Description  |
| <a href="#">Blue/Underlined</a> | Used to identify links to referenced material or websites.<br>Example: <a href="mailto:support@nastel.com">support@nastel.com</a>                          |
| <b>Bold Print</b>               | Used to identify topical headings, glossary entries, toggles or buttons used in procedural steps.<br>Example: Click <b>EXIT</b> .                          |
| <i>Italic Print</i>             | Used to place emphasis on titles, menus, screen names, or other categories.  |
| <b>Monospaced Bold</b>          | Used to identify keystrokes/data entries, file names, directory names etc.   |
| <i>Monospaced Italic</i>        | Used to identify variables in an address location.<br>Example: [C:\AutoPilot_Home]\documents.<br>Where the portion of address in brackets [ ] is variable. |
| Monospaced Text                 | Used to identify addresses, commands, scripts etc.   |
| Normal Text                     | Typically used for general text throughout the document.   |
| Table Text                      | Table text is generally a smaller size to conserve space. 10, 9, and 8 point type is used in tables through the AutoPilot product family of documents.     |

**This Page Intentionally Left Blank**

# Glossary

---

**Application Programming Interface (API):** a source code interface that an operating system, library or service provides to support requests made by computer programs.

**API:** *see* Application Programming Interface.

**Asynchronous:** Communication between computer and devices that can occur at any time and at irregular intervals.

**AutoPilot M6:** Nastel Technologies' Enterprise Application Management Platform. AutoPilot M6 is designed to monitor and control distributed IT services such as application servers, middleware, user applications, workflow engines, brokers, Service Oriented Architecture (SOA) and Enterprise Service Bus (ESB) based applications and their impact on business services.

**AutoPilot M6 for WMQ:** Nastel Technologies' WebSphere MQ management solution. Re-designated as M6 for WMQ with release 6.0, prior releases retain the AP-WMQ or MQControl trademark.

**AutoPilot M6 Web:** A browser-based interface that provides monitoring and operational control over managed resources and applications. It allows users to monitor health, recover from a failure, view historical performance graphs and visualize impacts of a failure.

**BSV:** *see* Business View.

**Business View (BSV):** A collection of rules that define a desired state of an eBusiness environment. Business Views can be tailored to present information in the form most suited to a given user, as defined by the user.

**CEP:** *see* Complex Event Processing.

**CEP Server:** A container that can host any number of AutoPilot M6 services such as experts, managers, policies, etc.

**Client:** Any programming component that uses the AutoPilot M6 infrastructure; for example, the AutoPilot M6 User Console.

**Complex Event Processing (CEP):** A technology for building and managing event-driven information systems. CEP is primarily an event processing concept that deals with the task of processing multiple events from an event cloud with the goal of identifying the meaningful events within the event cloud. CEP employs techniques such as detection of complex patterns of many events, event correlation and abstraction, event hierarchies, and relationships between events such as causality, membership, and timing, and event-driven processes.

**Console:** The console acts as the graphical interface for AutoPilot M6.

**Contacts:** A subordinate to a given Manager or Expert.

**Data Source Name (DSN):** Logical name that is used by Open Database Connectivity (ODBC) to refer to the drive and other information that is required to access data. The name is used by Internet Information Services (IIS) for a connection to an ODBC data source, (Example: Microsoft SQL Server database). The ODBC tool in Control Panel is used to set the DSN. When ODBC DSN entries are used to store the connection string values externally, you simplify the information that is needed in the connection string. This makes changes to the data source completely transparent to the code itself.

**Decision Support System (DSS):** An AutoPilot M6-based service designed to monitor, store, and display any event information generated by AutoPilot M6-enabled middleware and applications.

**Deploy:** To put to use, to position for use or action.

**Domain Server:** A specialized CEP server that maintains the directory of CEP servers, experts etc. The domain server is also capable of hosting experts, managers etc.

**DSN:** *see* Data Source Name

**DSS:** *see* Decision Support System

**Enterprise Service Bus (ESB):** A software architecture construct, implemented by technologies found in a category of middleware infrastructure products usually based on standards that provides foundational services for more complex architectures via an event-driven and standards-based messaging engine.

**ESB:** *see* Enterprise Service Bus

**EVT:** Event Log file extension (e.g.: sample.evt)

**Event:** Something that happens to an object. Events are logged by AutoPilot M6 and are available for use by AutoPilot M6 Policies or the user.

**Expert:** Services that monitor specific applications such as an applications server, web-server or specific components within the applications -- (Example - channels in WebSphere MQ). Experts generate facts.

**Facts:** Single pieces of data that have a unique name and value. One or more facts are used to determine the health of the object, application or server.

**File Monitor:** Built-in expert that monitors application error logs and publishes logs to AutoPilot M6 as facts and events.

**Graphical User Interface (GUI):** A type of environment that represents programs, files, and options by means of icons, menus, and dialog boxes on the screen. The user can select and activate these options by pointing and clicking with a mouse or, often, with the keyboard. Because the graphical user interface provides standard software routines to handle these elements and report the user's actions (such as a mouse click a particular icon or at a particular location in text, or a key press); applications call these routines with specific parameters rather than attempting to reproduce them from scratch.

**Grids:** In AutoPilot M6, a collection of clusters that allows users to define and automate CEP server failover. Defined under domain server\naming\grid folder.

**GUI:** *see* Graphical User Interface.

**IIS:** *see* Internet Information Services

**Internet Information Services (IIS):** Microsoft's brand of Web server software, utilizing HTTP to deliver World Wide Web documents. It incorporates various functions for security, allows CGI programs, and also provides for Gopher and FTP services.

**Java EE:** *see* Java Platform, Enterprise Edition.

**Java:** A platform-independent, object-oriented programming language developed and made available by Sun Microsystems.

**Java Database Connectivity (JDBC):** The JDBC API provides universal data access from the Java programming language. Using the JDBC 2.0 API, you can access virtually any data source, from relational databases to spreadsheets and flat files. JDBC technology also provides a common base on which tools and alternate interfaces can be built. The *JDBC Test Tool* that was developed by Merant and Sun Microsystems may be used to test drivers, to demonstrate executing queries and getting results, and to teach programmers about the JDBC API.

**Java Developer's Kit (JDK):** A set of software tools developed by Sun Microsystems, Inc., for writing Java applets or applications. The kit, which is distributed free, includes a Java compiler, interpreter, debugger, viewer for applets, and documentation.

**Java Management Extensions (JMX):** Java technology that supplies tools for managing and monitoring applications, system objects, devices (e.g. printers) and service oriented networks. Those resources are represented by objects called MBeans (for *Managed Bean*).

**Java Message Service (JMS):** Java message-oriented middleware API for sending messages between two or more clients.

**Java Platform, Enterprise Edition (Java EE):** A widely used platform for server programming in the Java programming language. The Java EE Platform differs from the Standard Edition (SE) of Java in that it adds additional libraries which provide functionality to deploy fault-tolerant, distributed, multi-tier Java software, based largely on modular components running on an application server.

**Java Server Pages (JSP):** JSP technology enables rapid development of web-based applications that are platform independent. Java Server Pages technology separates the user interface from content generation enabling designers to change the overall page layout without altering the underlying dynamic content. Java Server Pages technology is an extension of the Java™ Servlet technology.

**Java Virtual Machine (JVM):** The "virtual" operating system that JAVA-written programs run. The JVM is a hardware- and operating system-independent abstract computing machine and execution environment. Java programs execute in the JVM where they are protected from malicious programs and have a small compiled footprint.

**JDBC:** *see* Java Database Connectivity

**JDK:** *see* Java Developer's Kit.

**JMS:** *see* Java Message Service.

**JMX:** *see* Java Management Extensions.

**JRE:** JAVA Run-time Environment. The minimum core JAVA required to run JAVA Programs.

**JSP:** *see* Java Server Pages

**JVM:** *see* JAVA Virtual Machine.

**Kerberos Authentication:** Mutual authentication between server and client without exchanging a password. Kerberos prevents eavesdropping or replay attacks, and ensures the integrity of the data. Its designers aimed primarily at a client-server model, and it provides mutual authentication — both the user and the server verify each other's identity. Kerberos builds on symmetric key cryptography and requires a trusted third party.

**LDAP:** *see* Lightweight Directory Access Protocol.

**Lightweight Directory Access Protocol (LDAP):** A networking protocol for querying and modifying directory services running over TCP/IP.

**Manager: Managers** are the home or container for policies. All business views must reside on managers, and managers must be deployed prior to deploying a business view or policy.

**Message-Oriented Middleware (MOM):** Message-oriented middleware (MOM) is a client/server infrastructure that increases the interoperability, portability, and flexibility of an application by allowing the application to be distributed over multiple heterogeneous platforms.

**Message Queue Interface (MQI):** The Message Queue Interface (MQI) is part of IBM's Networking Blueprint. It is a method of program-to-program communication suitable for connecting independent and potentially non-concurrent distributed applications.

**MOM:** *see* Message-Oriented Middleware.

**MQI:** *see* Message Queue Interface

**Naming Service:** A common server records "names" of objects and associates them with references, locations and properties.

**Object Request Broker (ORB):** A piece of middleware software that allows programmers to make program calls from one computer to another via a network.

**ODBC:** *see* Open Database Connectivity.

**Open Database Connectivity (ODBC):** Provides a standard software API method for using database management systems.

**ORB:** *see* Object Request Broker.

**Package Manager:** The command line utility that allows users to list, install, uninstall, verify and update AutoPilot M6 installation on any CEP server.

**PKGMAN:** *see* Package Manager

**Policy/Business Views:** Business views are a collection of one or more sensors. Business views are used to visually present the health and status of the different systems as well as automatically issue remedial actions.

**Process Wrapper:** Built-in wrapper that monitors a process or script started by AutoPilot M6.

**Sensor:** A rule that is used to determine the health of an object or application based on one or more facts. Actions can then be issued, based on the health.

**Service-Oriented Architecture (SOA):** An evolution of distributed computing and modular programming. SOAs build applications out of software services. Services are relatively large, intrinsically unassociated units of functionality, which have no calls to each other embedded in them. They typically implement functionalities most humans would recognize as a service, such as filling out an online application for an account, viewing an online bank statement, or placing an online book or airline ticket order. Instead of services embedding calls to each other in their source code, protocols are defined which describe how one or more services can talk to each other. This architecture then relies on a business process expert to link and sequence services, in a process known as orchestration, to meet a new or existing business system requirement.

**Simple Mail Transfer Protocol (SMTP):** A TCP/IP protocol for sending messages from one computer to another on a network. This protocol is used on the Internet to route e-mail. *See also* communications protocol, TCP/IP. *Compare* CCITT X series, Post Office Protocol.

**SMTP:** *see* Simple Mail Transfer Protocol

**SOA:** *see* Service-Oriented Architecture

**Synchronous:** communication within a computer that occurs at regular intervals. Usually governed by the microprocessor clock.

**TCP/IP:** *see* Transmission Control Protocol/Internet Protocol.

**Transmission Control Protocol/Internet Protocol (TCP/IP):** A protocol developed by the Department of Defense for communications between computers. It is built into the UNIX system and has become the de facto standard for data transmission over networks, including the Internet.

**UDP:** *see* Universal Datagram Protocol.

**Universal Datagram Protocol (UDP):** A connectionless protocol that runs on top of IP networks. Provides very few error recovery services, offering instead a direct way to send and receive datagrams over an IP network. It's used primarily for broadcasting messages over a network.

**Virtual Machine (VM):** Software that mimics the performance of a hardware device, such as a program that allows applications written for an Intel processor to be run on a Motorola chip. *Also See* Java Virtual Machine.

**Visual Source Safe (VSS):** Microsoft VSS helps you manage your projects by saving them to a database. When you need to share files between two or more projects, you can share them quickly and efficiently. When you add a file to VSS, the file is backed up on the database, made available to other people, and changes that have been made to the file are saved so you can recover an old version at any time. Members of your team can see the latest version of any file, make changes, and save a new version in the database.

**VM:** *see* Virtual Machine

**VSS:** *see* Visual Source Safe.

**WebSphere MQ:** IBM's message queuing product.

**Websphere\_MQ\_Manager:** A specialized manager capable of hosting one or more MQSeries specific policies, apart from the regular policies.

**Wireless Application Protocol (WAP):** An open global specification that is used by most mobile telephone manufacturers. WAP determines how wireless devices utilize Internet content and other services. WAP enables devices to link diverse systems contents and controls.

**XEN:** A free software virtual machine monitor for IA-32, x86-64, IA-64 and PowerPC architectures. It is software that runs on a host operating system and allows several guest operating systems to be run on top of the host on the same computer hardware at the same time. Modified versions of Linux and NetBSD can be used as hosts.