



IBM Tivoli Monitoring  
*User's Guide*  
*Version 5.1*

SH19-4569-00





IBM Tivoli Monitoring  
*User's Guide*  
*Version 5.1*

SH19-4569-00

## IBM Tivoli Monitoring User's Guide, Version 5.1

### Copyright Notice

© Copyright IBM Corporation 1999, 2002. All rights reserved. May only be used pursuant to a Tivoli Systems Software License Agreement, an IBM Software License Agreement, or Addendum for Tivoli Products to IBM Customer or License Agreement. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without prior written permission of IBM Corporation. IBM Corporation grants you limited permission to make hardcopy or other reproductions of any machine-readable documentation for your own use, provided that each such reproduction shall carry the IBM Corporation copyright notice. No other rights under copyright are granted without prior written permission of IBM Corporation. The document is not intended for production and is furnished "as is" without warranty of any kind. **All warranties on this document are hereby disclaimed, including the warranties of merchantability and fitness for a particular purpose.**

U.S. Government Users Restricted Rights—Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corporation.

### Trademarks

Tivoli, the Tivoli logo, AIX, OS/390, S/390, Tivoli Enterprise, Tivoli Enterprise Console, and TME are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Microsoft, Windows and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.



Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.

### Notices

References in this publication to Tivoli Systems or IBM products, programs, or services do not imply that they will be available in all countries in which Tivoli Systems or IBM operates. Any reference to these products, programs, or services is not intended to imply that only Tivoli Systems or IBM products, programs, or services can be used. Subject to valid intellectual property or other legally protectable right of Tivoli Systems or IBM, any functionally equivalent product, program, or service can be used instead of the referenced product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by Tivoli Systems or IBM, are the responsibility of the user. Tivoli Systems or IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, New York 10504-1785, U.S.A.

### Use of Rhino: JavaScript for Java

The use of Rhino: JavaScript for Java is governed by a licence fully detailed in Appendix E, "Use of Rhino: JavaScript for Java" on page 205.

### ISO 9001 Certification

This product was developed using an ISO 9001 certified quality system.

Certification has been awarded by Bureau Veritas Quality International (BVQI) (Certification No. BVQI - 92086 / A).

BVQI is a world leader in quality certification and is currently recognized by more than 20 accreditation bodies.

---

# Contents

<b>Figures</b> .....	<b>xi</b>
----------------------	-----------

<b>Tables</b> .....	<b>xiii</b>
---------------------	-------------

<b>Preface</b> .....	<b>xv</b>
----------------------	-----------

Who Should Read This Guide .....	xv
What is New in this Release .....	xv
Changes to This Edition of the User's Guide .....	xvi
What This Guide Contains .....	xvi
Publications .....	xviii
IBM Tivoli Monitoring Library.....	xviii
Prerequisite Publications.....	xviii
Accessing Publications Online .....	xix
Ordering Publications.....	xix
Providing Feedback about Publications .....	xx
Contacting Customer Support .....	xx
Conventions Used in This Guide .....	xx
Typeface Conventions .....	xx
Operating System-dependent Variables and Paths .....	xxi

---

<b>Part I. Installing, Configuring, and Using</b> .....	<b>1</b>
---	----------

<b>Chapter 1. Introduction</b> .....	<b>5</b>
--------------------------------------	----------

Overview .....	5
Main Features .....	6
Understanding Tivoli Monitoring.....	6
Resources .....	6
Resource Models.....	6
Cycles.....	7
Thresholds .....	7
Parameters .....	7
Indications .....	7
Occurrences and Holes .....	8
Events.....	8
Clearing Events.....	10
Monitoring of Events and Indications.....	10
Tivoli Enterprise Console Server .....	10
Web Health Console .....	11

	Tivoli Business Systems Manager .....	11
	Recovery Actions .....	11
	Built-In Actions.....	11
	Tivoli Framework Tasks .....	12
	Scheduling .....	12
	Logging .....	12
	Profiles .....	12
	The Heartbeat Function .....	12
I	Gathering Historical Data Component .....	12
	Components .....	13
	Tivoli Monitoring Base Component .....	14
	Web Health Console .....	15
	Tivoli Business Systems Manager Adapter .....	15
	Endpoint Component.....	15
	Collecting the Data .....	15
	Resource Model Data Flow .....	15
	Obtaining Resource Data at the Endpoint.....	16
I	Security Considerations .....	17
	Tivoli Monitoring Workbench .....	17
	<b>Chapter 2. Installing .....</b>	<b>19</b>
	Supported Platforms .....	19
	Hardware Requirements.....	20
	Prerequisite Software.....	20
	Tivoli Management Framework .....	21
	Tivoli Management Framework Patches Required for Version 3.7B .....	21
	Other Software .....	21
	Coexistence with Other Versions of Tivoli Monitoring .....	22
	Products within the Tivoli Monitoring Family .....	22
	Products within the Tivoli Distributed Monitoring (Classic Edition) Family .....	23
I	Backward Compatibility .....	23
	Installing/Upgrading the Product .....	23
	Before You Begin .....	24
	Installing with Tivoli Software Installation Service.....	24
	Installing from the Tivoli Desktop .....	25
	Installing from the Command Line .....	28
	Upgrading from Tivoli Distributed Monitoring (Advanced Edition) Version 4.1 .....	29
	Upgrading the Server/Gateway Components.....	30
	Installing a Default Resource Model .....	31

---

Adding the Notice Group . . . . .	31
Installing Java Runtime Environment . . . . .	32
Installation Options . . . . .	32
Enabling Tivoli Monitoring to Work with Tivoli Enterprise Data Warehouse . . . . .	34
Uninstalling the Product . . . . .	34
Uninstalling the Endpoint Components . . . . .	34
Uninstalling from Servers and Gateways . . . . .	34

**Chapter 3. Using the Product . . . . . 37**

Overview . . . . .	38
Profiles . . . . .	38
Key Points About Profiles . . . . .	38
Setting up Managed Resources . . . . .	38
Creating Profile Managers and Profiles . . . . .	39
Using the CLI to Manage Profiles at the Server . . . . .	40
Default Resource Models . . . . .	41
Adding a Default Resource Model to a Profile . . . . .	41
Using the CLI to Add a Default Resource Model to a Profile . . . . .	43
Adding a Customized Resource Model to a Profile . . . . .	43
Using the CLI to Add a Customized Resource Model to a Profile . . . . .	44
Customizing a Resource Model . . . . .	44
Customizing Thresholds and Event Rules . . . . .	44
Specifying Built-in Actions for an Event (Windows) . . . . .	47
Editing a Built-In Action . . . . .	47
Removing a Built-In Action . . . . .	48
Adding a Built-In Action . . . . .	48
Specifying Tasks for an Event . . . . .	48
Sending an e-mail in response to an event . . . . .	49
Sending a notice in response to an event . . . . .	50
Customizing Parameters . . . . .	51
Customizing Scheduling Information . . . . .	52
Setting the Monitoring Period . . . . .	53
Creating Schedule Rules . . . . .	53
Customizing Data Logging Information . . . . .	53
Using the CLI to Customize a Resource Model for a Profile . . . . .	55
Setting Distribution Options for a Profile . . . . .	55
Profile Distribution using MDist2 . . . . .	56
Rerunning the Failed Distributions . . . . .	57

Managing Profiles and Resource Models at Endpoints .....	58
Managing Tivoli Monitoring at Gateways .....	58
<b>Chapter 4. Heartbeat Function.....</b>	<b>59</b>
Heartbeat Statuses .....	59
Installing and Configuring the Heartbeat .....	59
Controlling the Heartbeat.....	60
Heartbeat Data Flows .....	61
Endpoint Registration .....	61
Heartbeat Monitoring.....	63
Viewing the Endpoint Cache .....	64
<b>Chapter 5. Integration with Tivoli Enterprise Console Server .....</b>	<b>67</b>
Overview .....	67
Identifying the Tivoli Enterprise Console Server to Tivoli Monitoring .....	67
Enabling Monitoring of Events and Heartbeat Messages on the Tivoli Enterprise Console Server ...	68
Enable Monitoring of the Events on the Availability Intermediate Manager.....	70
Slots of the Tivoli Enterprise Console Classes .....	70
Slots in the TMW_Event Class .....	70
Slots in the HeartBeat_Event Class .....	71
<b>Chapter 6. Integration with Tivoli Business Systems Manager.....</b>	<b>73</b>
Overview .....	73
Installing Tivoli Business Systems Manager Adapter .....	73
Configuring Tivoli Business Systems Manager Adapter .....	74
Using the Tivoli Business Systems Manager Adapter .....	76
Viewing Tivoli Business Systems Manager Data .....	77
Enabling and Disabling Tivoli Business Systems Manager Integration .....	78
Managing the Tivoli Business Systems Adapter from the CLI .....	79
<hr/>	
<b>Part II. Web Health Console.....</b>	<b>81</b>
<b>Chapter 7. Introduction to the Web Health Console.....</b>	<b>83</b>
Overview .....	83
Understanding Resource Health .....	83
Connecting the Web Health Console .....	84
The Request Manager Processor .....	84
Software Requirements .....	85
Installing the Web Health Console .....	85



---

	Installing on Windows . . . . .	85
	Installing on UNIX . . . . .	86
	Troubleshooting the Installation . . . . .	87
	Enabling Secure Socket Layer Support. . . . .	90
	<b>Chapter 8. Using the Web Health Console . . . . .</b>	<b>93</b>
	Logging onto the Web Health Console . . . . .	93
	Logging on to the Web Health Console for the first time . . . . .	94
	Setting Your Preferences . . . . .	94
	Managing Endpoint Lists. . . . .	94
	Managing General Preferences. . . . .	96
	Managing Chart Preferences . . . . .	96
	Common Web Health Console Features . . . . .	97
	Using the Endpoint List View . . . . .	98
	Working with the Endpoints . . . . .	99
	Using the Resource Model List View. . . . .	99
	Using the Endpoints by Resource Model View. . . . .	100
	Working with the Endpoints. . . . .	101
	Working with the Endpoint Health View . . . . .	101
	Working with the Resource Models Frame. . . . .	102
	Working with the Resource Models . . . . .	102
	Indications Frame . . . . .	103
	Historical Data Frame . . . . .	103
	Working with the Chart Views . . . . .	104
	Online Data Graph . . . . .	104
	Historical Data Graph . . . . .	105
	<b>Appendix A. Commands . . . . .</b>	<b>107</b>
	Upgraded Command Names. . . . .	107
	Commands for Managing Profiles at the Tivoli Server . . . . .	108
	Command for Managing Default Resource Models at the Tivoli Server . . . . .	108
	Command for Managing the Tivoli Business Systems Manager Adapter . . . . .	108
	Commands for Managing Tivoli Monitoring at the Endpoints . . . . .	108
	Commands for Managing Tivoli Monitoring at Managed Nodes/Gateways . . . . .	109
	wdmcmd . . . . .	110
	wdmconfig . . . . .	111
	wdmdiscovery . . . . .	114
	wdmdistrib . . . . .	116
	wdmdumpprf . . . . .	118

---

---

wdmeditprf . . . . .	122
wdmeng . . . . .	129
wdmheartbeat . . . . .	131
wdmloadprf . . . . .	133
wdmlseng . . . . .	135
wdmmn . . . . .	139
wdmmngcache . . . . .	141
wdmrm . . . . .	144
wdmtrceng . . . . .	146
wtmntaddrm . . . . .	148
wtmntdefrm . . . . .	149
wtmntrmrm . . . . .	150

**Appendix B. Error, Warning, and Information Messages . . . . . 151**

Identifying a Message . . . . .	151
Notation . . . . .	151
Messages . . . . .	152

**Appendix C. Troubleshooting . . . . . 179**

Logs and Traces . . . . .	179
I The Log Record Format . . . . .	179
I Trace Record Format . . . . .	180
I Tool to Create XML File . . . . .	181
Server Traces . . . . .	181
Distribution Logs . . . . .	181
Tivoli Management Framework Logs . . . . .	181
I Profile Core Trace . . . . .	181
Gateway Traces . . . . .	182
Heartbeat Engine Traces . . . . .	182
Task Engine Trace . . . . .	183
Tivoli Business Systems Manager Engine Traces . . . . .	183
Endpoint Upcall Traces . . . . .	184
Tivoli Business Systems Manager Adapter Trace . . . . .	185
Tivoli Business Systems Manager Transport Trace . . . . .	185
I Request Manager Trace . . . . .	185
Endpoint Logs and Traces . . . . .	186
Windows Endpoint Logs . . . . .	186
Endpoint Engine Log . . . . .	186
WMI Log . . . . .	186

---

	Profile Distribution Endpoint Logs . . . . .	187
	Endpoint Engine Log . . . . .	187
	Non-Windows Endpoints Logs . . . . .	187
	Endpoint Engine Update Log . . . . .	187
	Endpoint Engine Log and Trace . . . . .	187
	Endpoint Native Trace . . . . .	188
	Endpoint JMX Log . . . . .	188
	Web Health Console Logs and Traces . . . . .	189
	Modifying Web Health Console Tracing Parameters . . . . .	189
	Autotrace . . . . .	190
	Installing the Autotrace trace collector . . . . .	190
	Configuring Autotrace . . . . .	190
	Product File . . . . .	191
	Config File . . . . .	191
	Controlling Autotrace . . . . .	192
	Syntax . . . . .	192
	Description . . . . .	192
	Options . . . . .	193
	Endpoint Tools (Windows) . . . . .	193
	Problem Determination . . . . .	193
	Web Health Console Troubleshooting . . . . .	196
	Tivoli Monitoring 5.1 Internal Processes . . . . .	196
	WebSphere Application Server and HTTPServer . . . . .	196
	Unable to Connect to the Main Web Health Console Web Page . . . . .	197
	Slow or Unresponsive Web Health Console Screens or Panels . . . . .	197
	Unable to Perform any Operations at Some Random Point during a Web Health Console Session . . . . .	197
	Error Messages Displaying Graphs . . . . .	197
	Other Possible Problems . . . . .	198
	Starting and Stopping WebSphere Application . . . . .	199
	Starting and Stopping HTTPServer . . . . .	199
	Serviceability Tasks . . . . .	199
	<b>Appendix D. Accessibility . . . . .</b>	<b>203</b>
	Using Assistive Technologies . . . . .	203
	Magnifying What is Displayed on the Screen . . . . .	203
	Documentation in Accessible Formats . . . . .	204
	Using Alternative Text . . . . .	204
	<b>Appendix E. Use of Rhino: JavaScript for Java . . . . .</b>	<b>205</b>

---

---

The Mozilla Public License Version 1.1 .....	205
The Netscape Public License Version 1.1 .....	211
<b>Appendix F. Environment Variables .....</b>	<b>213</b>
<b>Appendix G. Migration Considerations .....</b>	<b>215</b>
Migrating from Tivoli Distributed Monitoring (Classic Edition) to Tivoli Monitoring .....	215
Overview .....	215
Guidelines for Migrating to Tivoli Monitoring 5.1 .....	217
Design Time .....	217
Deployment Time .....	218
Run Time .....	219
Migration Process .....	219
Sentry Profile Analyzer .....	219
Examples .....	223
Compatibility Mode .....	224
Wizard Process .....	226
How Everything Fits Together .....	226
Migrating from Tivoli Web Component Manager .....	228
Tims2XML .....	229
Installation and Customization .....	229
Structure of the XML file .....	230
System Monitoring Tasks .....	232
Process Monitor Task .....	232
CPU Monitor Task .....	233
UNIX File System Monitor Task .....	234
Resource Models .....	236
Logical Disk Resource Model .....	236
Processor Resource Model .....	237
Memory Resource Model .....	238
File System Resource Model .....	238
Migration Sample: Memory Monitor Task .....	238
<b>Glossary .....</b>	<b>241</b>
<b>Index .....</b>	<b>245</b>

---

## Figures

1. Graph showing CPU usage generating indications, holes and an event . . . . .	9
2. Components and event monitors of IBM Tivoli Monitoring. . . . .	14
3. Data flow for profile distribution and resource model monitoring activities . . . . .	16
4. Data flow for heartbeat control. . . . .	61
5. Data flow for endpoint registration . . . . .	62
6. Data flow for heartbeat function at gateway . . . . .	63
7. Data flow using wdmnmgcache command . . . . .	64
8. Data flows for bulk/delta discovery . . . . .	76
9. Data flow for viewing Tivoli Monitoring data in Tivoli Business Systems Manager. . . . .	77
10. The request manager processor. . . . .	85



---

## Tables

1. Example threshold defaults. . . . .	7
2. Counting occurrences. . . . .	9
3. High CPU Usage percentages over 12 cycles: holes and occurrences . . . . .	9
4. High CPU Usage percentages over 12 cycles: health. . . . .	11
5. Supported platforms . . . . .	19
6. Hardware Prerequisites for Tivoli Monitoring . . . . .	20
7. Hardware Prerequisites for Tivoli Business Systems Manager Adapter. . . . .	20
8. Prerequisite Software . . . . .	21
9. Slots of the TMW_Event Class . . . . .	70
10. Slots of the HeartBeat_Event Class . . . . .	71
11. Health Determination Example. . . . .	84
12. Navigation Icons . . . . .	98
13. Health alert icons. . . . .	98
14. New and old command names . . . . .	107
15. Problems and Their Workarounds . . . . .	194
16. Environment variables set by triggering events.. . . . .	213
17. More environment variables.. . . . .	213





---

## Preface

IBM® Tivoli® Monitoring allows you to monitor availability and performance status of *resources* on your systems to identify bottlenecks and potential resource problem. This guide describes how to install, customize, and use IBM Tivoli Monitoring.

IBM Tivoli Monitoring was formerly known as Tivoli Distributed Monitoring (Advanced Edition). You should note that a separate product exists, called Tivoli Distributed Monitoring (Classic Edition), formerly known as Tivoli Distributed Monitoring. Descriptions of the IBM Tivoli Monitoring and Tivoli Distributed Monitoring (Classic Edition) products are given in the glossary on page 241.

## Who Should Read This Guide

The target audiences for this guide are system administrators and system installers.

To make effective use of the product you require knowledge as well as practical experience of the following:

- Importing, installing, and managing the Tivoli Management Framework and the Tivoli Management Environment®
- System administration on the systems where you plan to install IBM Tivoli Monitoring components and which you plan to monitor using the product

You should also be familiar with the Tivoli Enterprise Console® product.

## What is New in this Release

IBM Tivoli Monitoring Version 5.1 is a development of, and upgrades, Tivoli Distributed Monitoring (Advanced Edition) Version 4.1, and provides the following additional functionality:

### **Mdist2 Support**

The product now uses Mdist2 to distribute profiles to the endpoints (see Chapter 3, “Using the Product” on page 37).

### **Web-based Health Console**

The former Health Console has become a Web-based Health Console (see Chapter 7, “Introduction to the Web Health Console” on page 83 and Chapter 8, “Using the Web Health Console” on page 93).

### **Tools and guidelines for migrating from Tivoli Distributed Monitoring (Classic Edition)**

A set of guidelines is available for the migration, and a compatibility mode is now provided to enable Tivoli Monitoring 5.1 users to use Tivoli Distributed Monitoring (Classic Edition) monitors inside a Tivoli Monitoring 5.1 resource model (see Appendix G, “Migration Considerations” on page 215).

### **Tivoli Enterprise Data Warehouse Support**

The product now uses the Gathering Historical Data component to store data in the Tivoli Enterprise Data Warehouse schema (see “Enabling Tivoli Monitoring to Work with Tivoli Enterprise Data Warehouse” on page 34).

### **Additional Response Actions**

The product is now able to act on an event by sending an e-mail to a specified address, sending a notice to a notice group, running a program (see Chapter 3, “Using the Product” on page 37).

### **Tools and guidelines for migrating from Tivoli Web Component Manager**

A tool and a set of resource models are now available to enable Tivoli Web Component Manager users to migrate to Tivoli Monitoring (see Appendix G, “Migration Considerations” on page 215)

### **Serviceability Features**

Several new serviceability features have been added to the product. For details, see Appendix C, “Troubleshooting” on page 179.

## **Changes to This Edition of the User’s Guide**

This edition is an updated version of the previous version of this guide. All technical changes and all new information are marked by change bars in the left margin.

The following information has been added:

- Part II, “Web Health Console” on page 81 has been added to describe the new web-based Health Console.
- Appendix F, “Environment Variables” has been added to document the environment variables used by Tivoli Monitoring.
- Appendix G, “Migration Considerations” on page 215 has been added to describe the migration from Tivoli Distributed Monitoring (Advanced Edition) 4.1 and from Tivoli Web Component Manager.

The following information has been changed:

- Chapter 1, “Introduction” has been updated to reflect the new functions of the product.
- Chapter 2, “Installing” has been updated to reflect new software pre-requisites and installation features.
- Chapter 3, “Using the Product” has been updated with new screen captures and information about the new functionality.
- Chapter 4, “Heartbeat Function” has been updated to include a new configuration key for the **wdmconfig** command.
- Appendix A, “Commands” has been updated to include new/changed parameters.
- Appendix C, “Troubleshooting” on page 179 has been updated to describe serviceability aspects.

## **What This Guide Contains**

This guide contains the following sections and chapters:

### **Part I. Installing, Configuring, and Using**

- Chapter 1, “Introduction”  
Provides an introduction to IBM Tivoli Monitoring, its features and functions and its role in the Tivoli environment. It also provides a brief introduction to the IBM Tivoli Monitoring Workbench, an application you can use to develop your own resource models, or to extend existing ones.

- Chapter 2, “Installing”  
Describes the steps for installing IBM Tivoli Monitoring in your Tivoli environment.
- Chapter 3, “Using the Product”  
Describes the purpose of profiles in the Tivoli environment and presents the range of customization options you can apply to the default resource models supplied with the product or generated using IBM Tivoli Monitoring Workbench. It also describes how to distribute resource models to endpoints, and what to do if the distribution fails at one or more endpoints.
- Chapter 4, “Heartbeat Function”  
Gives full details of the purpose, use and management of the heartbeat function.
- Chapter 5, “Integration with Tivoli Enterprise Console Server”  
Describes how to enable monitoring of IBM Tivoli Monitoring events on a Tivoli Enterprise Console server (event server).
- Chapter 6, “Integration with Tivoli Business Systems Manager”  
Describes the integration of IBM Tivoli Monitoring with Tivoli Business Systems Manager and provides the steps for enabling the monitoring of IBM Tivoli Monitoring events on the Tivoli Business Systems Manager workstation.

## **Part II. Web Health Console**

- Chapter 7, “Introduction to the Web Health Console”  
Describes the purpose and benefits of the Web Health Console in day-to-day monitoring of resources.
- Chapter 8, “Using the Web Health Console”  
Describes the steps for starting, configuring, and using the Web Health Console to monitor the resources of a specific list of endpoints.

## **Part III: Appendixes**

- Appendix A, “Commands”  
Lists and describes the IBM Tivoli Monitoring commands that you can issue from the command line.
- Appendix B, “Error, Warning, and Information Messages”  
Lists the error, warning, and information messages that can appear in IBM Tivoli Monitoring. It provides a description of the message, what caused the error, and how to resolve the cause of the error.
- Appendix C, “Troubleshooting”  
Provides information about logs and traces, and some diagnostics.
- Appendix D, “Accessibility”  
Provides information about the support provided in the product and its documentation for users with a visual or physical handicap.
- Appendix E, “Use of Rhino: JavaScript for Java”  
Provides the text of the license for the use of Rhino: JavaScript™ for Java™, which is incorporated in the product.
- Appendix F, “Environment Variables”  
Documents the environment variables used by the product.

- Appendix G, “Migration Considerations”  
Describes the migration from Tivoli Distributed Monitoring (Advanced Edition) 4.1 and the migration from Tivoli Web Component Manager.

## Publications

This section lists publications in the IBM Tivoli Monitoring library and any other related documents. It also describes how to access Tivoli publications online, how to order Tivoli publications, and how to make comments on Tivoli publications.

### IBM Tivoli Monitoring Library

The following documents are available in the IBM Tivoli Monitoring library:

- *IBM Tivoli Monitoring: User's Guide*, SH19-4569  
Describes how to install, customize, and use IBM Tivoli Monitoring to manage system and application resources.
- *IBM Tivoli Monitoring: Workbench User's Guide*, SH19-4571  
Describes how to use IBM Tivoli Monitoring Workbench to create new resource models or to modify existing ones.
- *IBM Tivoli Monitoring: Resource Model Reference*, SH19-4570  
Provides information about using and customizing the resource models that can be used with IBM Tivoli Monitoring.
- *IBM Tivoli Monitoring: Release Notes*, GI10-5793  
Provides the most current information about IBM Tivoli Monitoring.
- *IBM Tivoli Monitoring: Deployment Supplement*, SC23-4799  
Describes how to use the deployment wizard for a custom installation.
- *IBM Tivoli Monitoring: Quick Deployment Road Map*, GI11-0938  
Describes typical deployment scenarios.
- *IBM Tivoli Monitoring: Deployment Supplement*, SC23-4799  
Describes how to use the deployment wizard for a custom installation.

Versions of these documents in PDF and HTML formats can be found on the IBM Tivoli Monitoring and IBM Tivoli Monitoring Workbench product CDs. They are stored in the Books directory, and can be accessed by selecting the file `Books/infocenter.html` with your Web browser. This displays an HTML page from which all of the documents can be accessed in either format.

Updated versions of these documents might be placed from time-to-time on the Tivoli Customer Support Web site (see “Accessing Publications Online” on page xix, for more details).

### Prerequisite Publications

Before beginning the installation, read the following prerequisite and related documentation for more information about the management options that the IBM Tivoli Monitoring products provide:

- *Tivoli Distributed Monitoring (Classic Edition): Release Notes*, GI10-3023  
Provides updated information about the Tivoli Distributed Monitoring (Classic Edition) product (formerly known as Tivoli Distributed Monitoring).

- *Tivoli Decision Support for Server Performance Prediction: Release Notes*, GI11-0859  
Provides updated information about the integration of Tivoli Decision Support for Server Performance Prediction with Tivoli Distributed Monitoring (Advanced Edition).
- *Tivoli Management Framework: User's Guide*  
Provide prerequisite information about setting up and using the Tivoli Management Framework and Tivoli environment.
- *Tivoli Management Framework: Installation Guide*  
Provide prerequisite information about setting up and using the Tivoli Management Framework and Tivoli environment.
- *Tivoli Management Framework: Planning for Deployment Guide*  
Provide prerequisite information about setting up and using the Tivoli Management Framework and Tivoli environment.
- *Tivoli Management Framework: Reference Manual*  
Provide prerequisite information about setting up and using the Tivoli Management Framework and Tivoli environment.
- *Tivoli Software Installation Service: User's Guide*  
Provides task-oriented information on how to import, manage, and install Tivoli Management Environment software on selected machines and managed nodes within your Tivoli management region.
- *Tivoli Software Installation Service: Release Notes*  
Provides important information about using and installing the Tivoli Software Installation Service (SIS)
- *Tivoli Enterprise Console: Rule Builder's Guide*  
Provides information about using the Tivoli Enterprise Console rule editor and graphical rule builder to modify existing rules and create new rules to match your specific event management needs.
- *Tivoli Enterprise Console: Release Notes*, Version 3.7 or later  
Provides the most current information about Tivoli Enterprise Console.
- *Tivoli Business Systems Manager: User's Guide* (if you intend to use Tivoli Business Systems Manager to monitor Tivoli Monitoring events)  
Describes how to use the Tivoli Business Systems Manager product.
- *Tivoli Business Systems Manager: Console User's Guide* (if you intend to use Tivoli Business Systems Manager to monitor Tivoli Monitoring events)  
Describes how to use the Tivoli Business Systems Manager console.

## Accessing Publications Online

You can access many Tivoli publications online at the Tivoli Customer Support Web site:  
<http://www.tivoli.com/support/documents/>

These publications are available in PDF or HTML format, or both. Translated documents are also available for some products.

## Ordering Publications

You can order many Tivoli publications online at the following Web site:  
<http://www.elink.ibm.com/public/applications/publications/cgibin/pbi.cgi>

You can also order by telephone by calling one of these numbers:

- In the United States: 800-879-2755
- In Canada: 800-426-4968
- In other countries, for a list of telephone numbers, see the following Web site:  
[http://www.tivoli.com/inside/store/lit\\_order.html](http://www.tivoli.com/inside/store/lit_order.html)

## Providing Feedback about Publications

We are very interested in hearing about your experience with Tivoli products and documentation, and we welcome your suggestions for improvements. If you have comments or suggestions about our products and documentation, contact us in one of the following ways:

- Send an e-mail to [pubs@tivoli.com](mailto:pubs@tivoli.com).
- Complete our customer feedback survey at the following Web site:  
<http://www.tivoli.com/support/survey/>

## Contacting Customer Support

If you have a problem with any Tivoli product, you can contact Tivoli Customer Support. See the *Tivoli Customer Support Handbook* at the following Web site:  
<http://www.tivoli.com/support/handbook>

The handbook provides information about how to contact Tivoli Customer Support, depending on the severity of your problem, and the following information:

- Registration and eligibility
- Telephone numbers and e-mail addresses, depending on the country you are in
- What information you should gather before contacting support

## Conventions Used in This Guide

This book uses several conventions for special terms and actions, operating system-dependent commands and paths, and margin graphics.

### Typeface Conventions

The following typeface conventions are used in this book:

- |                  |   |
|------------------|---|
| <b>Bold</b>      | Lowercase and mixed-case commands, command options, and flags that appear within text appear like <b>this</b> , in <b>bold</b> type.<br>Graphical user interface elements (except for titles of windows and dialogs) and names of keys also appear like <b>this</b> , in <b>bold</b> type.  |
| <i>Italic</i>    | Variables, values you must provide, new terms, and words and phrases that are emphasized appear like <i>this</i> , in <i>italic</i> type.   |
| <b>Monospace</b> | Commands, command options, and flags that appear on a separate line, code examples, output, and message text appear like <code>this</code> , in monospace type.<br>Names of files and directories, text strings you must type, when they appear within text, names of Java methods and classes, and HTML and XML tags also appear like <code>this</code> , in monospace type. |

## Operating System-dependent Variables and Paths

This book uses the UNIX<sup>®</sup> convention for specifying environment variables and for directory notation.

When using the Windows<sup>®</sup> command line, replace *\$variable* with *%variable%* for environment variables and replace each forward slash (/) with a backslash (\) in directory paths.

**Note:** If you are using the bash shell on a Windows system, you can use the UNIX conventions.





---

# Part I. Installing, Configuring, and Using

<b>Chapter 1. Introduction</b> .....	<b>5</b>
Overview .....	5
Main Features .....	6
Understanding Tivoli Monitoring.....	6
Resources .....	6
Resource Models.....	6
Cycles.....	7
Thresholds .....	7
Parameters .....	7
Indications .....	7
Occurrences and Holes .....	8
Events.....	8
Clearing Events.....	10
Monitoring of Events and Indications.....	10
Tivoli Enterprise Console Server .....	10
Web Health Console .....	11
Tivoli Business Systems Manager .....	11
Recovery Actions .....	11
Built-In Actions.....	11
Tivoli Framework Tasks .....	12
Scheduling .....	12
Logging .....	12
Profiles .....	12
The Heartbeat Function .....	12
Gathering Historical Data Component .....	12
Components .....	13
Tivoli Monitoring Base Component .....	14
Web Health Console .....	15
Tivoli Business Systems Manager Adapter.....	15
Endpoint Component.....	15
Collecting the Data .....	15
Resource Model Data Flow .....	15
Obtaining Resource Data at the Endpoint.....	16
Security Considerations .....	17
Tivoli Monitoring Workbench .....	17

---

<b>Chapter 2. Installing</b> .....	<b>19</b>
Supported Platforms .....	19
Hardware Requirements .....	20
Prerequisite Software .....	20
Tivoli Management Framework .....	21
Tivoli Management Framework Patches Required for Version 3.7B .....	21
Other Software .....	21
Coexistence with Other Versions of Tivoli Monitoring .....	22
Products within the Tivoli Monitoring Family .....	22
Products within the Tivoli Distributed Monitoring (Classic Edition) Family .....	23
Backward Compatibility .....	23
Installing/Upgrading the Product .....	23
Before You Begin .....	24
Installing with Tivoli Software Installation Service .....	24
Installing from the Tivoli Desktop .....	25
Installing from the Command Line .....	28
Upgrading from Tivoli Distributed Monitoring (Advanced Edition) Version 4.1 .....	29
Upgrading the Server/Gateway Components .....	30
Installing a Default Resource Model .....	31
Adding the Notice Group .....	31
Installing Java Runtime Environment .....	32
Installation Options .....	32
Enabling Tivoli Monitoring to Work with Tivoli Enterprise Data Warehouse .....	34
Uninstalling the Product .....	34
Uninstalling the Endpoint Components .....	34
Uninstalling from Servers and Gateways .....	34
 <b>Chapter 3. Using the Product</b> .....	 <b>37</b>
Overview .....	38
Profiles .....	38
Key Points About Profiles .....	38
Setting up Managed Resources .....	38
Creating Profile Managers and Profiles .....	39
Using the CLI to Manage Profiles at the Server .....	40
Default Resource Models .....	41
Adding a Default Resource Model to a Profile .....	41
Using the CLI to Add a Default Resource Model to a Profile .....	43
Adding a Customized Resource Model to a Profile .....	43

---

Using the CLI to Add a Customized Resource Model to a Profile . . . . .	44
Customizing a Resource Model . . . . .	44
Customizing Thresholds and Event Rules. . . . .	44
Specifying Built-in Actions for an Event (Windows) . . . . .	47
Editing a Built-In Action . . . . .	47
Removing a Built-In Action. . . . .	48
Adding a Built-In Action. . . . .	48
Specifying Tasks for an Event . . . . .	48
Sending an e-mail in response to an event . . . . .	49
Sending a notice in response to an event . . . . .	50
Customizing Parameters. . . . .	51
Customizing Scheduling Information . . . . .	52
Setting the Monitoring Period . . . . .	53
Creating Schedule Rules . . . . .	53
Customizing Data Logging Information . . . . .	53
Using the CLI to Customize a Resource Model for a Profile . . . . .	55
Setting Distribution Options for a Profile. . . . .	55
Profile Distribution using MDist2 . . . . .	56
Rerunning the Failed Distributions. . . . .	57
Managing Profiles and Resource Models at Endpoints . . . . .	58
Managing Tivoli Monitoring at Gateways . . . . .	58
<b>Chapter 4. Heartbeat Function. . . . .</b>	<b>59</b>
Heartbeat Statuses . . . . .	59
Installing and Configuring the Heartbeat . . . . .	59
Controlling the Heartbeat. . . . .	60
Heartbeat Data Flows . . . . .	61
Endpoint Registration . . . . .	61
Heartbeat Monitoring. . . . .	63
Viewing the Endpoint Cache . . . . .	64
<b>Chapter 5. Integration with Tivoli Enterprise Console Server . . . . .</b>	<b>67</b>
Overview . . . . .	67
Identifying the Tivoli Enterprise Console Server to Tivoli Monitoring . . . . .	67
Enabling Monitoring of Events and Heartbeat Messages on the Tivoli Enterprise Console Server . . .	68
Enable Monitoring of the Events on the Availability Intermediate Manager. . . . .	70
Slots of the Tivoli Enterprise Console Classes . . . . .	70
Slots in the TMW_Event Class . . . . .	70
Slots in the HeartBeat_Event Class . . . . .	71

---

## **Chapter 6. Integration with Tivoli Business Systems Manager..... 73**

Overview .....	73
Installing Tivoli Business Systems Manager Adapter .....	73
Configuring Tivoli Business Systems Manager Adapter .....	74
Using the Tivoli Business Systems Manager Adapter .....	76
Viewing Tivoli Business Systems Manager Data .....	77
Enabling and Disabling Tivoli Business Systems Manager Integration .....	78
Managing the Tivoli Business Systems Adapter from the CLI .....	79

# 1

## Introduction

---

This chapter describes the main features of IBM Tivoli Monitoring 5.1 (hereafter also referred to as Tivoli Monitoring) and its role in the Tivoli environment. It is divided into these main sections:

- “Overview” gives a general overview of the product.
- “Understanding Tivoli Monitoring” explains the concepts behind the product.
- “Components” details the components that can be installed, and explains the relationships between them.
- “Collecting the Data” describes the data flows for the various functions of the product.
- “Security Considerations” discusses the implementation of the product across firewalls.
- “Tivoli Monitoring Workbench” describes how this separate component interfaces with the product.

### Overview

IBM® Tivoli® Monitoring 5.1 is a Tivoli application that applies preconfigured, automated best practices to the automated monitoring of essential system resources. The application detects bottlenecks and other potential problems and provides for the automatic recovery from critical situations, which eliminates the need for system administrators to manually scan through extensive performance data. The application also integrates seamlessly with other Tivoli Availability solutions, including Tivoli Business Systems Manager® and Tivoli Enterprise Console®. This application was previously called Tivoli Distributed Monitoring (Advanced Edition).

Tivoli Monitoring is not a programming tool; most features can be used as supplied or can be modified manually using the graphical user interfaces (GUIs) provided.

## Main Features

The main features of Tivoli Monitoring are as follows:

- An *off-the-shelf* solution for monitoring Windows<sup>®</sup>, UNIX<sup>®</sup> and Linux systems.
- Ready-to-use resource models that report on specific aspects of a system's status. For example, the Process resource model provides information on the status of processes, CPU usage, and so on. The resource monitoring is an implementation of the Common Information Model (CIM). CIM is an approach to system and network management that applies object-oriented techniques to model the system.
- Data collection and problem analysis is performed locally on the system.
- Resource models that can easily be added (point and click) to a Tivoli profile, which can be distributed to multiple systems simultaneously.
- The ability to view both real-time and historical data for any system from a centralized monitoring application called the *Web Health Console*, which is supplied with the product. Only the results of the data collection and problem analysis are retrieved by the Web Health Console.
- Options are also available to send the results from the data collection and analysis to the Tivoli Enterprise Console or to the Tivoli Business Systems Manager.
- The ability to specify automatic corrective or preventative actions to resolve situations that could develop into real problems.
- The ability to modify resource models, for example, by changing threshold levels to match users' own requirements.
- A scheduling feature that allows monitoring to take place at user-specified times.
- A heartbeat function running at gateways, that regularly checks the availability and status of attached endpoints and makes the information available to the Tivoli Enterprise Console server, Tivoli Business Systems Manager or Tivoli Monitoring Notice Group.

## Understanding Tivoli Monitoring

Although Tivoli Monitoring provides access to large amounts of performance data, the concepts behind the product are relatively straightforward. This section describes the main concepts required for an understanding of how to use the product.

### Resources

Tivoli Monitoring monitors *resources* at distributed systems. In this context, a resource is anything that affects the operation of a computer system and includes physical and logical disks, CPUs, memory, printers, as well as the processes running, and the services, such as LanMan, the Windows event log, the UNIX syslogd (logging system daemon) and TCP/IP.

### Resource Models

Tivoli Monitoring uses out-of-the box, predefined *resource models* to specify which resource data are accessed from the system at runtime and how these data are processed. For example, the Process resource model obtains data related to processes running on the system. Performance data are automatically collected by the resource model and processed by an appropriate algorithm to determine whether or not the system is performing to your expectations. Generally you can use the default values and still obtain useful data. However, if necessary you can customize the resource models to suit your local requirements.

Full details of the resource models supplied with the product are given in the *IBM Tivoli Monitoring: Resource Model Reference*.

## Cycles

When a resource model is run at an endpoint, it gathers data at regular intervals, known as *cycles*; the duration of a cycle is the *cycle time*. A resource model with a cycle time of 60 seconds gathers information every 60 seconds. The data collected are a snapshot of the status of the resources specified in the resource model. Each of the supplied resource models has a default cycle time, which you can modify as required.

## Thresholds

Each resource model defines one or more *thresholds*. A threshold is a named property of the resource with a default value which you can modify in the customization phase. Typically, the value specified for a threshold represents a significant reference level of a performance-related entity, which, if exceeded or not reached, a system administrator might want to know about. However, some thresholds are used as reference values to limit the scope of the resource model. For example, in the Windows Process resource model, the following are amongst the thresholds used:

*Table 1. Example threshold defaults*

Threshold	Default Value
High CPU Usage	60
Maximum Processes	5

The threshold High CPU Usage is used to monitor excessive CPU usage, that perhaps might be damaging to other processes that might be running at the same time. The value for this threshold is measured as a percentage of the CPU capacity, the default being 60%. On the other hand, the Maximum Processes threshold determines that only the top 5 (the default value) highest CPU-using processes will be considered in the monitoring of high CPU usage.

Alternatively, a threshold can be a minimum below which a certain performance characteristic should not fall. For example, the Windows Logical Disk resource model has a Low Disk Space threshold showing the minimum percentage disk space that must always be available (the default is 5%).

## Parameters

Some resource models have one or more parameters. Each parameter can take the form of a list of strings, a list of numeric values, a Boolean list of predetermined values from which you can make any combination of selections, or a choice list of mutually exclusive alternatives. For example, the Windows Parametric TCP/IP Ports resource model has a parameter where you list the ports to be monitored and another to choose the port states to monitor.

## Indications

Each resource model will generate an *indication* if certain conditions implied by the resource model's thresholds are not satisfied in a given cycle. Each resource model has its own algorithm to determine which combinations of thresholds should generate an indication. Indications might be generated in any one of the following circumstances:

- A single threshold is exceeded. For example, in the Windows Process resource model, the Process High CPU indication is generated when the High CPU Usage threshold is exceeded (for any process that has a non-zero process ID).

- A combination of two or more thresholds are exceeded. For example, in the Windows Logical Disk resource model a High Read Bytes per Second indication is generated when both the following thresholds are exceeded:
  - The amount of bytes transferred per second (being written or read) exceeds the High Bytes per Second threshold
  - The percent of time that the selected disk drive spends for read or write requests exceeds the High Percent Usage threshold.
- A combination of other factors has changed. For example, in the Windows Process resource model the Process Handle Leak indication is generated when a process is leaking handles. There is no threshold for this indication. The resource model compares the number of handles of the five processes with the most handles in consecutive cycles. If the number of handles has increased, the indication is generated.

The specific definition of the indications that have been created for each resource model are documented in *IBM Tivoli Monitoring: Resource Model Reference*.

### Occurrences and Holes

An *occurrence* is the term used to refer to a cycle during which an indication occurs for a given resource model.

A *hole* is the term used to refer to a cycle during which an indication does *not* occur for a given resource model. In other words, none of the conditions specified for the generation of any indication have been met. This does not mean that none of the thresholds have been exceeded. For example, in the Windows Logical Disk resource model a High Read Bytes per Second indication is *not* created when the percentage disk time is higher than the High Percent Usage threshold, provided that the Low Disk Space threshold is exceeded.

### Events

An *event* is used to verify the persistence of a given indication, by eliminating unrepresentative peaks and troughs for the indication. For example, a process that generates the Process High CPU indication in one cycle may be behaving perfectly normally, and be of no threat to other processes if the high usage is not repeated. However, an indication that persists over several cycles is more likely to be a problem. Thus, an event defines the number of consecutive occurrences of the indication that are significant.

However, given that you have decided that a certain number of consecutive cycles of Process High CPU greater than the chosen threshold value is significant, you may feel that if, during the accumulation of the consecutive occurrences, one or two cycles fall below the threshold, it should not stop the counting of consecutive occurrences. Thus, an event allows you to define how many consecutive *holes* in the sequence of consecutive occurrences are permitted.

So, an event is an aggregation of a defined number of consecutive occurrences during which there can be a defined number of consecutive holes. For example, consider an event that is defined as three occurrences and two holes. This means that one or two consecutive holes will be ignored by the aggregation algorithm in detecting three consecutive occurrences of the indication.

In the following two series of events, 1 represents an occurrence and 0 represents a hole.

- 0 1 0 1 0 0 1
- 0 0 1 0 0 1 1



Notice that, ignoring the sequences of one or two consecutive holes as stated in this example event definition, each series contains three consecutive occurrences. Therefore, in this example, each sequence represents an event.

Defining an event as three occurrences and two holes also means that the maximum number of cycles needed to trigger an event is seven cycles. In this case, each occurrence would be following by the maximum of two holes as shown below.

■ 1 0 0 1 0 0 1

Further, if the monitoring algorithm has counted two consecutive occurrences and then observes three consecutive holes – more holes than that allowed in the definition of an event for this example – it will set the count of occurrences to zero. See Table 2.

Table 2. Counting occurrences

Sequence	Count of Occurrences
1 0 0	1
1 0 0 1	2
1 0 0 1 0	2
1 0 0 1 0 0	2
1 0 0 1 0 0 0	0

Figure 1, shows a graph of the actual level of CPU usage on a hypothetical system, where the indication threshold is at 60% of CPU usage.

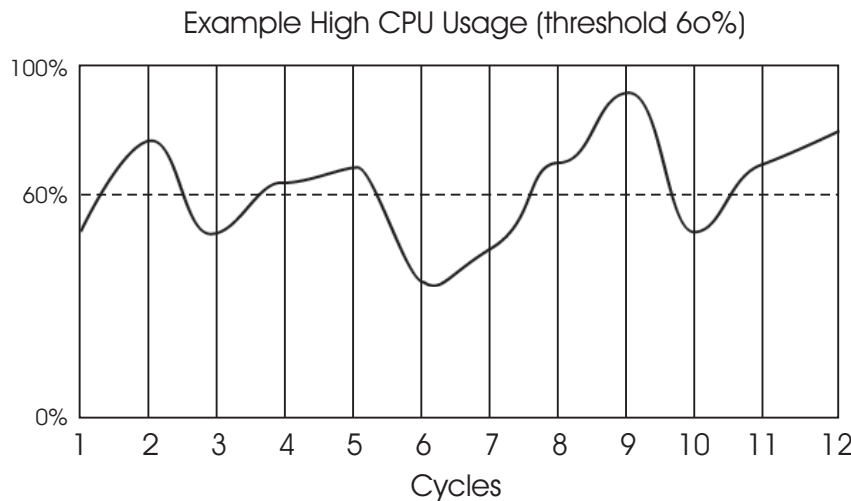


Figure 1. Graph showing CPU usage generating indications, holes and an event

The snapshots of the CPU usage obtained by the resource model are shown in Table 3. The first row shows the cycle number and the second the percentage of CPU usage.

Table 3. High CPU Usage percentages over 12 cycles: holes and occurrences

Cycle	1	2	3	4	5	6	7	8	9	10	11	12
CPU%	55	73	54	63	68	42	50	70	90	55	70	80
Occurrence or Hole	H	O	H	O	O	H	H	O	O	H	O	O

*Table 3. High CPU Usage percentages over 12 cycles: holes and occurrences (continued)*

<b>Occurrence Count</b>	0	<b>1</b>	1	<b>2</b>	<b>3</b>	3	0	<b>1</b>	<b>2</b>	2	<b>3</b>	<b>4</b>
-------------------------	---	----------	---	----------	----------	---	---	----------	----------	---	----------	----------

Cycles that exceed the threshold, and are thus occurrences, are shown in bold type; the other cycles are holes.

For this example, it has been determined that an event will be created for this indication if there are four consecutive occurrences with only one hole permitted. The final row in Table 3 on page 9 shows the count that the resource model would make of the consecutive occurrences. The count is set to zero at the 7th cycle snapshot because the resource model has encountered two consecutive holes. In this example, an event is triggered in the 12th cycle. The count is then set to zero and the process continues.

### Clearing Events

A clearing event is a resource model function that, if enabled, allows Tivoli Monitoring to close an error event when the circumstances that have caused the event are no longer present. Clearing events can be processed by the Tivoli Enterprise Console server and by Tivoli Business Systems Manager. Clearing events have a severity of *harmless*, regardless of the severity of the original event, but have the same event ID as the original event.

For example, a service stops and the Tivoli Enterprise Console server receives an event notifying it of this problem. Until the service restarts, the problem is still present on the endpoint, but Tivoli Monitoring does not send any further event notifications to the Tivoli Enterprise Console server. If the Clearing Event function is enabled, as soon as the service restarts, a clearing event is sent to the Tivoli Enterprise Console server, thereby closing the original event. The Clearing Event itself will not normally appear on the server, as its only function is to clear the original error event.

**Note:** The clearing event function is not always enabled. For example for the Parametric Event Log and the Event Log resource models the clearing event function is disabled.

**Note:** Correlated events cannot be cleared.

More information about the definition, enabling, and disabling of clearing events is given in the *IBM Tivoli Monitoring: User's Guide*.

### Monitoring of Events and Indications

Events can be sent to the Tivoli Enterprise Console server, the Web Health Console, and Tivoli Business Systems Manager.

#### Tivoli Enterprise Console Server

Events can be viewed by a Tivoli Enterprise Console server provided that you have compiled and loaded the relevant *Basic Recorder of Objects in C* (BAROC) files on the server. The event contains a set of properties that can help to identify the problem. For example, the information in the ProcessHandleLeak event includes values for the following:

- Current Process ID
- Number of handles allocated to the process
- Name of the process

Clearing events can also be processed by the Tivoli Enterprise Console server. If the default procedure is used to enable the monitoring of events (see “Enabling Monitoring of Events and Heartbeat Messages on the Tivoli Enterprise Console Server” on page 68), the Tivoli Enterprise Console server uses the clearing event to close the associated error event.

However, if you choose not to install the clearing events rule incorporated in the above-mentioned default procedure, the clearing event is displayed as a separate entity with the same id as the original error event.

### Web Health Console

The Web Health Console, which is an optional part of Tivoli Monitoring, obtains events and indications from endpoints. The Web Health Console displays the health of each potential problem as a numeric value between 100 (perfect health) and zero (with zero meaning that the conditions for the corresponding event have been met). Intermediate values show the percentage of occurrences currently registered with respect to the total number of occurrences needed to trigger an event. For example, Table 4 is based on Table 3 on page 9, but additionally shows the health percentage:

*Table 4. High CPU Usage percentages over 12 cycles: health*

Cycle	1	2	3	4	5	6	7	8	9	10	11	12
CPU%	55	73	54	63	68	42	50	70	90	55	70	80
Occurrence or Hole	H	O	H	O	O	H	H	O	O	H	O	O
Occurrence Count	0	1	1	2	3	3	0	1	2	2	3	4
Health (%)	100	75	75	50	25	25	100	75	50	50	25	0

In this example, the health percentage changes in steps of 25% because 4 occurrences were required to trigger an event; if the indication had required 5 occurrences, the health percentage would have changed by steps of 20%.

### Tivoli Business Systems Manager

Events can also be sent to the Tivoli Business Systems Manager, provided that the Tivoli Business Systems Manager Adapter component has been installed on the gateways of the endpoints that are to be monitored. A full description of Tivoli Business Systems Manager can be found in the Tivoli Business Systems Manager documentation. For more information about the integration of Tivoli Business Systems Manager with Tivoli Monitoring, see Chapter 6, “Integration with Tivoli Business Systems Manager” on page 73.

Clearing events can also be processed by the Tivoli Business Systems Manager, which uses the clearing event to close the associated error event.

## Recovery Actions

For any event recovery actions can be run automatically, both built-in actions (for Windows), or Tivoli Framework Tasks. The actions can take positive steps to remedy the situation, or can ensure that information about the event is distributed to the appropriate authorities or entities.

### Built-In Actions

Certain events may have one or more actions predefined for them. An action can be either the execution of a CIM class method, or the execution of a program. Both type of actions can be implemented only through the workbench. For example, an event that detects the failure of a service could have the restart of that service as its built-in action. Thus, without any human intervention, Tivoli Monitoring would detect the failure of a service and automatically restart it. Built-in actions are defined by default as part of an event, but can be removed. They have the same event ID as the event that they are designed to correct.

## Tivoli Framework Tasks

For each event you can select one or more Tivoli Framework Tasks to be performed when the event is triggered. See the *Tivoli Management Framework: User's Guide* and the *Tivoli Management Framework: Reference Manual* for more details about Tivoli Framework Tasks.

**Note:** The tasks that are triggered by a Tivoli Monitoring event can access the event name and event thresholds of the triggering event by accessing the environment variables. Refer to Appendix F, "Environment Variables" on page 213 for the list of environment variables used by the product.

## Scheduling

Tivoli Monitoring contains a scheduling feature that allows you to determine a period within which monitoring should take place, and specific scheduling rules.

The monitoring period is determined by defining a from and a to date.

The scheduling rules allow you to define time periods on specific weekdays during which monitoring will take place. Any number of rules can be defined, letting you set up a complex pattern of resource monitoring for a profile, covering the periods important to you.

The scheduled times are always interpreted as local times, allowing you to set up a single rule that will monitor the same local time period in different time zones. For example, if your region covers several time zones, but you wish to monitor morning activities in each time zone, a single rule defining the monitoring period as being between 08:00 and 13:00, is interpreted locally in each of the time zones, so that you monitor the same relative period.

You should note also that all times of events or activities reported from endpoints or gateways are also logged in the local time of the system from where they originated.

## Logging

For any endpoint, you can log the data collected by a resource model and write it in a local database. Then you can view it through the History View of the Web Health Console. You can choose to store raw or aggregated data.

For more information, see "Customizing Data Logging Information" on page 53.

## Profiles

Tivoli Monitoring is a profile-based application that runs in a Tivoli environment. Different profiles can be defined containing different selections of resource models. All aspects of existing profiles can be modified, including the addition, deletion and customization of resource models. You can distribute multiple profiles to each endpoint.

## The Heartbeat Function

In addition to the monitoring processes described above, Tivoli Monitoring operates a heartbeat function, which monitors the basic system status at endpoints attached to the gateway at which it is enabled. Events may be sent to the Tivoli Business Systems Manager (provided that the Tivoli Business Systems Manager Adapter component is installed at the gateway), the Tivoli Enterprise Console and the *IBM Tivoli Monitoring* Notice Group. Full details of this function are given in Chapter 4, "Heartbeat Function" on page 59.

## Gathering Historical Data Component

The Gathering Historical Data component enables Tivoli Monitoring to use Tivoli Decision Support for Server Performance Prediction and Tivoli Enterprise Data Warehouse.

The Gathering Historical Data component uses data collected by specific Tivoli Monitoring resource models to populate a database on the Tivoli server where it is installed. The

collected data is aggregated every 24 hours and added to the Tivoli Monitoring database, from where it can be used in analyses that help plan network growth using key system metrics.

**Note:** For installing, configuring, and using Gathering Historical Data, follow the same information given about the TDS Configuration component in *Tivoli Decision Support for Server Performance Prediction (Advanced Edition): Release Notes, Version 2.1*.

For more information about Tivoli Enterprise Data Warehouse, see the the following publications:

- *Tivoli Distributed Monitoring Warehouse Enablement Pack: Implementation Guide*
- *Enabling an Application for Tivoli Enterprise Data Warehouse*

## Components

Tivoli Monitoring in a Tivoli environment and can be installed and configured from any *Tivoli management region server* (Tivoli server). Tivoli Monitoring has components that can be installed on the Tivoli server and gateways. On the endpoint, the product component is automatically installed at the first profile distribution.

Figure 2 on page 14 is a diagram of the components of the product. It also shows the various monitors that can receive information from the product.

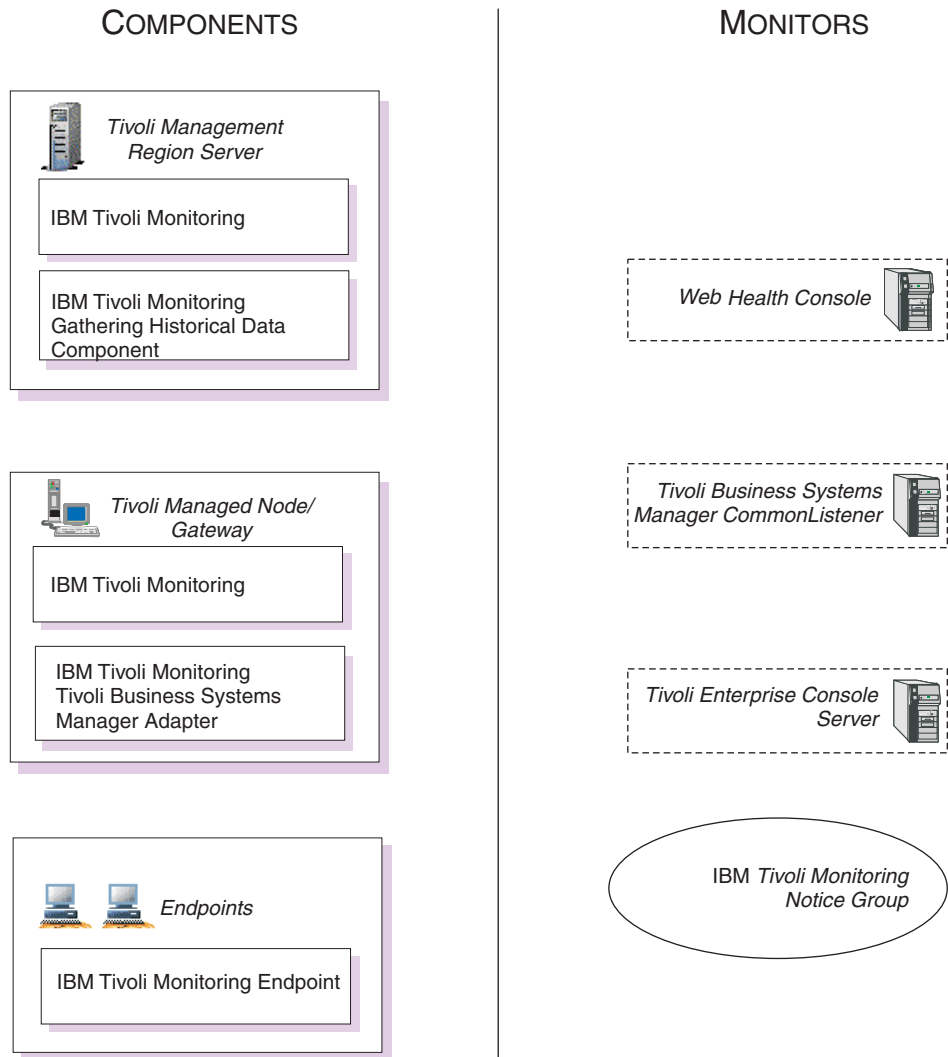


Figure 2. Components and event monitors of IBM Tivoli Monitoring

Details of the components are as follows.

### Tivoli Monitoring Base Component

The product component should be installed on the Tivoli management region server and all gateways to which are attached endpoints that you wish to monitor. In this guide, the term *gateway* is used to mean a managed node configured as a gateway. Installation instructions can be found in Chapter 2, “Installing” on page 19.

The component comprises a graphical user interface and a command line interface, available at both server and gateway, and all functions of the product can be controlled from either platform. However, you should note that the database of available default resource models is maintained at the server; commands issued to manage resource models from gateways are routed to and performed on the server.

In addition it can be configured to operate the heartbeat function for all endpoints directly attached to the system on which it is installed.

For more information, see Chapter 4, “Heartbeat Function” on page 59.

## Web Health Console

The Web Health Console is the web-based graphical interface for Tivoli Monitoring that runs on any system that can be connected via TCP/IP to the Tivoli management region. The Web Health Console allows you to drill down through a specific endpoint, a profile, and resource model, to view real time information about a specific problem. In this way you can display and check the status or health of a set of endpoints. You can use the Web Health Console to work with real-time data or with historical data that has previously been logged to a local database.

For more information, see Part II, “Web Health Console” on page 81

## Tivoli Business Systems Manager Adapter

This component is installed alongside the product component on a gateway (or a server which is directly attached to endpoints). It feeds discovery information and Tivoli Monitoring events to the Tivoli Business Systems Manager.

For more information see Chapter 6, “Integration with Tivoli Business Systems Manager” on page 73

## Endpoint Component

The endpoint component requires a Tivoli management agent to be installed on the endpoint. It performs the resource management through one or more resource models that are distributed to the endpoint with a Tivoli Monitoring profile. It is installed automatically when a Tivoli Monitoring profile is distributed to the endpoint for the first time.

## Collecting the Data

This section describes how the product obtains the data required for monitoring at the endpoints.

### Resource Model Data Flow

Figure 3 on page 16 shows the data flow for a resource model.

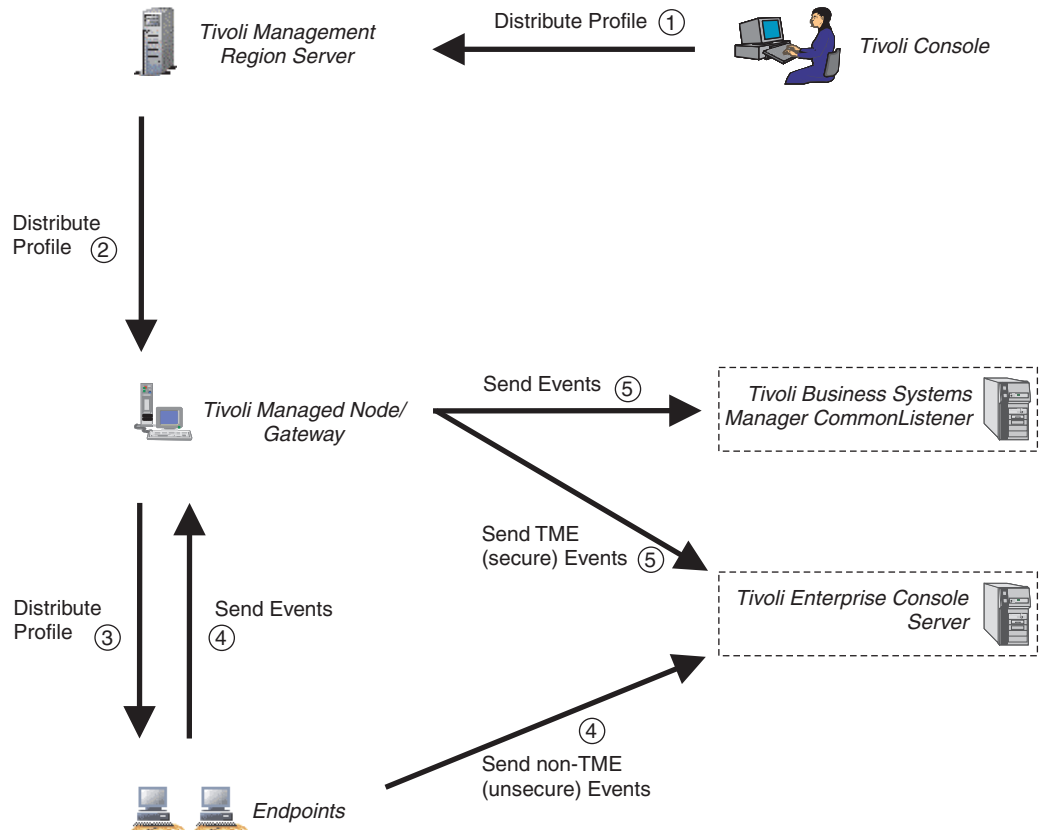


Figure 3. Data flow for profile distribution and resource model monitoring activities

The resource model is distributed (pushed) as part of a profile to the endpoint via the gateway (steps 1, 2 and 3). It commences monitoring the endpoint’s resources and sends information about all events to the gateway and about unsecure events directly to the Tivoli Enterprise Console server (step 4). The gateway, depending on which monitors are enabled, distributes events to The Tivoli Business Systems Manager CommonListener and secure events to the Tivoli Enterprise Console server (step 5).

## Obtaining Resource Data at the Endpoint

Tivoli Monitoring uses processes that form part of the endpoints’ operating systems to obtain resource data, as follows:

### Windows operating systems

On Windows operating systems, the product uses Windows Management Instrumentation (WMI), which is Microsoft’s implementation of CIM. WMI enables applications, including Tivoli Monitoring, to retrieve information about the current status of a system.

**Note:** WMI is supplied as part of Windows 2000. On Windows NT<sup>®</sup>, WMI must be installed on each system.

Other collection agents can also be used, and resource models can be defined that monitor resources not managed by WMI.

### UNIX and Linux operating systems

On UNIX and Linux operating systems, the information collection agent is incorporated in the product, based on CIM specifications.



## Security Considerations

Tivoli Monitoring exploits the infrastructure provided by Tivoli Management Framework, Version 3.7.1, to enable the functionality of the product across firewalls. To protect the privacy and data integrity, Tivoli Management Framework 3.7.1 enables you to either configure Bulk Data Transfer (BDT) proxy mechanism and Secure Socket Layer 3 (SSL3) encryption support, or install the Tivoli Management Framework Firewall Security Toolbox.

The Web Health Console, which does not run under Tivoli Management Framework, can communicate with the other components across or within a firewall.

For further information, refer to the following documentation:

- *Tivoli Management Framework Firewall Security Toolbox Release Notes, Version 1.2*  
This provides information about installing and configuring the components of the Tivoli Management Framework Firewall Security Toolbox.
- *Tivoli Management Framework: Release Notes, Version 3.7.1*  
This book contains information about Security and New Command Options.
- *Tivoli Management Framework: Planning for Deployment Guide, Version 3.7.1*  
this book contains information about Secure Sockets Layer Data Encryption.
- *Secure Socket Layer Introduction and iKeyman User's Guide*  
This provides further background information about SSL3.
- *Tivoli Enterprise Management across Firewalls (redbook SG24-5510-01)*  
This provides background information and includes scenarios that refer to Tivoli Distributed Monitoring (Classic Edition).

## Tivoli Monitoring Workbench

Tivoli Monitoring Workbench is a stand-alone product that provides an integrated environment specifically for developing, debugging, and packaging resource models for Tivoli Monitoring. You can use the workbench to develop your own resource models or to modify existing resource models provided with Tivoli Monitoring.

For more information about the workbench, refer to the *IBM Tivoli Monitoring: Workbench User's Guide*.



# 2

## Installing

This chapter lists the supported operating systems and the prerequisite hardware and software for installing and running IBM Tivoli Monitoring and gives details of how to perform the following tasks:

- Installing the product with the Tivoli Software Installation Service
- Installing the product from the Tivoli Desktop
- Installing the product from the command line
- Upgrading from Tivoli Distributed Monitoring (Advanced Edition) 4.1
- Uninstalling the endpoint components
- Uninstalling the product from servers and gateways
- Adding the Notice Group
- Installing Java Runtime Environment

For information about integrating Tivoli Monitoring with other Tivoli products, see:

- Chapter 5, “Integration with Tivoli Enterprise Console Server” on page 67
- Chapter 6, “Integration with Tivoli Business Systems Manager” on page 73

## Supported Platforms

Table 5 details the supported platforms for servers, gateways, and endpoints:

Table 5. Supported platforms

System	Versions	Server	Gateway	Endpoint
AIX®	4.3.1 and 4.3.2	✓	✓	
	4.3.3 and 5.1	✓	✓	✓
Solaris	2.6	✓	✓	✓
	2.7 and 2.8	✓	✓	✓
Windows NT, Version 4.0	Service Packs 4, 5, 6, and 6a	✓	✓	✓
Windows 2000	Server, Advanced Server, Professional	✓	✓	✓
Windows XP (Whistler) <sup>(1)</sup>	Professional	✓	✓	✓
Red Hat Linux for Intel	7.0, 7.1	✓	✓	✓
Turbo Linux for Intel	6.1, 6.5	✓	✓	✓
SuSE Linux for Intel	7.1, 7.2	✓	✓	✓
SuSE Linux for S/390®	7.0	✓	✓	✓

Table 5. Supported platforms (continued)

System	Versions	Server	Gateway	Endpoint
HP-UX <sup>(2)</sup>	10.2	✓	✓	
	11.x	✓	✓	✓

Supported platforms notes:

1. The Tivoli Business Systems Manager Adapter cannot be installed on HP-UX gateways.

**Note:** Use of particular components or functions of the product may require the use of other than the minimum operating system versions shown here (see “Prerequisite Software”).

## Hardware Requirements

Table 6 lists the minimum hardware requirements for installing Tivoli Monitoring on servers, gateways and endpoints.

Table 6. Hardware Prerequisites for Tivoli Monitoring

Hardware	Minimum Requirement	
Server	CPU	Same requirements as Framework
	RAM	
	Disk space	2 MB
Gateway	CPU	Same requirements as Framework
	RAM	
	Disk space	2 MB
Windows Endpoint	CPU	266 MHz
	RAM	128 MB
	Disk space	6 MB
UNIX/Linux Endpoint	CPU	Same requirements as Framework
	RAM	
	Disk space	12 MB

Table 7 lists the minimum hardware requirements for installing the Tivoli Business Systems Manager Adapter on gateways.

Table 7. Hardware Prerequisites for Tivoli Business Systems Manager Adapter

Hardware	Minimum Requirement	
Gateway	CPU	Same requirements as Framework
	RAM	
	Disk space	2 MB

## Prerequisite Software

Prerequisite software information is provided in separate sections for Tivoli Management Framework and for other software.

## Tivoli Management Framework

Tivoli Monitoring requires as a minimum Tivoli Management Framework Version 3.7B, but Version 3.7.1 is recommended (some platforms require patches, as detailed below). However, if you wish to use the product across a firewall, Version 3.7.1 must be used (see “Security Considerations” on page 17 for further information).

### Tivoli Management Framework Patches Required for Version 3.7B

Specific platforms have specific patch requirements, as follows:

**Linux** On Linux systems, version 3.7.1 of Tivoli Management Framework is not available.

#### Windows 2000

On Tivoli Management Framework, 3.7B, patch 3.7-TMF-0010 is a prerequisite for any version of the Windows 2000 operating system, and you are recommended to install it on all other platforms where it is available.

## Other Software

Table 8 gives details of other software prerequisites, depending on your working environment:

*Table 8. Prerequisite Software*

Feature	Required Software
Systems using the Web Health Console product component	<ul style="list-style-type: none"> <li>■ Netscape 6.x or Internet Explorer 6.x must be installed on the system.</li> </ul>
Endpoints running UNIX or Linux	<ul style="list-style-type: none"> <li>■ Java Runtime Environment (JRE), 1.3.0 or 1.3.1 must be installed on the endpoint (endpoints running Solaris must have JRE 1.3.1-01):               <ul style="list-style-type: none"> <li>• If you already have an appropriate copy of JRE installed on the target system, after installing IBM Tivoli Monitoring you should use the provided task <b>DMLinkJre</b> to link the product to your existing JRE (see “Installation Options” on page 32). Use task <b>DMRemoveLinkJre</b> to remove the link to JRE.</li> <li>• If a UNIX or Linux endpoint does not have an appropriate copy of JRE 1.3.0 installed, you can install it from the product CD, using the instructions provided on page 33.</li> </ul> </li> </ul>
Endpoints running Windows NT	<ul style="list-style-type: none"> <li>■ Windows Management Instrumentation (WMI) must be installed, before you can use a profile on the endpoint. The minimum version is 1.1, but 1.5 is recommended.</li> </ul>
Use of Gathering Historical Data on Tier1 Tivoli management region server	<ul style="list-style-type: none"> <li>■ Patch 3.7-TMF-0035 must be installed, if the Tier1 Tivoli management region server is connected to Tier2 endpoints.</li> </ul>
Use of the Data Logging facility on endpoints running Windows NT	<ul style="list-style-type: none"> <li>■ The ODBC driver for Microsoft® Access 2000 must be installed on the endpoint. If the endpoint does not have Microsoft Access 2000 installed:               <ol style="list-style-type: none"> <li>1. Run the mdac_typ.exe file that is provided with Microsoft Data Access Components 2.1 (or later), at <a href="http://www.microsoft.com/data">http://www.microsoft.com/data</a>.</li> <li>2. Install Jet 4.0 Service Pack 3.</li> </ol> </li> </ul>

Table 8. Prerequisite Software (continued)

Feature	Required Software
Use of Tivoli Enterprise Console server	<ul style="list-style-type: none"> <li>■ Tivoli Enterprise Console server 3.7 plus patch 3.7-TEC-0004, or later</li> <li>■ If you want to send secure events to the Tivoli Enterprise Console server, the Adapter Configuration Facility (ACF) must be installed on both the Tivoli server and Tivoli management gateways used to distribute profiles to the endpoints.</li> </ul>
Use of Tivoli Business Systems Manager Adapter	<ul style="list-style-type: none"> <li>■ Tivoli Monitoring 5.1 must be installed on the gateway.</li> <li>■ Java Runtime Environment (JRE), 1.3.0 must be installed on the gateway (see “Installing Tivoli Business Systems Manager Adapter” on page 73).</li> <li>■ The Tivoli Business Systems Manager patches 1.5-BSM-0010, 1.5-BSM-0016 should be installed on the Tivoli Business Systems Manager server.</li> <li>■ The adapter cannot be installed on HP-UX gateways.</li> </ul>
Use of Software Installation Service (SIS) for installing the product components	<ul style="list-style-type: none"> <li>■ The following patches are required on SIS: 3.7-SISCLNT-0003 and 3.7-SISDEPOT-0003.</li> </ul>
Upgrading Tivoli server or gateway from Tivoli Distributed Monitoring for Windows 4.1	<ul style="list-style-type: none"> <li>■ Tivoli Distributed Monitoring for Windows 4.1 server/gateway component already installed.</li> </ul>

## Coexistence with Other Versions of Tivoli Monitoring

There are two families of monitoring products available from Tivoli:

- The Tivoli Monitoring family, which consists of:
  - Tivoli Monitoring 5.1
  - Tivoli Distributed Monitoring (Advanced Edition) 4.1
  - Tivoli Distributed Monitoring for Windows 3.7 and previous versions
  - Tivoli Manager for Windows NT, all versions
- The Tivoli Distributed Monitoring (Classic Edition) family, which consists of:
  - Tivoli Distributed Monitoring (Classic Edition) 3.7
  - Tivoli Distributed Monitoring 3.7 and previous versions

This section explains how these products can coexist with, or be upgraded to, IBM Tivoli Monitoring.

### Products within the Tivoli Monitoring Family

Products within the Tivoli Monitoring family must be upgraded to Version 5.1. None of the products of this family can coexist in the same Tivoli management region.. Therefore, for example, you cannot install Tivoli Monitoring 5.1 in the same Tivoli management region where Tivoli Distributed Monitoring (Advanced Edition) 4.1 is installed.

Versions of Tivoli Manager for Windows NT and older versions of Tivoli Distributed Monitoring for Windows should be upgraded first to Tivoli Distributed Monitoring for Windows NT 3.7, then to Tivoli Distributed Monitoring (Advanced Edition) 4.1 and, finally, to Tivoli Monitoring 5.1.

Full details of the steps required to upgrade from Tivoli Distributed Monitoring for Windows NT, 3.7 to Tivoli Monitoring 5.1 Appendix G, “Migration Considerations” on page 215.

Workbench and the endpoint component of the product can coexist on the same system but only if they are aligned (same product version, release, and patch).

## Products within the Tivoli Distributed Monitoring (Classic Edition) Family

All products within the Tivoli Distributed Monitoring (Classic Edition) family can coexist with Tivoli Monitoring 5.1. That is, they can be installed on the same server and gateways, and can access the same endpoints.

**Note:** There is no automatic migration from the Tivoli Distributed Monitoring (Classic Edition) product family to Tivoli Monitoring 5.1. However, the following sections give guidelines on the considerations that affect the decision to move from the Tivoli Distributed Monitoring (Classic Edition) product family to the Tivoli Monitoring product family.

## Backward Compatibility

Carefully consider the following information, which applies to backward compatibility:

- BAROC file (needed to enable monitoring of Tivoli Monitoring events on a Tivoli Enterprise Console server; refer to Chapter 5, “Integration with Tivoli Enterprise Console Server” on page 67 for details):
  - The BAROC file available with Tivoli Monitoring 5.1 can also be used with Tivoli Distributed Monitoring (Advanced Edition) 4.1 or with Tivoli Distributed Monitoring for Windows 3.7 Patch 3.
  - The BAROC file available with Tivoli Distributed Monitoring (Advanced Edition) 4.1 can also be used with Tivoli Distributed Monitoring for Windows 3.7 Patch 3.
  - The BAROC file available with Tivoli Distributed Monitoring for Windows 3.7 Patch 3 can also be used with Tivoli Distributed Monitoring (Advanced Edition) 4.1.
- Resource models created using Workbench:
  - Resource models created using Tivoli Distributed Monitoring for Windows 3.7 Workbench can be used also with Tivoli Monitoring 5.1, or with Tivoli Distributed Monitoring (Advanced Edition) 4.1.
  - Resource models created using Tivoli Distributed Monitoring (Advanced Edition) 4.1 Workbench can be used also with Tivoli Monitoring 5.1 (although resource models for UNIX endpoints will need to be redistributed).
  - Resource models created using Tivoli Monitoring 5.1 Workbench can only be used with Tivoli Monitoring 5.1.

## Installing/Upgrading the Product

IBM Tivoli Monitoring works in the Tivoli Management Environment and can be installed and configured from any Tivoli management region server.

You can install Tivoli Monitoring on any node either as a new product, or, if you have previously installed Tivoli Distributed Monitoring (Advanced Edition) 4.1, you can upgrade it to this product.

In either case, you can install the product in the following ways:

- With the Tivoli Software Installation Service
- From the Tivoli desktop
- From the command line interface (CLI)

### Before You Begin

Please read the following notes before you begin to install or upgrade the product:

1. Decide on which servers or gateways you are going to install/upgrade the product. You must install the product on the server before installing/upgrading it on any gateway.
2. Ensure that the selected nodes meet the hardware requirements for the installation.
3. Ensure that you have the correct prerequisite software installed on each node.
4. Determine for each node if you need to install the product from scratch or if you can upgrade from a previous version of the product. Ensure that the selected nodes meet the software requirements for the installation or upgrade.
5. Ensure that you have Administrator or root privileges to perform the installation or upgrade.
6. Back up your database before you begin to install or upgrade the product.

**Note:** The IBM Tivoli Monitoring Task Library is also installed with the product. It contains a set of predefined tasks associated with the product. If you wish to define additional tasks, you should create a separate task library, in order to avoid that the tasks you have created are lost in the event of an uninstall operation being carried out.

A full description of how to carry out a fresh installation now follows; the product upgrade is described in Appendix G, “Migration Considerations” on page 215.

### Installing with Tivoli Software Installation Service

Tivoli Software Installation Service (SIS) can install multiple Tivoli products on multiple systems in parallel. This Java-based product can install more products on more systems in much less time than the Framework’s install facility. SIS performs product prerequisite checks and, if defined, user-specified prerequisite checks, ensuring as few install failures as possible. In most cases, failures occur only when machines are turned off or removed from the network.

SIS also creates an installation repository into which you can import the installation image of one or more Tivoli products. You can import only those interpreter types needed in your environment, which saves disk space and import time. The installation repository is then the source of all your Tivoli installations. You can even share a single installation repository across multiple Tivoli Management Regions.

For details of the platforms on which the Software Installation Service can run and for instructions on installing and using this product, refer to the *Tivoli Software Installation Service User’s Guide* and the *Tivoli Software Installation Service Release Notes*.

For instructions on how to install SIS in your Tivoli environment and how to use SIS to install products, refer to the *Tivoli Software Installation Service User’s Guide*.



## Installing from the Tivoli Desktop

You can install the following product components from the Tivoli Desktop:

### IBM Tivoli Monitoring, Version 5.1.0

Includes the files that enable you to create and distribute profiles and run IBM Tivoli Monitoring commands. Install this on the Tivoli Management Region server and the gateways to which profiles will be distributed in your region (it must be installed on the server before installing it on any gateway).

### Tivoli Monitoring TBSM Adapter, Version 5.1.0

Includes the files that enable you to use Tivoli Business Systems Manager to receive events sent from the resource models. Install this on the Tivoli management region gateway. If the Tivoli management region server also has the Tivoli Framework Gateway component installed on it to provide gateway services to directly-connected endpoints, install Tivoli Business Systems Manager Adapter also on the Tivoli management region server. For full instructions on how to install, configure and use this component, see Chapter 6, “Integration with Tivoli Business Systems Manager” on page 73.

### Tivoli Monitoring Gathering Historical Data Component, Version 5.1.0

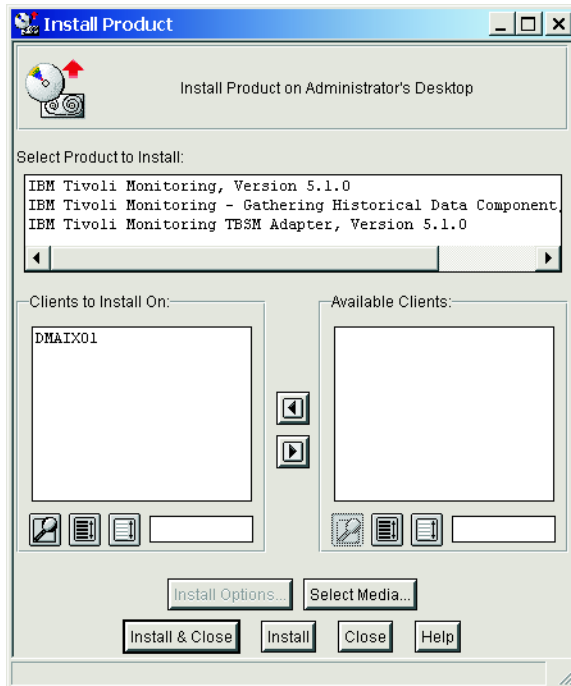
Includes the files that enable you to integrate Tivoli Decision Support for Server Performance Prediction and Tivoli Enterprise Data Warehouse with Tivoli Monitoring,

Using Tivoli Monitoring as a means of collecting information that are analyzed by those products. To install, configure, and use the Gathering Historical Data Component, refer to the following publications:

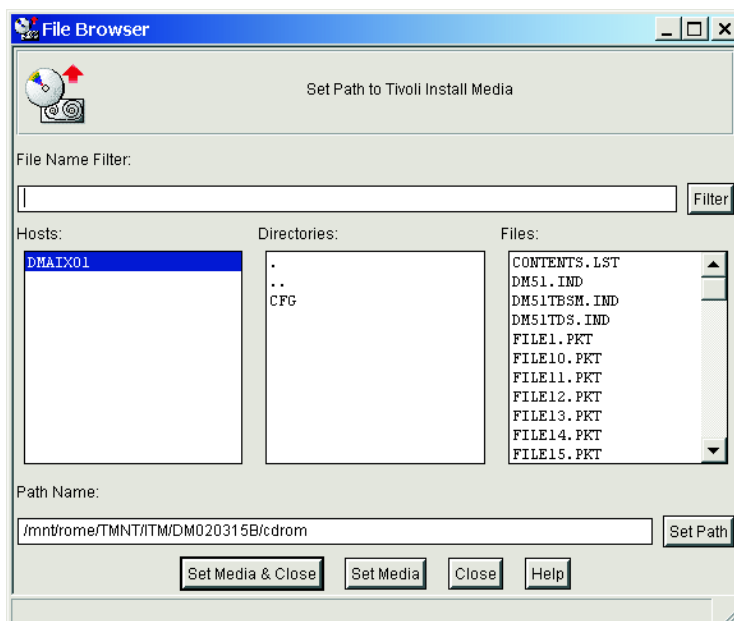
- *Tivoli Decision Support for Server Performance Prediction: Release Notes*
- *Tivoli Distributed Monitoring Warehouse Enablement Pack: Implementation Guide*
- *Enabling an Application for Tivoli Enterprise Data Warehouse*

To install the product from scratch from the Tivoli desktop, perform the following steps:

1. Insert the product CD-ROM into the CD-ROM drive.
2. Start the Tivoli desktop.
3. From the **Desktop** menu select **Install → Install Product**.  
The Install Product dialog opens.



4. If the IBM Tivoli Monitoring product components are displayed in the **Select Product to Install** scrolling list, go to step 8.
- OR—  
If the IBM Tivoli Monitoring product components are not displayed in the **Select Product to Install** scrolling list, continue with step 5
5. Click **Select Media**.  
The File Browser dialog opens.



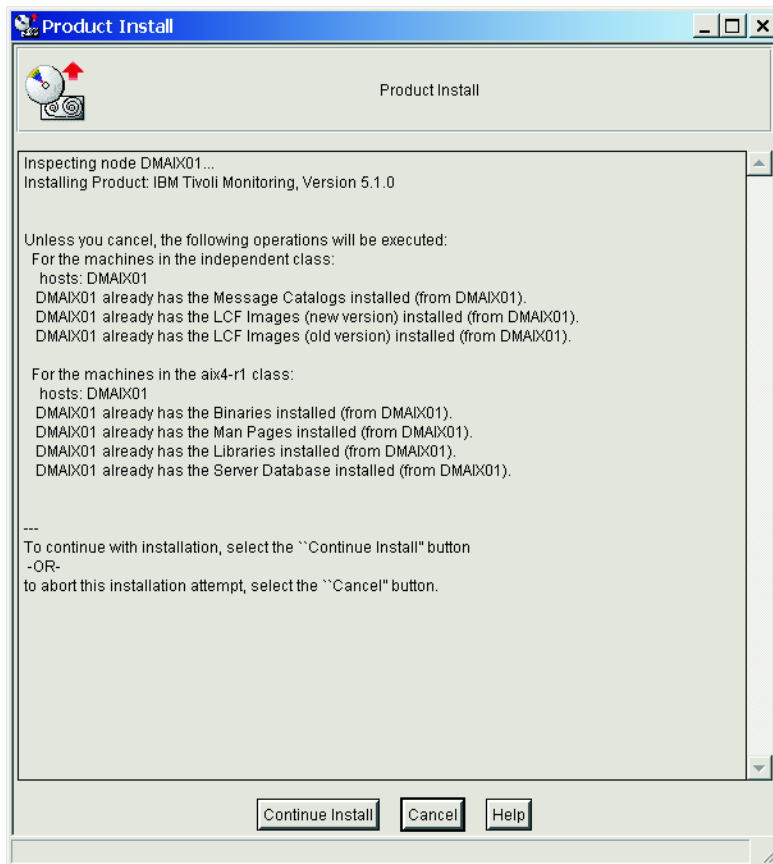
6. Type or select the location of the Tivoli CD-ROM in the **Path Name** field by completing one of the following tasks:

- Type the complete path name in the **Path Name** field.
  - Browse the file system by completing the following steps:
    - a. In the **Hosts** scrolling list, select the host (or drive) on which the CD-ROM is mounted, by double-clicking it. Choosing a host updates the **Directories** scrolling list to show the directories (under root) of the host you selected.
    - b. In the **Directories** scrolling list, double-click the directory that contains the install media. Choosing a directory updates the **Files** list. Descend the directory tree until you have found the appropriate directory.
7. Click **Set Media & Close**.
- The Install Product dialog reopens. The **Select Product to Install** scrolling list shows the products that are available for installation.
8. Select one of the following products from the **Select Product to Install** scrolling list:
- **Tivoli Monitoring 5.1.0**
  - **Tivoli Monitoring 5.1.0 - TBSM Adapter**

**Note:** This component cannot be installed on HP-UX gateways.
  - **Tivoli Monitoring 5.1.0 - Gathering Historical Data Component**

**Note:** This product is required to enable Tivoli Decision Support for Server Performance Prediction and Tivoli Enterprise Data Warehouse to work with Tivoli Monitoring. Refer to:

    - *Tivoli Decision Support for Server Performance Prediction: Release Notes*
    - *Tivoli Distributed Monitoring Warehouse Enablement Pack: Implementation Guide*
    - *Enabling an Application for Tivoli Enterprise Data Warehouse*
9. Ensure that the appropriate host is displayed in the **Clients to Install On** scrolling list. If it is not displayed, select the server on which you want to install the selected product from the **Available Clients** scrolling list, and use the left arrow to move the selected server to the **Clients to Install On** scrolling list.
10. Click **Install** to begin installing the product on gateways.
- The Product Install dialog opens. It provides a list of the operations to be performed and warns of any problems you might want to correct before installing.



11. Review the status information and click **Continue Install**.  
The Product Install dialog informs you when installation is complete.
12. If you are going to send heartbeat events to the Tivoli Monitoring notice group, follow the instructions in “Adding the Notice Group” on page 31.
13. Click **Close**.

## Installing from the Command Line

Use the **winstall** command to install Tivoli Monitoring from the command line.

To install Tivoli Monitoring on a gateway from a CD-ROM, enter the following command:

```
winstall -c path/cdrom -i index_file managed_node
```

where:

*path* Specifies the complete path to the **/cdrom** directory in the CD-ROM. (Use forward slashes to specify the path, regardless of the operating system you are running.)

*index\_file* Specifies the product index file from which Tivoli Monitoring is installed. This value can be one of the following:

- **DM51**. The index file for Tivoli Monitoring
- **DM51TBSM**. The index for Tivoli Business Systems Manager Adapter (Tivoli Monitoring TBSM Adapter, Version 5.1)

**Note:** This component cannot be installed on HP-UX gateways.

- **DM51TDS.** The index for Tivoli Decision Support for Server Performance Prediction (Tivoli Monitoring TDS Configuration, Version 5.1)

**Note:** This product is required to enable integration with Tivoli Decision Support for Server Performance Prediction, see *Tivoli Decision Support for Server Performance Prediction: Release Notes*.

*managed\_node*

The selected product will be installed on the gateway you specify. If you do not specify a gateway, the product is installed on all available clients.

If you are going to send heartbeat events to the Tivoli Monitoring notice group, follow the instructions in “Adding the Notice Group” on page 31.

For more information about all the parameters you can use with the **winstall** command, refer to the *Tivoli Management Framework Reference Manual*.

## Upgrading from Tivoli Distributed Monitoring (Advanced Edition) Version 4.1

The procedure for upgrading Tivoli Distributed Monitoring (Advanced Edition) 4.1 to Tivoli Monitoring 5.1 contains the following main steps:

1. Use the **wdmcmd –stop** command to stop the Tivoli Distributed Monitoring (Advanced Edition) processes on all endpoints.  
For information about using the **wdmcmd**, see the *Tivoli Distributed Monitoring: User’s Guide*
2. Install the server/gateway component at the Tivoli management region server (more details are given in “Upgrading the Server/Gateway Components” on page 30).
3. As part of the installation process, all existing resource models in the Tivoli management region server will be upgraded to be compatible with the new software.
4. Install the server/gateway component at all gateways attached to the Tivoli management region server, whose endpoints you wish to monitor. More details are given in “Upgrading the Server/Gateway Components” on page 30.
5. If you modified a resource model that was provided with Tivoli Distributed Monitoring (Advanced Edition) 4.1, perform the following steps otherwise skip to step 6:
  - a. Open the Tivoli Monitoring 5.1 resource model that corresponds to the resource model that you modified.
  - b. Extract all the dependencies belonging to the resource model.
  - c. Replace the dependencies in the modified resource model by importing the dependencies that you extracted.
  - d. Regenerate the resource model tar file.
  - e. Uninstall the old resource model and install the new tar file in the Tivoli management region.
  - f. The new resource model can now be added to the profile it belonged to.
6. On UNIX and Linux, redistribute all profiles to all endpoints. This redistribution will automatically install the Tivoli Monitoring endpoint component and will start the resource models.

On Windows endpoints, the engine binaries will be upgraded by issuing the **wdmcmd -restart** command. After you have stopped the engine, wait a minimum of 15 minutes before issuing the **wdmcmd -restart** command.

**Note:** On Windows, only the Tivoli Enterprise Event messages are modified in the resource models. To update the message catalog that was distributed to the endpoints, redistribute all profiles to all endpoints.

### Upgrading the Server/Gateway Components

To upgrade from Tivoli Distributed Monitoring (Advanced Edition) 4.1, perform the following steps:

1. Insert the product CD-ROM into the CD-ROM drive.
2. Start the Tivoli desktop.
3. Select **Install** → **Install Patch** from the Desktop menu.  
The Install Patch dialog opens.
4. If the Tivoli Monitoring upgrade product is already in the **Select Product to Install** scrolling list, go to step 7. Otherwise, continue with step 5.
5. Click **Select Media**.  
The File Browser dialog opens.
6. Specify the location of the upgrade directory (%UPGRADE) in the **Path Name** text box by completing one of the following tasks:
  - Type the complete path name in the **Path Name** text box.  
—OR—
  - Browse the file system by completing the following steps:
    - a. In the **Hosts** scrolling list, select the host (or drive) on which the upgrade directory is mounted, by double-clicking it. Choosing a host updates the **Directories** scrolling list to show the directories (under root) of the host you selected.
    - b. In the **Directories** scrolling list, double-click on the directory that contains the upgrade. Choosing a directory updates the **Files** scrolling list. Descend the directory tree until you have found the appropriate directory.
7. Click **Set Media & Close**.  
The Install Patch dialog reopens. The following upgrade that is available for installation is displayed in the **Select Patch to Install** scrolling list:

**IBM Tivoli Monitoring Upgrade, Version 4.1 (DMA) to 5.1.0**  
Includes the files that enable you to upgrade Tivoli Distributed Monitoring (Advanced Edition) 4.1 to Tivoli Monitoring 5.1. Install this on the Tivoli Management Region server and on each gateway.
8. Ensure that the appropriate host is displayed in the **Clients to Install On** scrolling list. If the required host is not in the list, use the left and right arrow buttons to move the clients from one list to the other.
9. In the Install Patch dialog, from the **Select Patch to Install** scrolling list, install the following upgrades in the order shown by repeating step 10 to step 13:
  - **IBM Tivoli Monitoring Upgrade, Version 4.1 (DMA) to 5.1.0**

10. Select the machine on which you want to install the selected product from the **Available Clients** scrolling list. Only gateways are displayed.
11. Use the left arrow to move the selected gateways and the Tivoli Management Region server to the **Clients to Install On** scrolling list.
12. To begin installing the upgrade, click **Install**.  
The Product Install dialog provides a list of the operations that occur and warns you of any problems that you might want to correct before installing.
13. Review the status information and click **Continue Install**.  
The Product Install dialog informs you when installation is complete.
14. Click **Close**.

**Note:** All resource models are automatically updated. You do *not* have to delete any resource models and add them to a profile again.

If you had the Tivoli Business Systems Manager Adapter or the Tivoli Decision Support components installed, then install the appropriate upgrade.

**Note:** If you have the TDS Configuration component already installed, the appropriate upgrade is called **IBM Tivoli Monitoring – Gathering Historical Data Component, Upgrade 4.1 (DMA) to 5.1.0**.

## Installing a Default Resource Model

If you want to install a new default resource model, or an existing default resource model that has been modified with Tivoli Monitoring Workbench, enter the following command from command line:

```
wdmrm -add resource_model_tarfile
```

For more information, refer to Appendix A, “Commands” on page 107.

If you want to remove an existing default resource model, enter the following command from command line:

```
wdmrm -remove resource_model_name
```

For more information, refer to Appendix A, “Commands” on page 107.

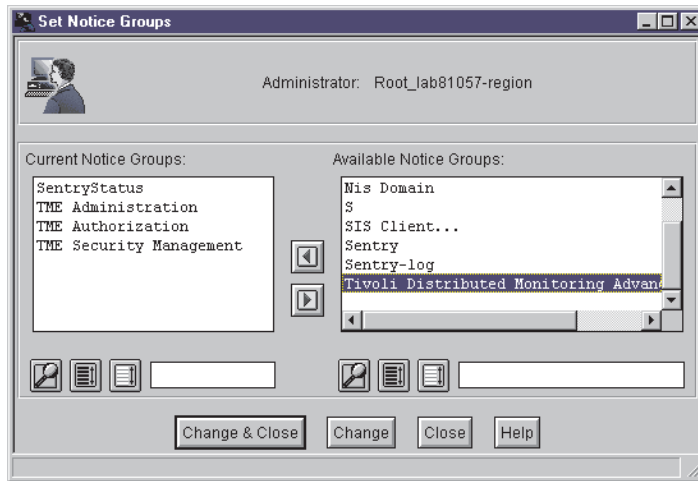
If you want to create a new default resource model or modify an existing one, you can do it using the Workbench. For more information on how to do it, see *IBM Tivoli Monitoring: Workbench User's Guide*.

## Adding the Notice Group

If you want to send events to the Tivoli Monitoring notice group, use the following procedure to create the link with the notice group:

1. From the Tivoli Desktop select the **Administrators** icon. The Administrators dialog opens.

2. Right-click on the Administrators icon for your region and select **Edit Notice Group Subscriptions**. The Set Notice Group dialog opens.



3. Select the Tivoli Monitoring notice group and click on the left arrow to move it to the **Current Notice Groups** pane.
4. Press **Change & Close** to set the notice group definitions to include the selected notice group.

## Installing Java Runtime Environment

Various components of Tivoli Monitoring require Java Runtime Environment (JRE), Version 1.3.0 (see Table 8 on page 21 for full details). There are three available installation methods:

- JRE 1.3.0 is available on the product CD for installation using the Tivoli Software Installation Service (SIS).
- JRE 1.3.0 is also available on the product CD in compressed format, for manual installation or installation through the **wmdistrib -J** command (for details on this command, refer to Appendix A, “Commands” on page 107).
- On UNIX/Linux endpoints, if the endpoint already has an appropriate version of JRE installed, you need only link the product component to the existing JRE using a task provided with the product.

### Important Installation Notes:

1. If JRE is not installed or linked as described here, the product will fail on the system in question.
2. The use of an IBM version of JRE is recommended for support purposes.
3. All versions of the JRE on the CD are supplied by IBM, except for the HP-UX version.
4. Prerequisites for JRE can be found on the appropriate readme file in the JRE directory on the product CD.

## Installation Options

The full details of the installation options are as follows:



**Installation using SIS**

Use SIS to select and install JRE. The JRE product name in SIS is *ITM 5.1 - JRE 1.3.0*.

**Manual Installation**

To install JRE from the compressed files on the Tivoli Monitoring 5.1 Tools CD, follow these steps:

1. Copy the `jre13.tar.gz` file from the directory on the Tools CD that applies to the operating system where JRE is to run, to the directory where you want to install JRE:

**AIX**            JRE/aix4-r1

**Linux**            JRE/linux-ix86

**Linux S/390**    JRE/linux-s390

**Solaris**          JRE/solaris2

**HP-UX**            Refer to the installation instructions, which can be found at HP's Web site ([www.hp.com](http://www.hp.com)), on the page entitled HP-UX Runtime Environment for the Java 2 Platform, Version 1.3.0.

**Windows**        JRE/w32-ix86

2. From the directory where you copied the `jre13.tar.gz` file, issue the command:  
`gzip -dc jre13.tar.gz | tar xvf -`
3. If JRE is being installed for a UNIX/Linux endpoint, run the **DMLinkJre** task, as described below.

**Link to an existing version using the DMLinkJre task**

On UNIX or Linux endpoints, if you already have the appropriate version of JRE installed on the target system, you can use a Tivoli task to link the product to JRE.

1. From the Tivoli Desktop select **Desktop** → **TMR Connections** → **Top Level Policy Regions**. The Top Level Policy Regions dialog opens.
2. Double-click the **TivoliDefaultMw2kRegion** icon. The Policy Region dialog for that policy region opens.
3. Double-click the Tivoli Monitoring Tasks icon. The Task Library dialog for that library opens.
4. Run the **DMLinkJre** task. The Execute Task dialog opens.
5. Select the endpoint or endpoints on which you want to link the JRE, and click **Execute** (you can also identify endpoints using one or more profile managers).

**Note:** Ensure that the path where JRE is physically installed, is the same on all selected endpoints.

6. On the **DMLinkJre** dialog, enter the complete path where JRE is installed (not including the `/bin` directory). Click **Set & Execute**. JRE will be linked to the product on all selected endpoints.

## Enabling Tivoli Monitoring to Work with Tivoli Enterprise Data Warehouse

After installing the Gathering Historical Data component, to enable Tivoli Monitoring to collect data that can be used by Tivoli Enterprise Data Warehouse:

1. From a command prompt on a Tivoli management region server where the Tivoli environment was set, run the following script:  
`twh_enable.sh`
2. From a command prompt on the RIM host where the Tivoli environment was set, switch to the `$BINDIR/TME/Tmw2k/TDS/rdbcfg` directory and run the following script:  
`run_query.sh twh_enable_update`

**Note:** If you are using a DB2 or Informix RDBMS, the script is to be run from a DB2 or Informix command line on the database server. This because you cannot modify the DB2 database from the DB2 Client Command Line Processor without an existing database connection, and because Informix uses the `dbaccess` facility, which is shipped as part of the Informix server.

## Uninstalling the Product

This section describes how to uninstall the product components from endpoints, gateways and servers, and how to remove the automatic endpoint reboot capability.

**Note:** The Tivoli Monitoring Task Library is also uninstalled with the product. If it contains any user-defined tasks they will be lost. Therefore you should create any user-defined tasks in a separate library.

To uninstall the product follow this sequence:

1. Uninstall the endpoint components
2. Uninstall from server and gateways

### Uninstalling the Endpoint Components

Tivoli Monitoring provides a script to remove Tivoli Monitoring from endpoints. The script is named `dm_endpoint_uninstall.sh` and is stored in `$BINDIR/TME/Tmw2k`.

The `dm_endpoint_uninstall.sh` script performs the following steps:

1. Stops the application if it is running.
2. Removes all the Tivoli Monitoring files that were downloaded to endpoints with the `Tmw2k` profile from those endpoints.
3. Where appropriate, removes the application keys from the Windows registry.

To uninstall the endpoint components, run the script:

```
dm_endpoint_uninstall.sh {endpoint_label}...
```

where *endpoint\_label* is the label of the endpoint. If more than one endpoint label is specified, they should be separated by spaces.

### Uninstalling from Servers and Gateways

The Tivoli Management Framework provides the `wuninst` command to remove Tivoli applications from a specified node or from the entire region. You can use the `wuninst` command with product tags that are specific to Tivoli Monitoring to remove the product from any system in your environment or from the entire region.

---

To uninstall Tivoli Monitoring from servers or gateways, enter the following command:

```
wuninst tagname destination_target -rmfiles
```

where:

*tagname* Is one of the registered product tags for IBM Tivoli Monitoring that are provided by Tivoli.

**Note:** These tags *do not* indicate the version or release of the product)

- **DM\_Advanced\_Edition\_TDS**, to indicate Tivoli Decision Support for Server Performance Prediction.
- **DM\_Advanced\_Edition\_TBSMA**, to indicate Tivoli Business Systems Manager Adapter.
- **TMNT\_3.6.2**, to indicate Tivoli Monitoring.

*destination\_target*

Is the gateway or server from which you want to remove the product. If you specify a server, the product is uninstalled from all its gateways.

For more information about command line syntax and usage of the **wuninst** command, refer to the *Tivoli Management Framework Reference Manual* and the *Tivoli Management Framework Release Notes: Server, Gateway, and Endpoint*.

---

# 3

## Using the Product

---

This chapter describes the purpose of profiles in the Tivoli environment and presents the range of customization options you can use to fine-tune resource models to ensure optimal availability of resources.

These options are discussed in the context of using the graphical user interface (GUI) to perform them. However, many users will want to use the commands provided in the command line interface (CLI), and there are some tasks which can be performed only using the CLI, such as adding new resource models at the Tivoli server. All users are recommended to read through this chapter to understand the concepts of using the product and the structure of profiles and resource models. Those that want or need to use the command line interface will find full details of the commands in Appendix A, “Commands” on page 107. In addition, reference will be made in the various sections of this chapter to the corresponding commands.

This chapter provides details of how to perform the following tasks:

- Setting up Managed Resources
- Creating Profile Managers and Profiles
- Default Resource Models
- Adding a Default Resource Model to a Profile
- Adding a Customized Resource Model to a Profile
- Customizing a Resource Model:
  - Customizing Thresholds and Event Rules; includes:
    - Specifying Built-in Actions for an Event (Windows)
    - Specifying Tasks for an Event
  - Customizing Parameters
  - Customizing Scheduling Information; includes:
    - Setting the Monitoring Period
    - Creating Schedule Rules
  - Customizing Data Logging Information
- Setting Distribution Options for a Profile
- Managing Profiles and Resource Models at Endpoints
- Managing Tivoli Monitoring at Gateways

## Overview

Before you can monitor resources at an endpoint, you must create one or more monitoring profiles. You then add the required resource models to the profile. You can add resource models with the default values preconfigured by Tivoli, or you can customize the resource models as required.

### Profiles

A profile is a collection of information corresponding to a Tivoli application resource. In most Tivoli applications, a profile contains information that is specific to a particular application or a particular database grouping.

Tivoli Monitoring profiles enable you to define resource models and distribute these resource models to subscribing endpoints. When you first set up a Tivoli Monitoring, no monitoring occurs until you distribute the profile. Resource models can be added to a profile, deleted from it, or modified. These changes to the profile do not take effect until you redistribute the profile. The subscribers can be endpoints and other profile managers.

### Key Points About Profiles

There is a relationship between profiles, profile managers, policy regions, endpoints and resource models:

- Profile managers are created within a policy region and must be a managed resource of that region. See “Setting up Managed Resources”, for more details.
- Profile managers contain profiles; they also administer profiles and subscriber lists (groups of endpoints). See “Creating Profile Managers and Profiles” on page 39, for more details.
- Profiles include one or more resource models, which can be customized or can be defined using the defaults. If you want to modify the default resource models, see “Default Resource Models” on page 41.

## Setting up Managed Resources

Before creating profile managers and profiles for Tivoli Monitoring, you must make the Tivoli Monitoring profile available for use as a managed resource.

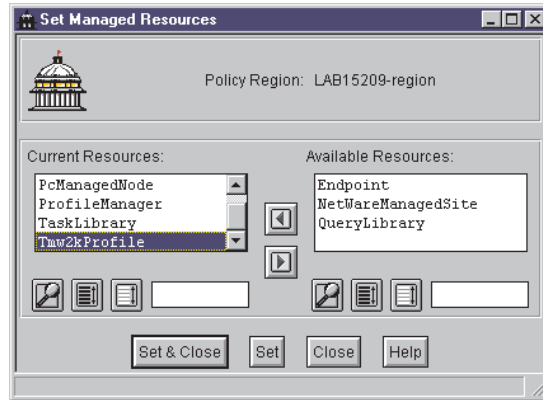
**Note:** This procedure must be performed only once for each policy region in which Tivoli Monitoring profiles will be used.

To set up managed resources, perform the following steps:

1. In the Policy Region dialog, select **Properties** → **Managed Resources**.



The Set Managed Resources dialog opens.



2. Ensure that the **Tmw2kProfile** is displayed in the **Current Resources** list.
3. Click **Set & Close**.  
The Policy Region dialog reopens.

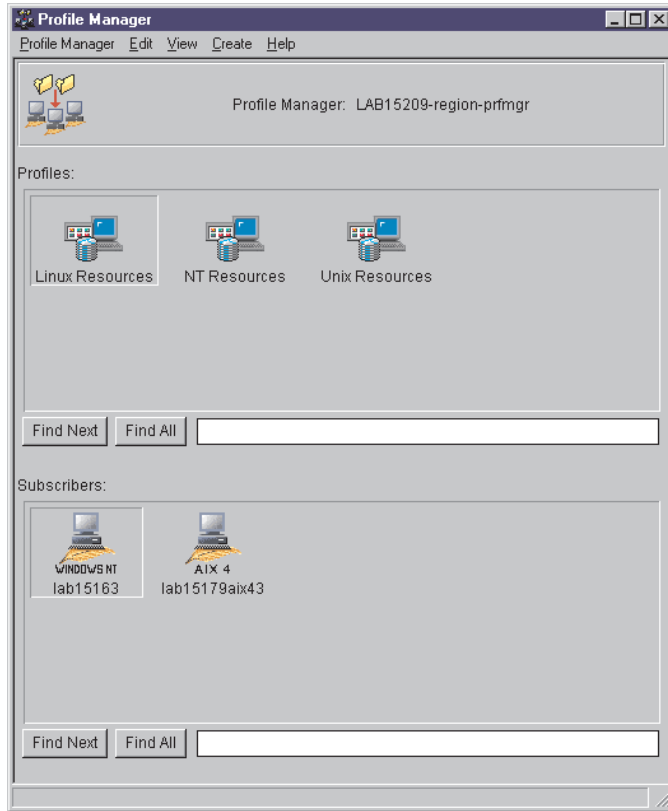
## Creating Profile Managers and Profiles

To create a new profile, perform the following steps:

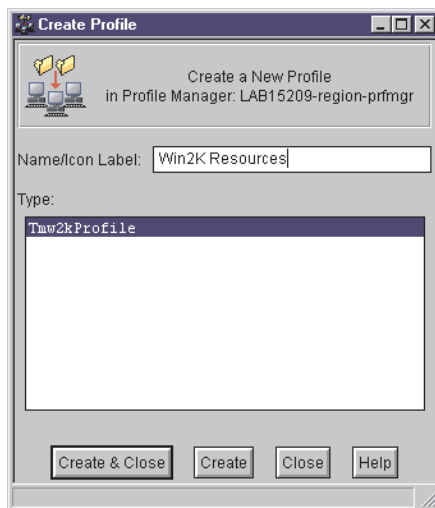
1. From a Policy Region dialog, select **Create** → **Profile Manager**.  
The Create Profile Manager dialog opens.



- a. Type the name of the new profile manager.
  - b. If you want to be able to push the profile to endpoints, click **Dataless Endpoint Mode**.  
**Note:** You cannot subscribe other profile managers to the profile manager using the dataless endpoint mode.
  - c. Click **Create & Close**.  
The new profile manager is created in the Profile Manager dialog.
2. In the Policy Region dialog, double-click the **Profile Manager** icon.  
The Profile Manager dialog opens.



3. Select **Create → Profile**.  
The Create Profile dialog opens.



4. Type the name of the new profile in the **Name/Icon Label** field.
5. Select **Tmw2kProfile** from the **Type** list.
6. Click **Create & Close**.  
The Profile Manager dialog reopens. It now contains an icon for the new profile.

## Using the CLI to Manage Profiles at the Server

The commands for managing profiles at the server are as follows:



**wdmloadprf**

Loads and updates profiles at a Tivoli management region server

**wdmdumpprf**

Writes the full details of a profile to the standard output

**wdmeditprf**

Allows you to edit the definition of a profile, including all resource model details

**wdmdistrib**

Allows you to distribute profiles to the endpoints

Details of all commands can be found in Appendix A, “Commands” on page 107.

## Default Resource Models

As stated above, a profile includes one or more resource models, which can be customized or can be defined using the defaults. Default resource models can be changed, and new default resource models created, using the Tivoli Monitoring Workbench. Tivoli Monitoring also includes a command that makes those new or changed models available at the server, as follows:

**wdmrm** Adds, lists or removes a specified default resource model at the Tivoli management region server or managed node/gateway from where it is issued. It also adds the NLS catalog to an already installed default resource model.

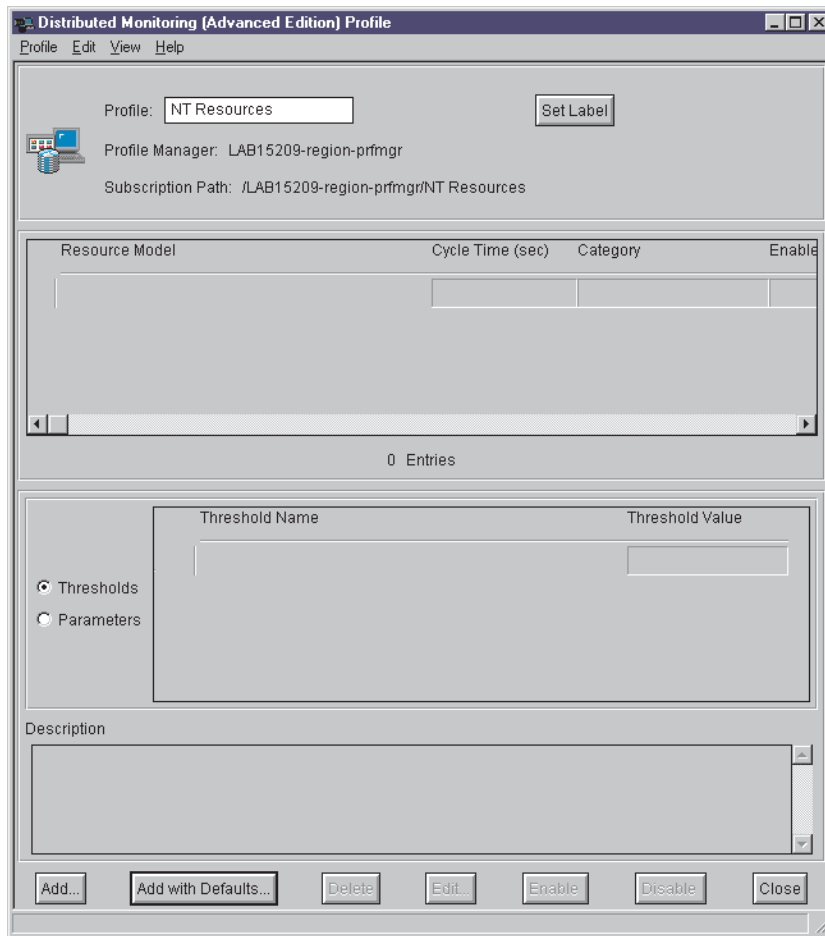
Details of all commands can be found in Appendix A, “Commands” on page 107.

## Adding a Default Resource Model to a Profile

To add a resource model to a profile using its default values, perform the following steps:

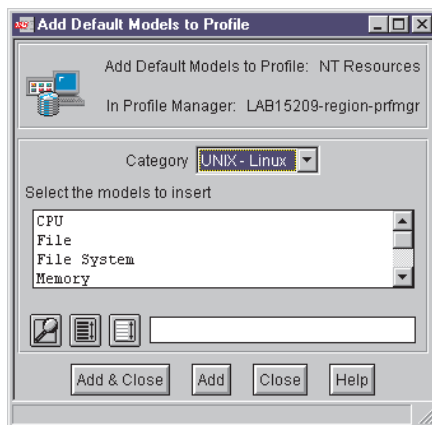
1. Open the Profile Manager dialog.
2. Double-click the profile icon to which you want to add a resource model.

The Tivoli Monitoring Profile dialog opens.



- 3. Click **Add with Defaults**.

The Add Default Models to Profile dialog opens.



- 4. Select the required resource models to add.

- 5. Click **Add & Close**.

The resource models are added to the Tivoli Monitoring Profile dialog.

## Using the CLI to Add a Default Resource Model to a Profile

Use the following command to add a default resource model to a profile:

**wdmeditprf** Allows you to edit the definition of a profile, including all resource model details.

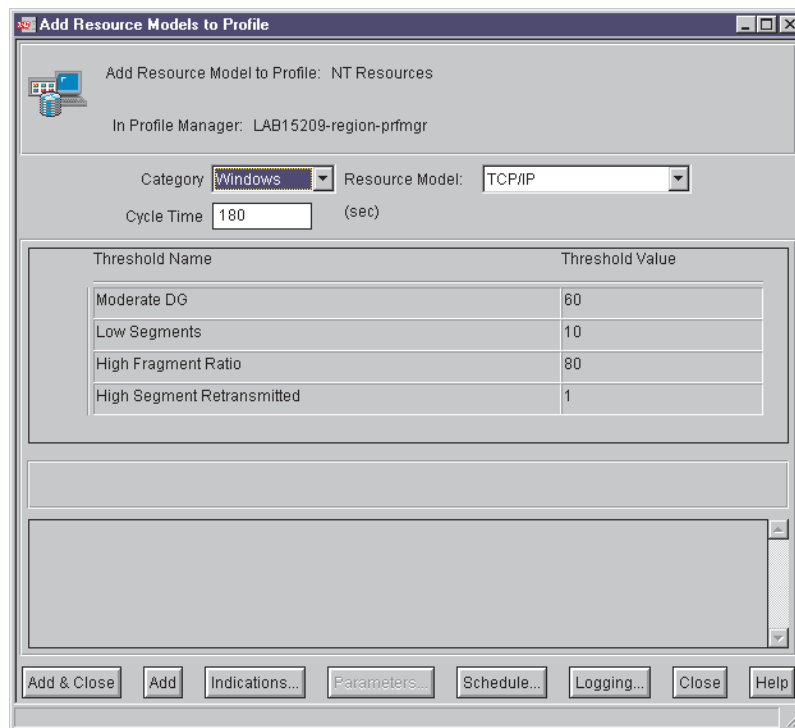
If the command is submitted with no arguments relating to resource model details, the model will be added with the defaults. Details of the command can be found in “wdmeditprf” on page 122.

## Adding a Customized Resource Model to a Profile

To customize the basic settings of a resource model and add it to a profile, perform the following steps:

1. Double-click the profile icon to which you want to add a customized resource model.
2. From the Tivoli Monitoring Profile dialog, click **Add**.

The Add Resource Model to Profile dialog opens.



3. Select the resource model category from the **Category** list. The defaults supplied with the product contain the following categories:

### UNIX – Linux

Makes only the resource models appropriate to all UNIX and Linux platforms selectable in the **Resource Model** list.

### Windows

Makes only the resource models appropriate to all Windows platforms selectable in the **Resource Model** list.

However, you should note that these categories can be modified and other categories can be added using the Tivoli Monitoring Workbench.

4. Select the required resource model from the **Resource Model** list.
5. To view all the settings of a resource model, select the resource model.  
An explanation of the resource model and its parameters, together with help on setting parameters is displayed in the **Description** text box.
6. In the **Cycle Time** text box, set the frequency with which the resource model monitors the data.
7. Change any of the **Threshold** values as required:
  - a. Select the **Threshold Name** that you want to change.  
A description of the threshold name is displayed. The currently assigned threshold value for the threshold name is also displayed above the description.
  - b. Change the currently assigned threshold value to one that matches your requirements.
  - c. Click **Apply**.  
The new value is set and displayed in the **Threshold Value** text box.
8. If required, click the appropriate buttons to modify the **Indications** (see “Customizing Thresholds and Event Rules” on page 44) and **Parameters** (for parametric resource models only - see “Customizing Parameters” on page 51) or add **Schedule** information (see “Customizing Scheduling Information” on page 52). If you want to, you can leave these items to take their default values and customize them later (see “Customizing a Resource Model”).
9. Click **Add & Close** to save any changes you have made.  
The Tivoli Monitoring Profile dialog now shows the customized resource model.

## Using the CLI to Add a Customized Resource Model to a Profile

Use the following command to add a customized resource model to a profile:

**wdmeditprf** Allows you to edit the definition of a profile, including all resource model details.

The command can be submitted to add the resource model, changing any combination of the resource model details. Details of the command can be found in “wdmeditprf” on page 122.

## Customizing a Resource Model

You can customize resource models by doing any or all of the following:

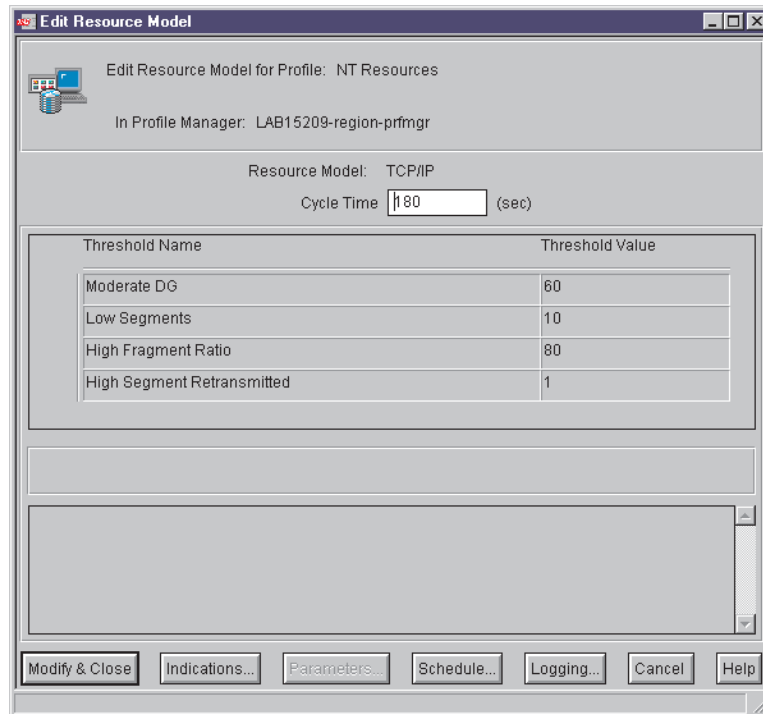
- Customizing indications and event rules, including:
  - Specifying built-in actions
  - Specifying corrective tasks
- Customizing parameters (parametric resource models only)
- Customizing scheduling information
- Customizing data logging information

## Customizing Thresholds and Event Rules

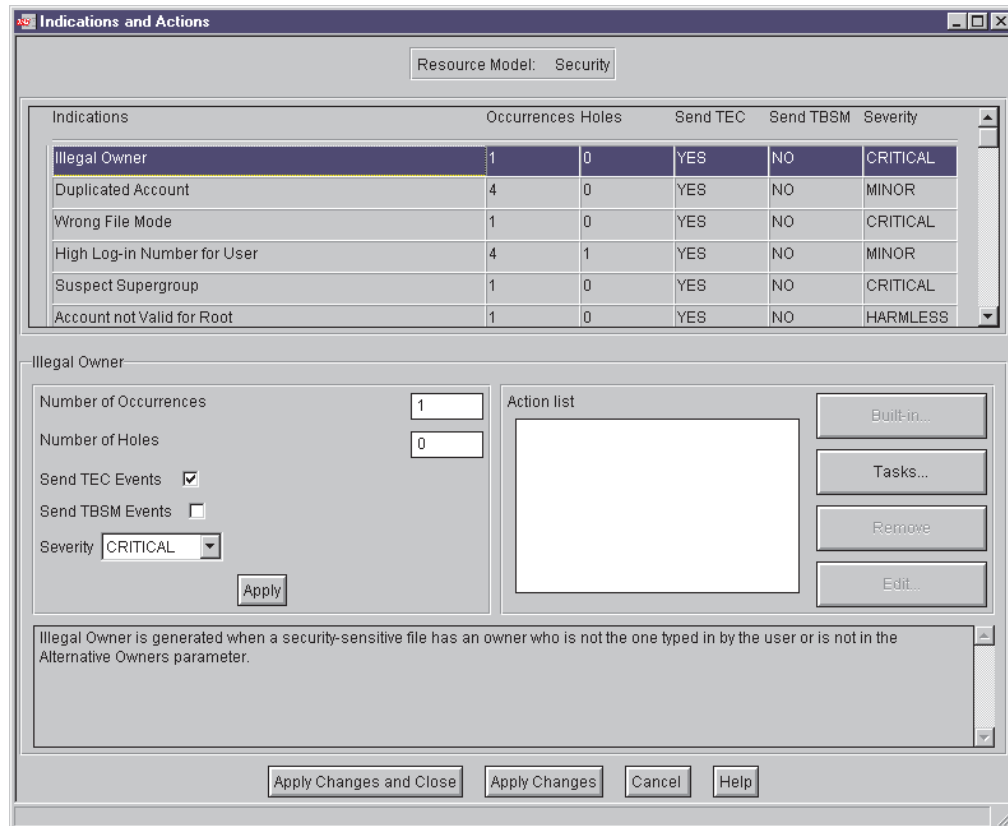
To customize the rules that determine how indications generated by a resource model are aggregated into events, perform the following steps:

1. Open the Tivoli Monitoring Profile dialog.

2. Select the resource model you want to customize.
3. Click **Edit**.  
The Edit Resource Model dialog opens.



4. Click **Indications**.  
The Indications and Actions dialog opens.



5. The indications appropriate to the selected resource model are displayed. If you want to change the values of the event associated with an indication, select the **indication**. The parameters you can change are as follows:

- **Number of Occurrences**

This number represents how many consecutive times the problem reported by the indication occurs before sending an indication. You can change this value to specify how persistent a problem is before an indication is triggered.

- **Number of Holes**

This is used to determine how many cycles that do not produce an indication can occur between cycles that do produce an indication, in order for the occurrences of the indication to still be considered consecutive. Each cycle without an indication is termed a *hole*. Thus, an event that has a value of 2 holes means that if up to 2 cycles without an indication occur between any two cycles with an indication, the cycles with an indication will be considered consecutive.

Use the value for the Number of Holes in conjunction with the Number of Occurrences parameter (see the above paragraph) and the Cycle Time (defined on the Add Resource Models to Profile dialog) to define a time window for the generation of an event. If, for example, you define Cycle Time as 10 seconds, Number of Occurrences as 5, and Number of Holes as 2, the time that must elapse before an event is created is between 50 and 130 seconds. The minimum elapsed time is the number of occurrences multiplied by the cycle time. The maximum time window assumes that the maximum number of holes will occur between each pair of occurrences, and is determined by the equation:

$$TW=CT \times (Oc + (H \times (Oc - 1)))$$

where:

- TW** Specifies the time window being calculated
- CT** Specifies the cycle time
- Oc** Specifies the number of occurrences
- H** Specifies the number of holes

- **Send TEC Events**

Select this check box to specify that if an event occurs, a Tivoli Enterprise Console event is to be sent to the specified event server (see “Identifying the Tivoli Enterprise Console Server to Tivoli Monitoring” on page 67)

- **Send to TBSM**

Select this check box to specify that if an event occurs, it is to be sent to the Tivoli Business Systems Manager’s Common Listener.

6. If you have previously deleted any of the built-in actions, the **Built-In** button will be enabled; see “Specifying Built-in Actions for an Event (Windows)”
7. If you want to specify tasks to be carried out when an event is recognized, click the **Tasks** button. See “Specifying Tasks for an Event” on page 48. If corrective tasks have already been defined you can remove them from the event definition.
8. Click **Apply Changes & Close** when you have made the required changes to the indications.

### Specifying Built-in Actions for an Event (Windows)

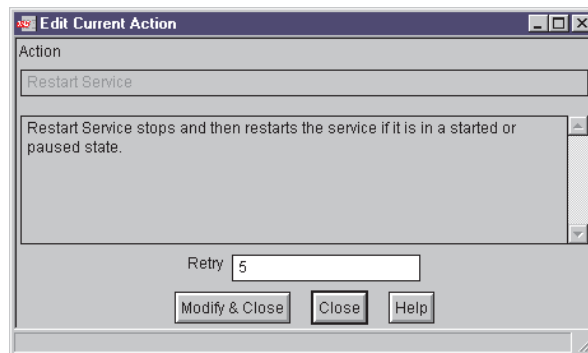
Built-in actions are available for some events for some Windows resource models, and are performed when an event is triggered. They are listed in the Action List panel of the Indications and Actions dialog.

#### Editing a Built-In Action

To change the parameters of an action, perform the following steps:

1. Select the action to edit in the Action List panel of the Indications and Actions dialog.
2. Click **Edit**.

The Edit Current Action dialog opens.



3. In the Retry field, modify the number of times the action should try to perform when an event is triggered. For example, if you set the value to three, the product will try to perform the action three times only. If you set this value to 0, the product continually tries to perform the action until it is successful.

4. Click **Close** if you want to leave the original value, or click **Modify & Close** to modify the selected Action with the value shown and close the dialog.

### Removing a Built-In Action

To remove a built-in action, select it in the **Action list** and press **Remove**.

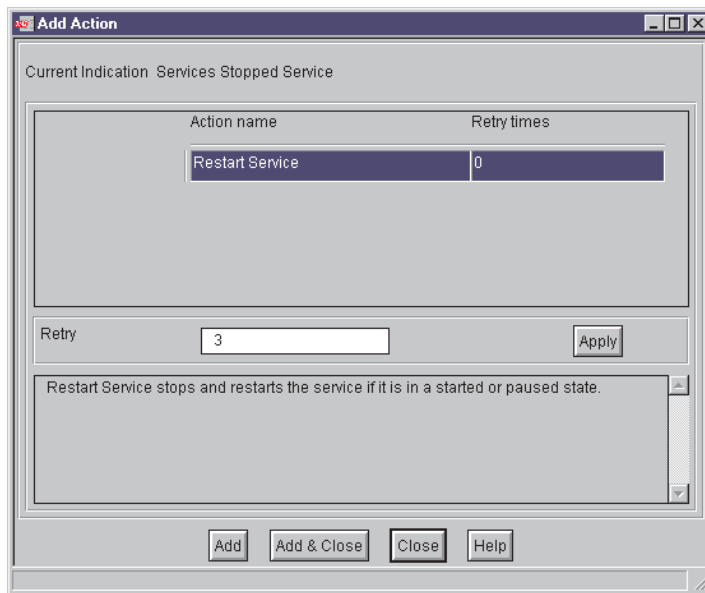
### Adding a Built-In Action

You can only add a built-in action if you have previously removed one. Only in these circumstances will the **Built-In** button in the Indications and Actions dialog become active.

To reinstate a previously removed built-in action, perform the following steps:

1. Click **Built-In** on the Indications and Actions dialog.

The Add Action dialog opens.



Select the action to run when an event occurs. The dialog shows the name of the currently selected indication. It also displays a list of selectable actions.

2. Select the required action to be run when an event occurs. In the Retry field, type the number of times the action tries to perform the action. For example, if you set the value to three, the product will try to perform the action three times only. If you set this value to 0, the product continually tries to perform the action until it is successful. Click **Apply** to save the new value.
3. Click **Add** to add the selected action with the values shown and keep the dialog open, or click **Add & Close** to add the selected Action with the values shown and close the dialog.

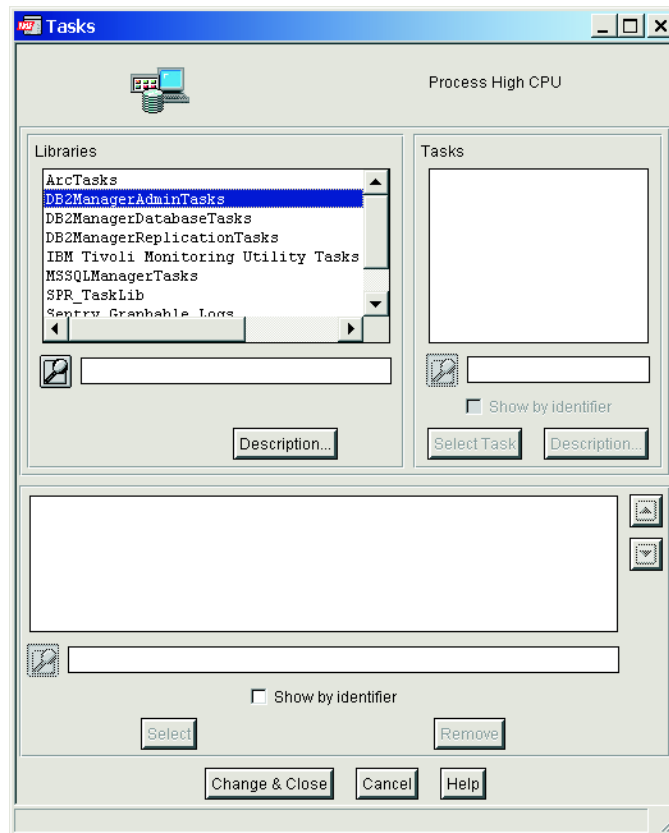
### Specifying Tasks for an Event

In the Indications and Actions dialog, perform the following steps to specify corrective or reporting tasks when an event is triggered:

1. Select an event for which a task is to be triggered.
2. Click **Tasks** to add a task that will be performed when the event is triggered.



The Tasks dialog opens.



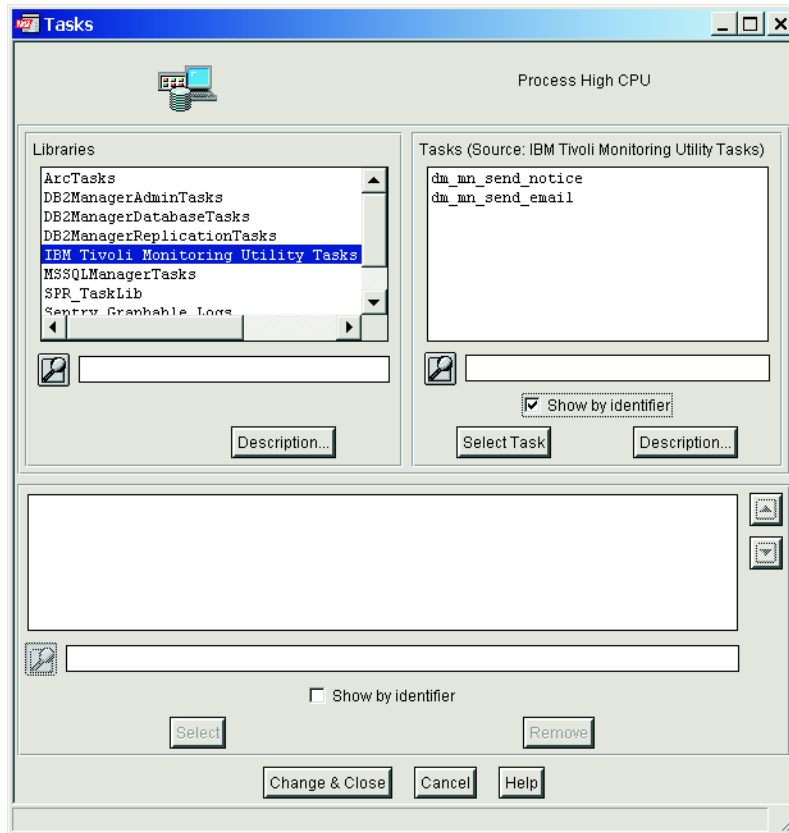
3. Double-click a task library in the **Libraries** list.  
The tasks contained in the library are displayed in the **Tasks** list.
4. Double-click a task in the **Tasks** list, then click **Change & Close** to add the task to the Action list in the Indications and Actions dialog.

### Sending an e-mail in response to an event

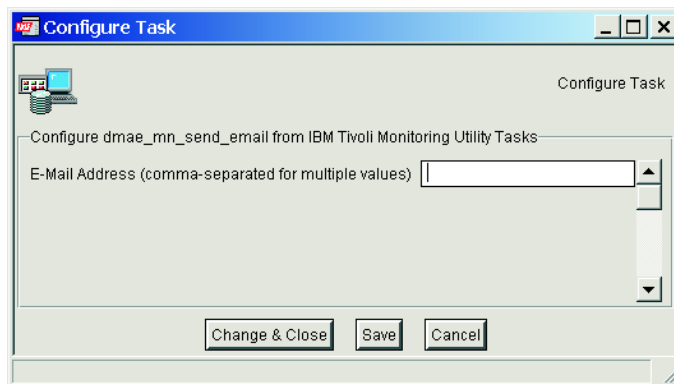
In the Tasks dialog, perform the following steps:

1. Double-click **IBM Tivoli Monitoring Utility Tasks** in the **Libraries** list.

The tasks contained in the IBM Tivoli Monitoring Utility Tasks library are displayed in the **Tasks** list.



2. Double-click **dm\_mn\_send\_email** in the **Tasks** list. The Configure Task dialog opens.



3. Specify the appropriate e-mail address. Refer to Appendix C “Configuring for SMTP E-Mail” of the *Tivoli Management Framework: Installation Guide* for information on configuring Tivoli for use with e-mail services.
4. Click **Change & Close** to add the task to the Action list.

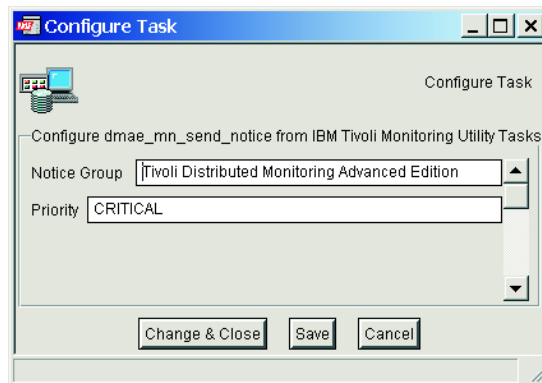
### **Sending a notice in response to an event**

In the Tasks dialog, perform the following steps:

1. Double-click **IBM Tivoli Monitoring Utility Tasks** in the **Libraries** list.

The tasks contained in the IBM Tivoli Monitoring Utility Tasks library are displayed in the **Tasks** list.

2. Double-click **dm\_mn\_send\_notice** in the **Tasks** list. The Configure Task dialog opens.



3. Specify the appropriate parameters in the Configure Task dialog (you may run the Tivoli Framework command **wlsnotif -g** to see the available Notice Groups), then click **Change & Close** to add the task to the Action list.

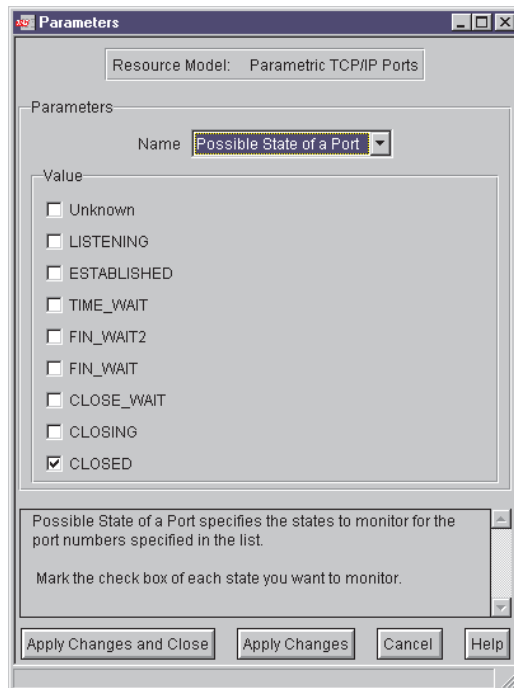
## Customizing Parameters

In parametric resource models, you can modify any parameters to customize and optimize the monitoring process. For more information about the parameters, refer to *IBM Tivoli Monitoring: Resource Model Reference*.

To customize parameters for a parametric resource model, perform the following steps:

1. In the Edit Resource Model dialog, click the **Parameters** button.

The Parameters dialog opens.



2. Select the type of parameter from the **Name** drop-down list.

3. Modify the parameter values as required.
4. When you have added or deleted the values as required, click **Apply Changes and Close** to apply and save the changes.

The Parameters dialog closes.

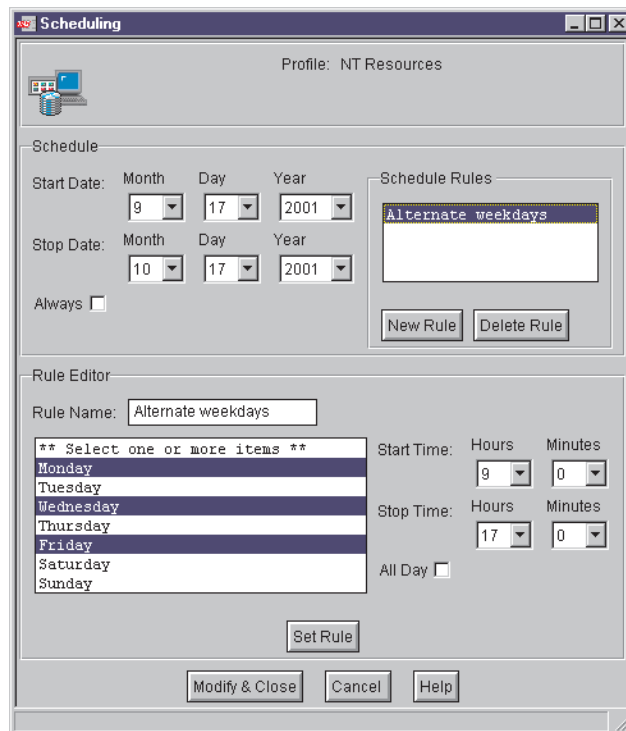
## Customizing Scheduling Information

Customizing the scheduling of the resource model is a two-stage process:

1. Set the schedule times to control the time period within which the resource model monitors resources.
2. Define the schedule rules to set the specific time intervals within the overall monitoring period when data is to be collected.

**Note:** The frequency with which data is collected is determined by the cycle time of the resource model.

In the Edit Resource Model dialog of the selected resource model click **Schedule**. The Scheduling dialog opens.



This dialog has the following group boxes:

### Schedule

For setting the data collection period. By default, all resource models are set to always collect data.

### Schedule Rules

For managing time intervals during which the resource model is active.

### Rule Editor

For creating and editing schedule rules.

## Setting the Monitoring Period

To set the monitoring period, perform the following steps:

1. In the Schedule group box, deselect **Always**.
2. Set a start and stop date to define the monitoring period.
3. If required, add one or more schedule rules that determine time periods on selected days during which monitoring will take place.

For information about creating schedule rules, see “Creating Schedule Rules”.

## Creating Schedule Rules

The schedule controls the overall period for monitoring a resource and the timing windows in which resource model data is to be collected. The overall collection period can be divided into active and inactive intervals by applying one or more rules.

To create a schedule rule, perform the following steps:

1. In the Scheduling dialog, click the **New Rule** button.
2. In the **Rule Editor** group box, type a name for the rule.
3. Specify the day or days on which you want the collections to be active during the collection period, by selecting one or more items in the list of weekdays.

**Tip:** Use the **Shift** key or **Ctrl** key as necessary, to select more than one day from the list.

4. Set start and stop times for the collection activity or select the **All Day** check box. Times are always interpreted as local time where the endpoint engine is running, so that, for example, setting a time interval of 08:00 to 13:00 will ensure that monitoring takes place between those times in all time zones to which you distribute the profile.
5. Click **Set Rule**.

Your new rule is added to the **Schedule Rules** list.

You can create another rule for addition to the list or delete an existing rule from the list.

**Tips:** To display the details of any rule, select the rule in the **Schedule Rules** list. Its settings are displayed in the **Rule Editor** group box.

If the Schedule Rules list contains more than one schedule rule, all the time intervals are respected, and rules are combined by adding together the time periods they define. For example, if you specify a rule that requests monitoring between 8:00 and 14:00 every day and another that requests all-day monitoring on Fridays, the sum of the two rules gives all-day monitoring only on Fridays, and monitoring between 8:00 and 14:00 on all other days. If the second rule instead requested monitoring from 12:00 to 18:00 on Fridays, the sum of the rules would give monitoring between 8:00 and 18:00 on Fridays and between 08:00 and 14:00 on all other days.

6. To save your rule, click **Add & Close**.

The Scheduling dialog closes.

## Customizing Data Logging Information

Using this feature, you can log data collected by a resource model and write it in a local database. Then you can view it through the Web Health Console. In the database, you can choose to store one of the following types of data:

**Raw data**

Data is written exactly as it is collected by the resource model. All the monitored values are collected and copied in the database.

**Aggregated data**

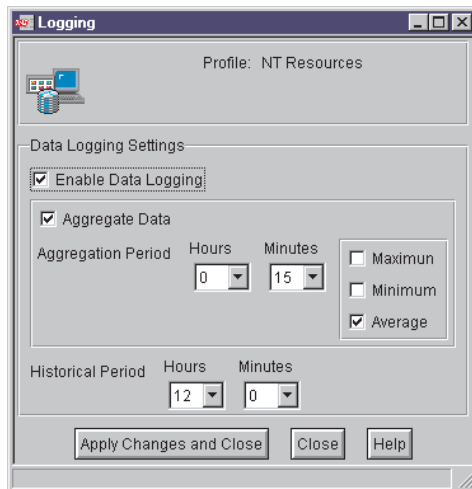
Data is collected and aggregated at fixed intervals you define (**Aggregation Period**). Then only the aggregated values are written in the database. The aggregated data can be calculated on the basis of one or more of the following options:

- Maximum
- Minimum
- Average

By default, data logging is not enabled.

To add data logging information to a resource model, perform the following steps:

1. In the Edit Resource Model dialog, click **Logging**.  
The Logging dialog opens.



2. In the **Data Logging Settings** group box, select the **Enable Data Logging** check box. This enables logging of all raw data collected by the resource model to a database.
3. To specify the aggregation rule to be applied to the data before it is written to the database, perform the following steps:
  - a. To specify the period for which data is to be aggregated, set **Hours** and **Minutes** of the **Aggregation Period** to the required values.
  - b. Select one or more of the following functions to be performed on the numerical data collected during the aggregation period before it is written to the database:
    - Maximum**      Calculates and logs the peak value in each aggregation period
    - Minimum**      Calculates and logs the lowest value in each aggregation period
    - Average**      Calculates and logs the average of all values in each aggregation period
4. To specify the period for which data is to be stored in the database, set **Hours** and **Minutes** of the **Historical Period** to the required values.

- To save the changes, click **Apply Changes and Close**.

The Logging dialog closes.

## Using the CLI to Customize a Resource Model for a Profile

Use the following command to customize a resource model for a profile:

**wdmeditprf** Allows you to edit the definition of a profile, including all resource model details.

The command can be submitted to edit the resource model for a selected profile, changing any combination of the resource model details. Details of the command can be found in “wdmeditprf” on page 122.

## Setting Distribution Options for a Profile

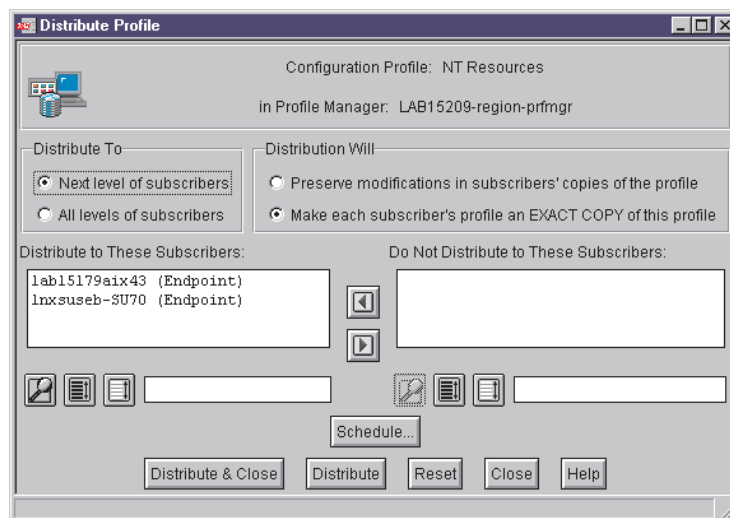
To set the distribution options for the profile, perform the following steps:

- In the Profile Manager dialog, open a profile to be distributed.

The Tivoli Monitoring Profile dialog opens.

- Click **Profile → Distribute**.

The Distribute Profile dialog opens.



- Select one of the **Distribute To** options based on the following descriptions:

- **Next level of subscribers**

Distributes the profile only to the subscribers named in the Distribute To these Subscribers scrolling list of the Distribute Profile dialog.

This selection distributes the profile only to the subscribers of the profile manager. It does not distribute to lower-level subscribers. If a profile manager with subscribers resides at the next lower level, you may need to perform the distribution process from profile managers at more than one level to reach all the profile endpoints.

- **All levels of subscribers**

Distributes the profile to all subscribers in the hierarchy. An example follows to illustrate the difference between distributing to the two levels of subscribers. You have a profile hierarchy in which a dataless profile manager is subscribed to a profile manager, and the dataless profile manager has an endpoint subscribed. If you

distribute to the next level of subscribers, the profile is distributed only to the dataless profile manager. If you distribute to all levels of subscribers, the profile is distributed to the dataless profile manager, and to the endpoint.

Select this option if you want to distribute a profile in which your endpoint is the only subscriber.

- Select the following Distribution Will option: Make subscribers profile an EXACT COPY of this profile  
Overwrites the subscribers profile with an exact copy of the profile being distributed. Do not distribute a profile to a Tivoli Enterprise™ endpoint with the preserve modifications option set. You must always use the make exact copy option.
- Select subscribers to receive the profile by choosing them from the Don't Distribute To These Subscribers scrolling list and moving them to the Distribute To These Subscribers scrolling list.

**Note:** Make sure that each of the subscribers in the **Distribute To These Subscribers** scrolling list is either a profile manager or a supported TMA endpoint. Tivoli Monitoring does not support other types of endpoints.

4. Click **Distribute & Close** to distribute the profile immediately and close the Distribute Profile dialog, saving the settings you have made, and return to the Tivoli Monitoring Profile dialog.

You can also click **Distribute** to distribute the profile immediately, saving the settings you have made, but leaving the Distribute Profile dialog open.

Alternatively, you can click **Schedule** to schedule the distribution of the profile with the Tivoli Scheduler. For details about using the Tivoli Scheduler, refer to the *Tivoli Management Framework User's Guide*.

## Profile Distribution using MDist2

Tivoli Monitoring uses Multiplexed Distribution (MDist2) to perform asynchronous profile data transfers through a hierarchy of repeaters. MDist2 returns a sequence of responses containing the distribution status from each endpoint to the application initiating the distribution. These responses are sent back to Tivoli Monitoring in a log file, on the MDist2 command line, or through the MDist2 GUI.

Tivoli Monitoring makes full use of these MDist2 functions:

- Asynchronous delivery
- Assured delivery
- Check-point and restart
- Data depoting

### Asynchronous Delivery

MDist2 uses an asynchronous interface to applications, which means that when Tivoli Monitoring submits a distribution request, it immediately gets back a distribution identifier and confirmation that the distribution is in progress. Only when the distribution is complete does MDist2 send the final distribution status for each endpoint through the callback function. This means that Tivoli Monitoring does not need to be kept waiting until the distribution is complete for every endpoint.



**Assured Delivery**

The distribution of Tivoli Monitoring profiles is assured even when there are network interruptions, power-offs on the machines, or disconnected endpoints. MDist2 is able to continue the distribution from where it reached before the network interruption. It also retries to establish connections to endpoints until it is successful or the distribution time expires.

**Check-point and Restart**

A data stream that has been interrupted can be resumed from the last successful check-point. This means that it is not necessary to resend all the Tivoli Monitoring Profile data when the distribution is resumed, but only the data that had not yet been sent when the interruption occurred.

**Data Depoting**

MDist2 allows the storage of distribution segments at a depot close to the endpoint, and for the distribution to be submitted to the endpoints from that depot. The data transferred to the endpoints is thus taken from the depot, and not from the source host. This considerably reduce network traffic and speed up the distribution.

**Rerunning the Failed Distributions**

When a distribution fails, Tivoli Monitoring creates a *profile manager* containing the endpoint subscribers that failed. To see the profile managers, go to the Region dialog and select **View** → **Refresh**.

If the distribution failed because of error `Bad_Interpreter`, then the profile manager name is derived as follows:

```
OriginalProfileName_Push_Failed_Bad_Interpreter
```

where:

*OriginalProfileName*

This is the name of the profile that you were distributing when the error occurred.

Error message **AMW089E** might be displayed at this point, indicating that the resource model type is not compatible with the endpoint operating system. For example, you might have distributed a Windows resource model to a UNIX-T endpoint, or vice versa. See Appendix B, “Error, Warning, and Information Messages” for more details.

If the distribution fails because of any other error, then the profile manager name is derived as follows:

```
OriginalProfileName_Distribution_Failed
```

Provided that the profile manager that you used for the original distribution was created without checking the “dataless endpoint mode” option, you can use these profile managers to redistribute the profile to the failed endpoints when you have fixed the problem that caused the original failure. To do this, just subscribe the profile managers that contain the failed endpoints to the profile manager that contained the original profile. You can then distribute the original profile to the failed endpoints by selecting these profile managers as the target for the distribution. The profile managers can also be edited to delete an endpoint from a group of failed endpoints before retrying the distribution.

If the redistribution is successful, the profile manager relating to the error condition is deleted. If the redistribution is wholly or partially unsuccessful, the profile manager is

retained, containing only the subscribers that were unsuccessful in the redistribution. You should continue the process of correcting the errors and retrying the distribution until all errors have been fixed.

If any of these problems persist and cannot be resolved, call Tivoli Customer Support.

## Managing Profiles and Resource Models at Endpoints

Tivoli Monitoring provides you with a set of commands to manage profiles and resource models after they have been distributed to endpoints:

- wdmcmd** Stops or restarts Tivoli Monitoring on one or more endpoints from a gateway or server.
- wdmdistrib** Distributes a profile to one or more subscribers.
- wdmeng** Stops or starts profiles or resource models at endpoints; also deletes profiles at endpoints.
- wdmlseng** Returns a list and the status of all resource models that have been distributed on a specified endpoint.
- wdmtrceng** Sets the trace parameters of the Tivoli Monitoring engine at the endpoint.

Details of all Tivoli Monitoring commands can be found in Appendix A, “Commands” on page 107; details of Tivoli Management Framework commands can be found in the *Tivoli Management Framework: Reference Manual*.

## Managing Tivoli Monitoring at Gateways

Tivoli Monitoring provides you with a command to manage the product on gateways:

- wdmmn** Stops or starts selected Tivoli Monitoring processes on one or all gateways.

Commands to manage the Tivoli Business Systems Manager Adapter component on gateways are described in “Managing the Tivoli Business Systems Adapter from the CLI” on page 79.

# 4

## Heartbeat Function

---

Tivoli Monitoring includes a heartbeat function, which monitors the basic signs of life at endpoints attached to the gateway at which it is enabled.

The heartbeat monitor regularly monitors the endpoints, checking if they are running correctly. Events may be sent to the Tivoli Business Systems Manager (provided that the Tivoli Business Systems Manager Adapter component is installed at the gateway), the Tivoli Enterprise Console (TME<sup>®</sup> events only) or the Tivoli Monitoring Notice Group.

### Heartbeat Statuses

The function can register the following statuses for an endpoint in its cache. They are divided into two groups, depending on whether an Information or an Error event is sent to the monitors:

#### Statuses for which an Information event is sent

- Alive
- Heartbeat has been stopped

#### Statuses for which an Error event is sent

- Resource model in error status
- Tivoli Monitoring engine has been stopped
- Endpoint not available in network

### Installing and Configuring the Heartbeat

The heartbeat function is an integral part of Tivoli Monitoring as installed on a gateway, and needs no specific installation. However, you can configure the recipients of heartbeat information using the **wdmconfig** command (see “wdmconfig” on page 111). This command lets you amend various aspects of the configuration of the Tivoli Monitoring components at a gateway. In particular, for the heartbeat, you can change the values of the following configuration keys:

#### **heartbeat.send\_events\_to\_tbsm**

Set this to **true** if you want to send heartbeat events to the Tivoli Business Systems Manager (the Tivoli Business Systems Manager Adapter must be installed and configured at the gateway); otherwise leave as the default value of **false**.

#### **heartbeat.send\_events\_to\_tec**

Set this to **true** if you want to send heartbeat events to the hostname of the Tivoli Enterprise Console server; otherwise leave as the default value of **false**.

**heartbeat.tec\_server**

If you have set **heartbeat.send\_events\_to\_tec** to **true**, enter here the name of the Tivoli Enterprise Console server (for example: *EventServer*).

**heartbeat.send\_events\_to\_notice**

Set this to **false** if you do not want to send heartbeat events to the notice group *Tivoli Distributed Monitoring (Advanced Edition)*; otherwise leave as the default value of **true**.

**heartbeat.reboot\_engine\_if\_down**

Set this to **true** if you want to enable the recovery function of the agent where Tivoli Monitoring is running; otherwise leave as the default value of **false**. When the configuration key is set to **true**, the recovery function restarts the engine whenever the engine stops running.

All these configuration keys and their values are contained in file: `$DBDIR/dmm1/.config`

This file can be read but it must not be modified manually. If you wish to modify it, use the **wdmconfig** command (note that the applied changes might not show).

## Controlling the Heartbeat

The heartbeat function is controlled by issuing one of the following commands from the Tivoli Console:

**wdmheartbeat**

Stops or starts the ‘heartbeat’ monitoring of endpoints, changes its frequency and queries the status of the heartbeat processor (see “wdmheartbeat” on page 131, for more details).

**wdmmn**

Stops the heartbeat engine on one or all managed nodes/gateways (see “wdmmn” on page 139, for more details).

Figure 4 on page 61 shows the data flow for the heartbeat control.

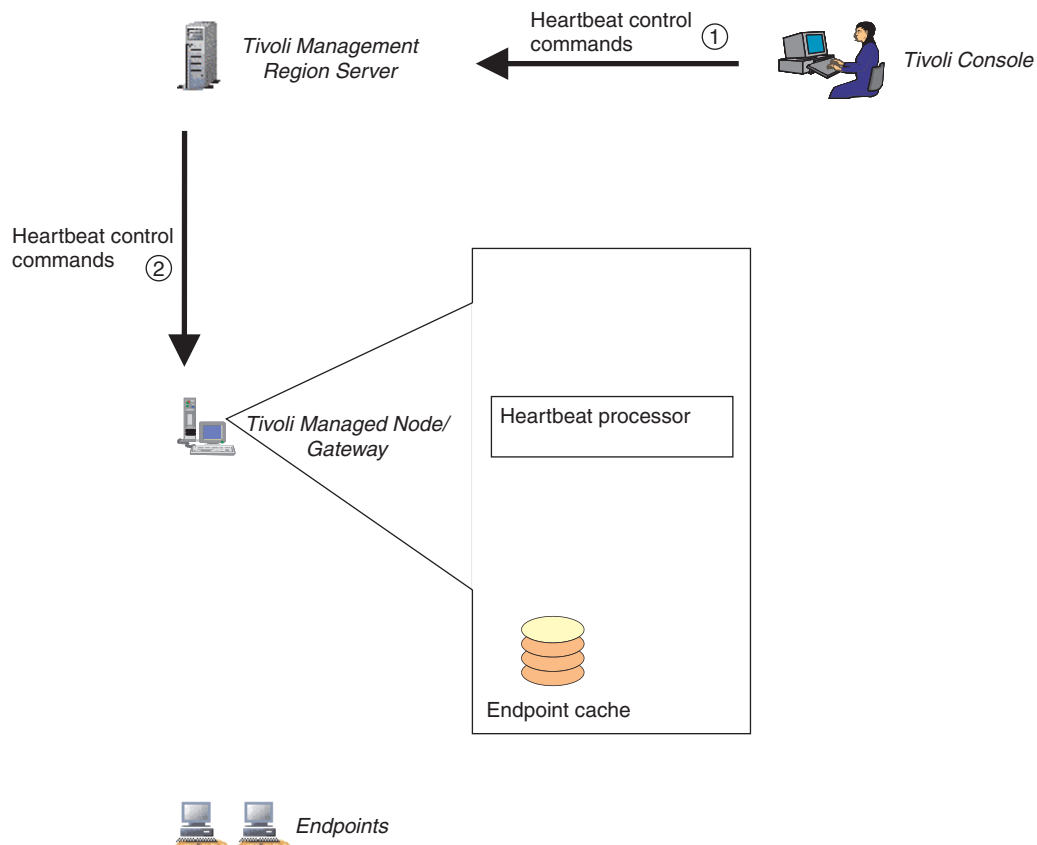


Figure 4. Data flow for heartbeat control

The heartbeat control commands are routed to the gateway (steps 1 and 2), where they are processed by the heartbeat processor.

**Note:** The heartbeat processor is protected from closures of the *oserv* such that if the *oserv* is closed and then re-opened, the heartbeat processor restarts at the same frequency it was using when the *oserv* was closed.

## Heartbeat Data Flows

The heartbeat function has these principle activities:

- Endpoint Registration
- Heartbeat Monitoring
- Viewing the Endpoint Cache

### Endpoint Registration

When Tivoli Monitoring pushes a resource model to an endpoint for the first time, or restarts the Tivoli Monitoring engine at an endpoint, the information in the endpoint cache is updated when the gateway receives a message from the endpoint saying that its engine has been started. Figure 5 on page 62 illustrates the data flow.

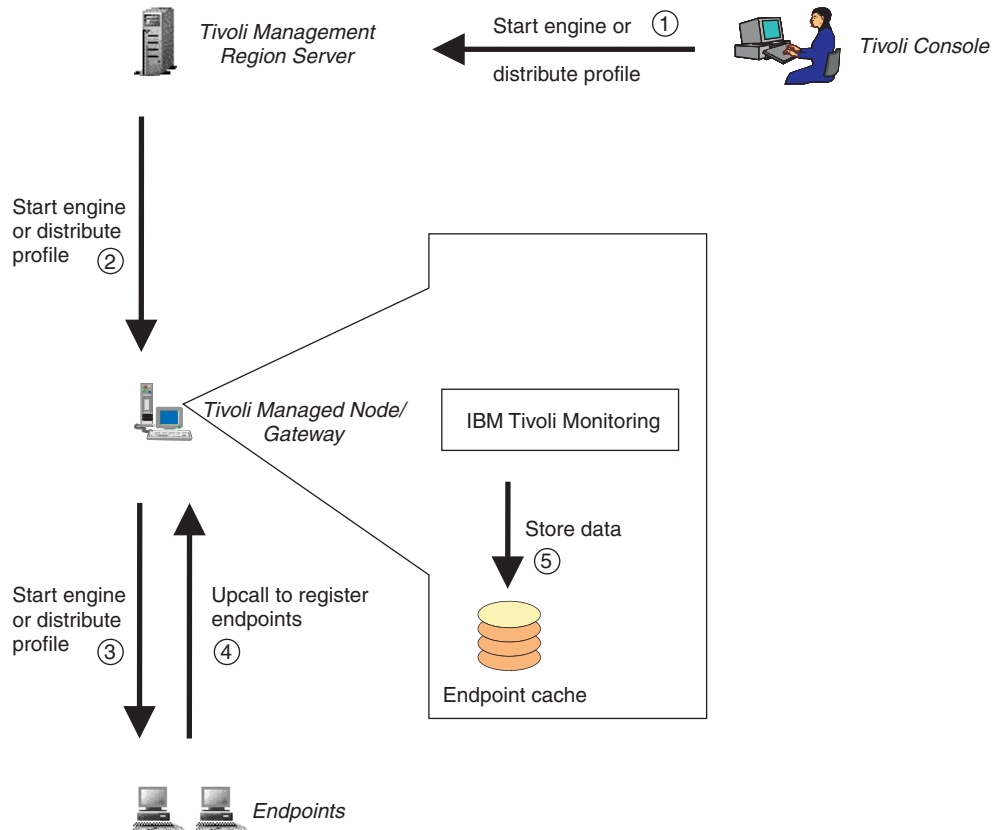


Figure 5. Data flow for endpoint registration

The figure shows a profile being distributed (pushed) from the Tivoli Desktop, or a command being issued to start an endpoint engine (steps 1, 2 and 3). The endpoint sends an upcall to the gateway (step 4), which registers the presence of the endpoint in the endpoint cache (step 5).

**Note:** Every 18 minutes the endpoint sends an upcall to the gateway. This behaviour cannot be modified.

## Heartbeat Monitoring

Figure 6 shows the data flow for the heartbeat function at a gateway in a Tivoli management region, with all possible monitors configured to receive the heartbeat events.

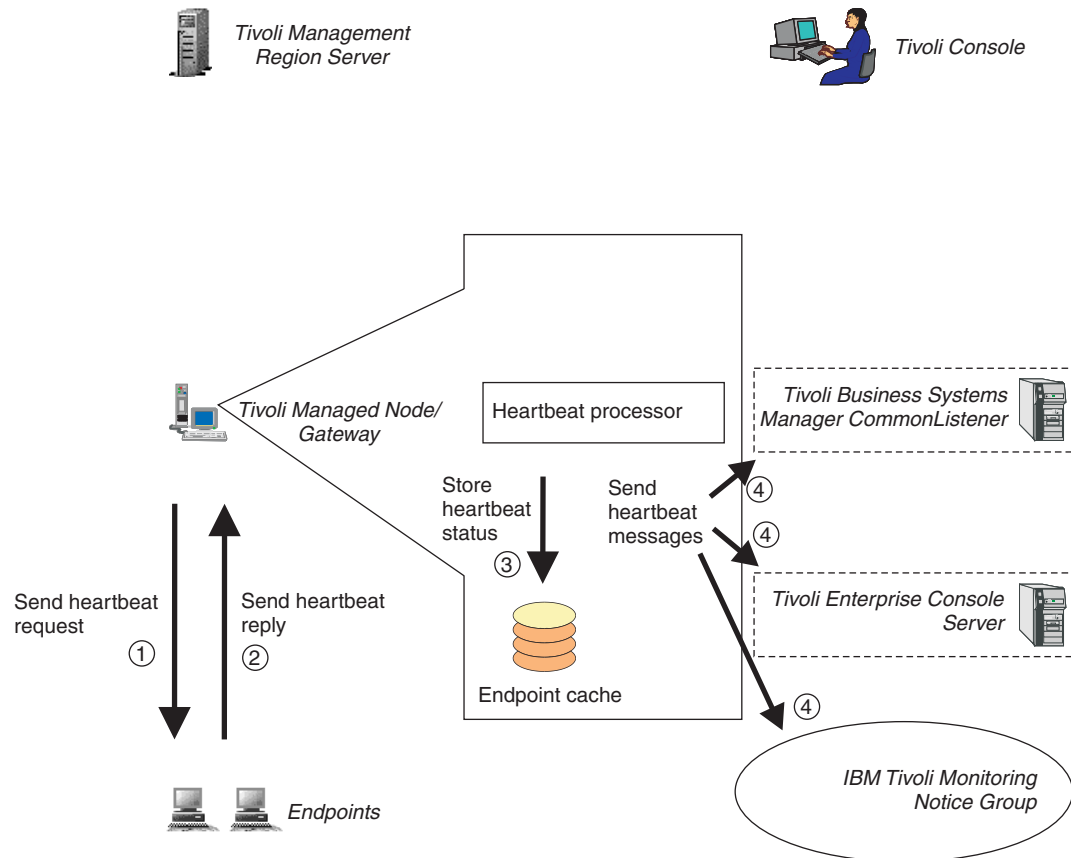


Figure 6. Data flow for heartbeat function at gateway

The gateway issues periodic heartbeat requests to all attached endpoints (step 1). The data that is returned by the endpoints (step 2) is stored in the endpoint cache (step 3) and sent to whichever of the monitors is configured to receive it (step 4).

## Viewing the Endpoint Cache

In addition to the monitors described in the previous section, it is also possible to view the heartbeat information in the endpoint cache, using the **wdmmngcache** command.

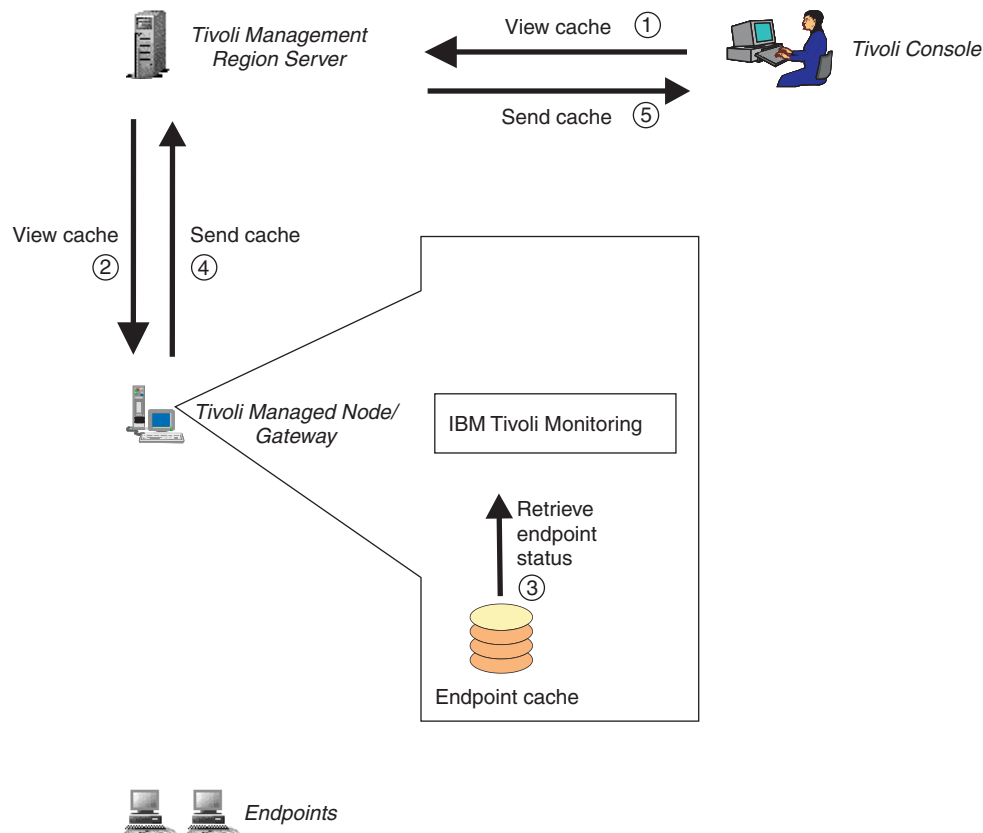


Figure 7. Data flow using **wdmmngcache** command

This diagram shows the data flow when a **wdmmngcache** command is issued at the Tivoli Desktop. A request is sent to the gateway (steps 1 and 2), which interrogates the cache (step 3) and sends the information back to the desktop (steps 4 and 5). There is no interaction with the endpoints in this data flow. Data can also be deleted from the cache using this command.

The **wdmmngcache** command has two output formats. If the **-v** option is not used, a line is provided for each endpoint, showing the heartbeat status, as follows:

```
Processing ManagedNode mpulp...
Processing ManagedNode dmw2k2...
Endpoint | Status
-----+-----
mpulp-ep | Alive
dmw2k2-ep | Alive
```

If the **-v** option is used, the command provides information about the discovery status of the endpoints, as follows:

```
Processing ManagedNode mcrudele...
Processing ManagedNode dmw2k2...
Warning: DM_Adv_Edition feature not installed on the Managed Node 'boccaccio'. Skipping...
Endpoint | HB status | TBSM status
```



-----+-----+		
dmw2k2-ep	DMEngineOff	Not discovered
mcrudele-ep	DMEngineOff	Not discovered



# 5

## Integration with Tivoli Enterprise Console Server

---

This chapter describes how to integrate Tivoli Monitoring with the Tivoli Enterprise Console server to enable Tivoli Monitoring events to be used by the Tivoli Enterprise Console server.

### Overview

Before you can send Tivoli Monitoring events to a Tivoli Enterprise Console server, you must:

- Identify the Tivoli Enterprise Console to Tivoli Monitoring
- Enable monitoring of product events and heartbeat messages on the Tivoli Enterprise Console server

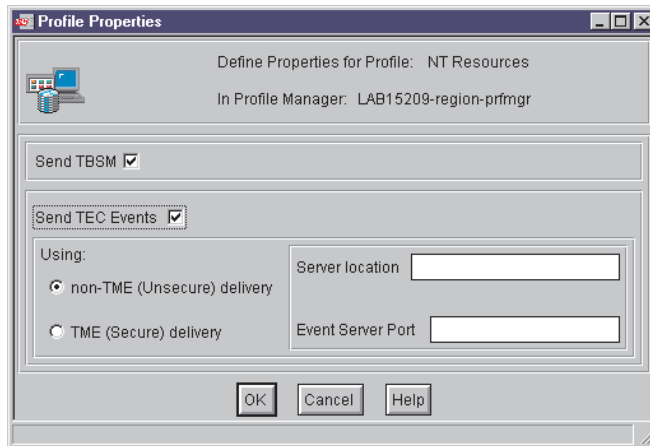
### Identifying the Tivoli Enterprise Console Server to Tivoli Monitoring

For each profile you create, you must identify the Tivoli Enterprise Console server that you want to use to monitor the Tivoli Monitoring events that it defines. You can also choose to send non-TME (unsecure) events to the Availability Intermediate Manager (AIM).

To set the identity of the Tivoli Enterprise Console server for all resource models in a profile, perform the following steps:

1. Open the Tivoli Monitoring Profile dialog.
2. Select **Edit → Properties**.

The Properties dialog opens and displays the name of the profile for which the server identity is to be set, and the profile manager in which the profile is located. It also displays the option **Send to TBSM**.



3. Check the **Send TEC Events** check box to send events to the Tivoli Enterprise Console server or to AIM (only for unsecure delivery events).

**Note:** If you do not select this check box, no events are sent to the Tivoli Enterprise Console server, even if you selected the **Send TEC Events** check box in the Indications and Actions dialog.

One this is checked, the **Using** section of the dialog is enabled. The **Using** section is disabled if the **Send TEC Events** check box is not selected. You *must* select a delivery mode, the choices are:

- **non TME (Unsecure) delivery**

You must specify the Event Server Port and Server Location where the event will be sent. You must specify the Event Server Port if the Tivoli Enterprise Console server or AIM runs on Windows NT (the default value is 5529), otherwise you must set this value to 0.

- **TME (Secure) delivery**

Select the Tivoli Enterprise Console server from the **Choose TEC Server** list. This list contains all Tivoli Enterprise Console servers (if any) in all the regions. The Enterprise Console Adapter Configuration Facility (ACF) must be installed on the gateway if you choose the TME (Secure) delivery mode.

4. Click **OK** to save the properties.

The Tivoli Monitoring Profile dialog is redisplayed.

## Enabling Monitoring of Events and Heartbeat Messages on the Tivoli Enterprise Console Server

To enable monitoring of Tivoli Monitoring events on a Tivoli Enterprise Console server, you need to import the Tivoli Monitoring BAROC files into the rule base used by the Tivoli Enterprise Console server.

To accomplish this you need to either follow the procedure detailed below, or run the Tivoli-provided script that carries out the procedure steps for you:

### Running the script

A script has been provided to enable you to update the rule base in a semi-automatic way. The script is named `dmae_tec_inst.sh`, and is stored in the `$BINDIR/TMNT_TEC` directory.

The script must be run on the Tivoli Enterprise Console server. It takes as an argument the name of the *rule\_base* of the Tivoli Enterprise Console server where the BAROC rules are to be stored. You can also optionally specify the parameter `-restartsvr` to restart the server at the end of the script.

#### Following the procedure

If you do not want to run the script, you can follow the individual steps that the script performs, which are detailed in the procedure set out below. For full information about performing each step of the procedure, refer to the *Tivoli Enterprise Console Rule Builder's Guide*.

Perform the following steps:

1. Select an existing rule base or create a new rule base to contain the Tivoli Monitoring BAROC files.
2. Import the required BAROC files from the `$BINDIR/TMNT_TEC` directory into the rule base.

It is important that you import the files in the following order:

- a. Import the `Tmw2k.baroc` file. For more information about it, see the *Tivoli Enterprise Console Rule Builder's Guide*.
  - b. Import the BAROC files for all the resource models whose events you want to send to Tivoli Enterprise Console server.
  - c. Import the `hb_events.baroc` file to enable heartbeat messages.
3. Import the required heartbeat rules file from the `$BINDIR/TMNT_TEC` directory into the rule base to enable the support of heartbeat messages. The file is called `hb_events.rls`.
  4. Import the required clearing event rules file from the `$BINDIR/TMNT_TEC` directory into the rule base to enable clearing events to close the error events to which they relate. The file is called `dmae_events.rls`. If you omit this step, the Tivoli Enterprise Console server will not close related error events, and the clearing event will be displayed.
  5. Compile and load the rule base.
  6. Stop and restart the Tivoli Enterprise Console server.

The Tivoli Enterprise Console server is now ready to receive Tivoli Monitoring events from the monitoring sources whose corresponding BAROC files you have imported into the active rule base.

To see the events sent by Tivoli Monitoring, from the Tivoli Enterprise Console main dialog, click the **All** icon. See the *Tivoli Enterprise Console: User's Guide* for more details.

**Note:** The BAROC available with Tivoli Monitoring 5.1 can also be used with Tivoli Distributed Monitoring (Advanced Edition) 4.1 or with Tivoli Distributed Monitoring for Windows 3.7 Patch 3.

## Enable Monitoring of the Events on the Availability Intermediate Manager

To enable monitoring of Tivoli Monitoring events on Availability Intermediate Manager (AIM), follow the same procedure described for Tivoli Enterprise Console server.

Only non-TME (unsecure) delivery events can be sent to AIM.

### Slots of the Tivoli Enterprise Console Classes

This section describes the slots (fields) related to the TMW\_Event and HeartBeat\_Event classes.

#### Slots in the TMW\_Event Class

The following table lists all the slots pertaining to the TMW\_Event class. All the event classes of the Tivoli Monitoring resource models inherit from the TMW\_Event class.

Table 9. Slots of the TMW\_Event Class

Slot	Slot Description	Slot Value
source	Event adapter known to the event server	TMNT
sob_source	A more detailed description of the source	N/A
origin	Protocol address or host name of the source system	<i>Endpoint_IP_address</i>
sub_origin	A more detailed description of the origin	N/A
hostname	The name of the system where the event occurred	<i>endpoint_label</i>
adapter_host	The name of the system where the event adapter that reported the event is located	<i>endpoint_label</i>
severity	Event severity	<i>event_severity</i>
msg	A descriptive message of the event	<i>message</i>
msg_catalog	The name of the message catalog to be used when displaying internationalized messages	none
msg_index	The offset to the event message catalog	0
repeat_count	Repeat counter	0
modelname	The name of the resource model	<i>resource_model_name</i>
profilename	The name of the profile	<i>profile_name</i>
eventid	A unique event ID	<i>eventid</i>
event_key	A list of key information	<i>key_list</i>

## Slots in the HeartBeat\_Event Class

The following table lists all the slots pertaining to the HeartBeat\_Event class. All the Tivoli Enterprise Console events that are sent by the heartbeat inherit from the HeartBeat\_Event class.

*Table 10. Slots of the HeartBeat\_Event Class*

Slot	Slot Description	Slot Value
source	Event adapter known to the event server	TMNT
sob_source	A more detailed description of the source	N/A
origin	Protocol address or host name of the source system	<i>Endpoint_IP_address</i>
sub_origin	A more detailed description of the origin	N/A
hostname	The name of the system where the event occurred	<i>endpoint_label</i>
adapter_host	The name of the system where the event adapter that reported the event is located	<i>managednode_label</i>
severity	Event severity	<i>event_severity</i>
msg	A descriptive message of the event	<i>message</i>
msg_catalog	The name of the message catalog to be used when displaying internationalized messages	none
msg_index	The offset to the event message catalog	0
repeat_count	Repeat counter	0





# 6

## Integration with Tivoli Business Systems Manager

---

This chapter describes how to integrate Tivoli Monitoring with Tivoli Business Systems Manager, so that Tivoli Monitoring events can be sent to and analyzed by Tivoli Business Systems Manager. The following topics are covered:

- Installing Tivoli Business Systems Manager Adapter
- Configuring Tivoli Business Systems Manager Adapter
- Using the Tivoli Business Systems Manager Adapter
- Enabling and Disabling Tivoli Business Systems Manager Integration
- Managing the Tivoli Business Systems Adapter from the CLI

### Overview

Tivoli Business Systems Manager is an application that enables you to perform distributed management, OS/390<sup>®</sup> management, or both. Even when a business system spans multiple platforms, Tivoli Business Systems Manager enables you to graphically monitor and control interconnected business components and operating system resources. A business component and its resources are referred to as a Line of Business (LOB). Using the LOB concept Tivoli Business Systems Manager helps you plan, define, and control your business system. Tivoli Business Systems Manager, together with other Tivoli management components, helps you manage the dependencies between business components and their underlying infrastructure.

The management facilities in Tivoli Business Systems Manager include the management of events generated by Tivoli Monitoring, so when you are using Tivoli Business Systems Manager, you can direct some or all of the events generated by Tivoli Monitoring to the Tivoli Business Systems Manager interface, letting you manage them in the same way as other aspects of your business systems.

Tivoli Monitoring events are sent to Tivoli Business Systems Manager by the Tivoli Business Systems Manager adapter, a Tivoli Monitoring component that should be installed on all gateways in a region.

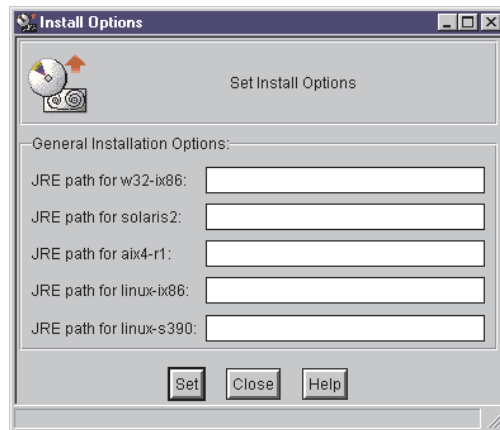
The adapter feeds Tivoli Business Systems Manager with information about resources managed by Tivoli Monitoring by means of a bulk/delta discovery. For discovered systems it can then supply heartbeat information and Tivoli Monitoring events.

### Installing Tivoli Business Systems Manager Adapter

The Tivoli Business Systems Manager adapter is installed as a product in the same way as Tivoli Monitoring is installed; see “Installing/Upgrading the Product” on page 23.

**Note:** The Tivoli Business Systems Manager Adapter cannot be installed on HPUX gateways.

During the installation, you must define the installation path of Java Runtime Environment, 1.3.0, to the Tivoli Business Systems Manager Adapter. The Install Options dialog opens.



1. Enter the following details:

### JRE path for ...

For each platform, type on which you want to install the adapter, enter the complete path of the root installation directory of Java Runtime Environment, 1.3.0 (not including the directory `/bin`).

**Note:** In a Windows NT workstation, if the install target path contains a directory with spaces in the name, the directory name must be specified between single quotes, as in this example:

```
D:\'Program Files'\jre
```

2. Press **Set** to save the paths.

If not all of your platforms of the same type have the same root directory for Java Runtime Environment, 1.3.0, the following is proposed:

1. Enter the most common path for the platform in question.
2. For each gateway that does not use the supplied path, use the **wdmconfig** command to amend the **tbsma.jre\_root** variable to the correct path.

## Configuring Tivoli Business Systems Manager Adapter

The Tivoli Business Systems Manager adapter is configured by using the command **wdmconfig**. You can address this command to one, a list, or all gateways in the region, and it allows you to supply any number of key/value pairs as parameters. The key/value pairs that you will need to configure are as follows:

### **adapter.trace.enable**

Set this to **true** if you want to write to the file identified in `trace.filename`, all trace messages regarding the operations of the adapter. The default is **false**.

### **adapter.trace.level**

Set this to **low**, **medium** or **high**, according to the level of details you require if you have enabled adapter trace messages. The default is **low**.

**adapter.working.dir**

Working directory that will be used by the adapter. The default, which you are recommended to use, is the Tivoli Monitoring middle layer directory (\$DBDIR/dmm1).

**subnet-mask:network\_id**

Subnet mask of the CommonListener component. For example:

```
subnet-mask:mynetworkid=255.255.255.0
```

**tbsma.jre\_root**

This parameter is set during the installation of the Tivoli Business Systems Manager Adapter (see “Installing Tivoli Business Systems Manager Adapter” on page 73), and you do not normally need to change it manually. However, if, for example, you want to install the adapter on a group of gateways using one instance of the install action or command, you will then need to change this parameter on any gateways in the group that have JRE installed at a location different to that supplied on the Install Options dialog.

Set this parameter to the complete path of the root directory of Java Runtime Environment, 1.3.0 (excluding the directory `/bin`).

**Note:** In a Windows NT workstation, if the install target path contains a directory with spaces in the name, the directory name must be specified between single quotes, as in this example:

```
D:\'Program Files'\jre
```

**trace.filename**

Filename to which will be written the trace messages from the adapter. The default is **dm.trc**.

**transport.mqe.usefiller**

Set this to **true** if the gateway on which the adapter is installed is running Windows NT, 4.0, Service Pack 5; otherwise leave as the default value of **false**.

**transport.server.ip.address**

IP address or hostname of the CommonListener component of Tivoli Business Systems Manager, which listens out for messages from the systems being managed. For example: `transport.server.ip.address=193.202.74.21`

**transport.server.mqe.port**

Port number of the CommonListener component. For example:  
`transport.server.mqe.port=8082`

**transport.trace.enable = false**

Set this to **true** if you want to write to the file identified in `trace.filename`, all messages regarding the transport of adapter-acquired data to the CommonListener. The default is **false**.

**transport.trace.level**

Set this to **low**, **medium** or **high**, according to the level of details you require if you have enabled transport trace messages. The default is **low**.

All these key/value pairs and their values are contained in file: `$DBDIR/dmm1/.config`

This file can be read but it must not be modified manually. If you wish to modify it, use the **wdmconfig** command (note that the applied changes might not show).

For full details of the command syntax see “`wdmconfig`” on page 111

## Using the Tivoli Business Systems Manager Adapter

Once installed and configured, the adapter provides its services in a fully automatic way, not needing to be independently started or stopped.

It carries out the following activities:

### Bulk discovery

The **wdm**discovery command can be issued from the Tivoli desktop to carry out a bulk discovery. The adapter sends details of all systems it has in its cache to the Tivoli Business Systems Manager’s CommonListener. For full details of the command syntax see Appendix A, “Commands” on page 107. Figure 8 shows the data flows for bulk discovery.

### Delta discovery

The same **wdm**discovery command can be issued from the Tivoli desktop to carry out a delta discovery. The adapter sends details of all changes since the last discovery, to the Tivoli Business Systems Manager’s CommonListener. For full details of the command syntax see Appendix A, “Commands” on page 107. Figure 8 shows the data flows for delta discovery.

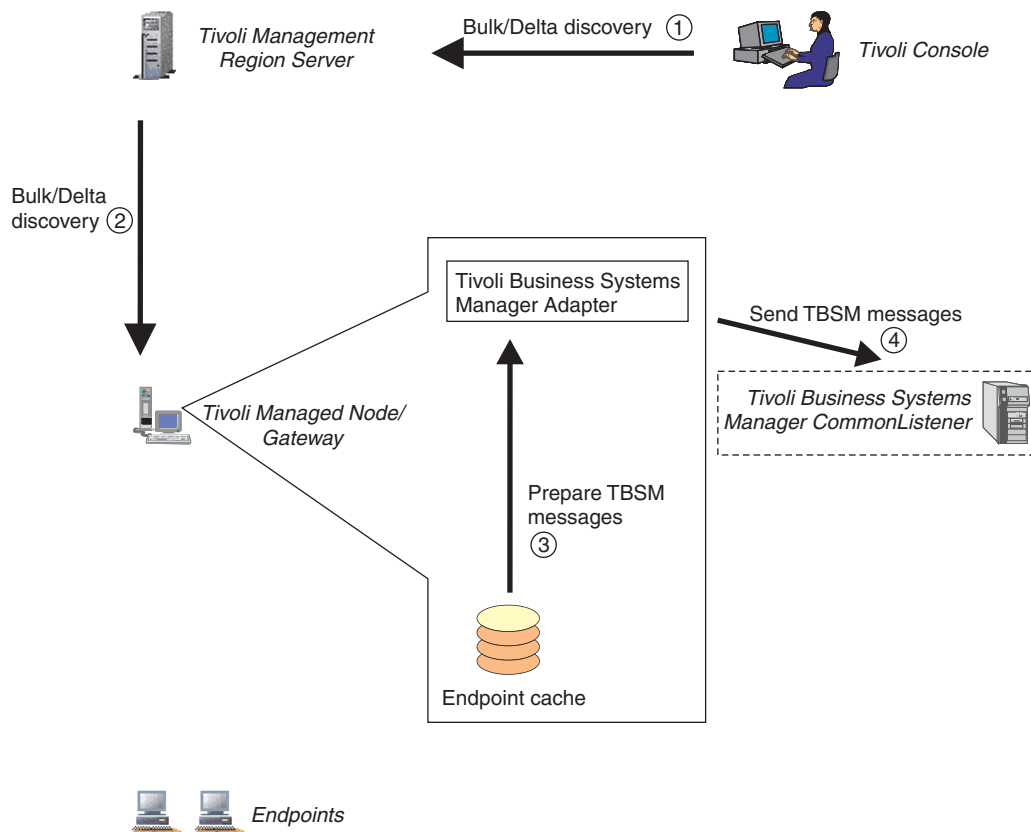


Figure 8. Data flows for bulk/delta discovery

The figure shows the data flow for bulk and delta discovery. The **wdm**discovery command is issued at the Tivoli Desktop (steps 1 and 2). At the target gateway, the Tivoli Business Systems Manager Adapter obtains the appropriate discovery information (bulk or delta) from the endpoint cache (step 3) and sends it to the Tivoli Business Systems Manager’s CommonListener (step 4),

**Note:** The Adapter expects that the CommonListener will respond to a discovery command within a configurable time-out (the default is 300 seconds). If the systems on which these components are running have misaligned time settings, the time-out value may seem to have been exceeded, with the discovery being rejected. The solution is either to align the systems' time settings or to increase the time-out value. Full details are given in "Problem Determination" on page 193.

## Viewing Tivoli Business Systems Manager Data

Tivoli Business Systems Manager data can be viewed from the Java Console, as shown in Figure 9. The activity takes place without any interaction with Tivoli Monitoring components.

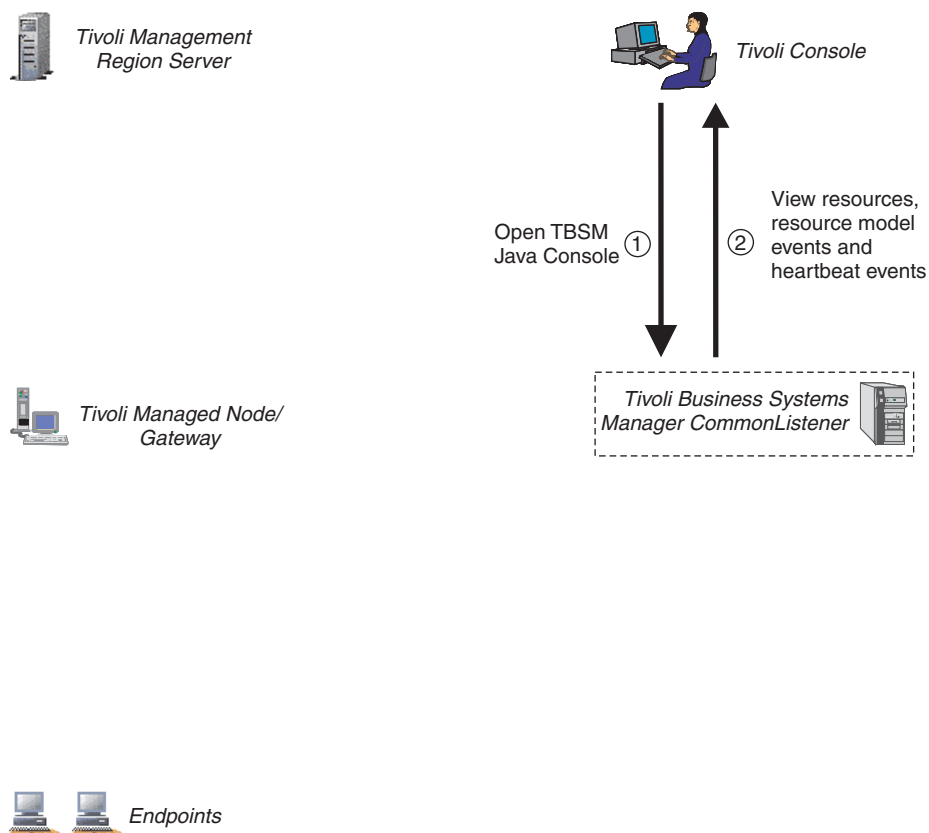
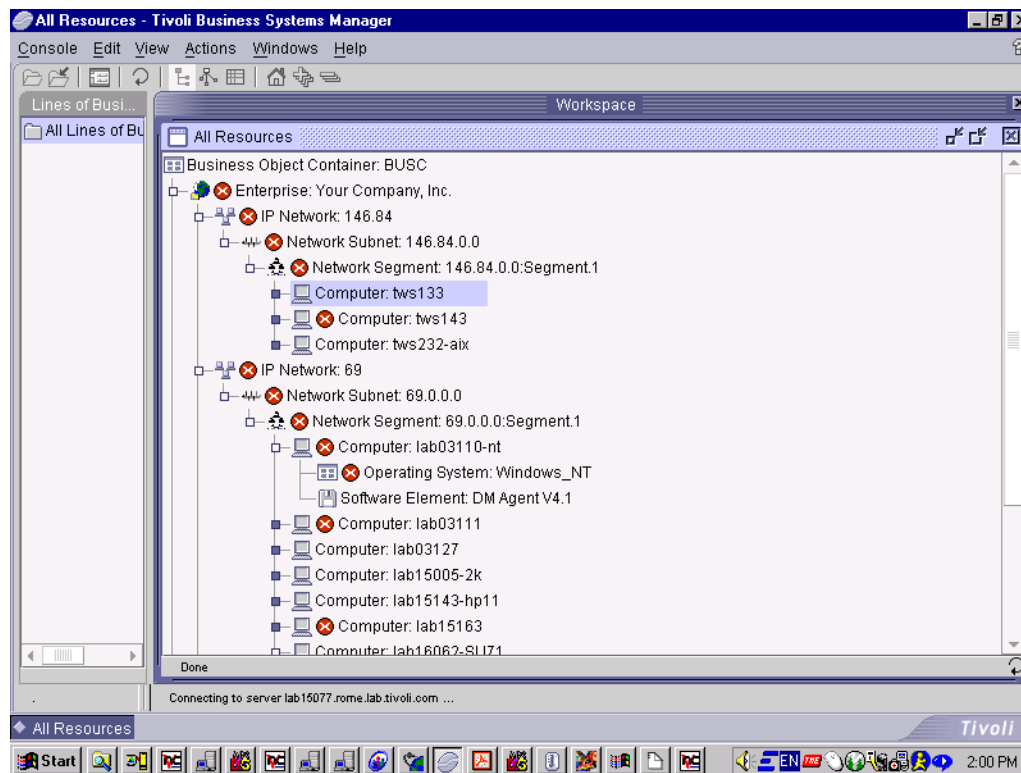


Figure 9. Data flow for viewing Tivoli Monitoring data in Tivoli Business Systems Manager

The data is viewed with Tivoli Business Systems Manager, as shown in the following Tivoli Business Systems Manager dialog.



This dialog shows the All Resources view of Tivoli Business Systems Manager. A Business Object Container has been opened to show an enterprise with at least two IP networks. IP network 69 has been opened to show a single subnet and segment within which are a number of computers. Computer lab03110-nt has been opened to show two entities:

### Operating System

The properties of this object contain information about the resources of the computer, including the Tivoli Monitoring events.

### Software Element

The properties of this object contain the heartbeat information.

**Note:** For every subnet in the enterprise hierarchy you can configure a subnet mask using the **wdmconfig** command (see “Configuring Tivoli Business Systems Manager Adapter” on page 74).

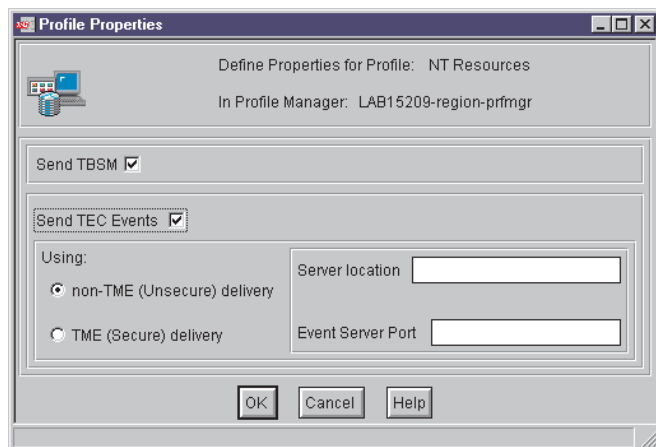
## Enabling and Disabling Tivoli Business Systems Manager Integration

For each profile you create, you can choose to disable the Tivoli Business Systems Manager integration for all resource models in the profile. This is an override that ignores the **Send to TBSM** option that may be set up for any indication in any resource model in the profile (see “Customizing Thresholds and Event Rules” on page 44).

To disable the Send to TBSM option, perform the following steps:

1. Open the Tivoli Monitoring Profile dialog.
2. Select **Edit** → **Properties**.

The Properties dialog opens and displays the name of the profile for which the server identity is to be set, and the profile manager in which the profile is located. It also displays the check box **Send to TBSM**.



3. If you want to send events in this profile to Tivoli Business Systems Manager, check the **Send to TBSM** option.

**Note:** If you do not select this check box, no events are sent to the Tivoli Business Systems Manager, even if you selected the **Send to TBSM** check box in the Indications and Actions dialog.

4. Click **OK** to save the properties.  
The Tivoli Monitoring Profile dialog reopens.

## Managing the Tivoli Business Systems Adapter from the CLI

The following commands are available for managing the Tivoli Business Systems Manager Adapter from the command line interface:

### **wdmconfig**

Updates the configuration file of the Tivoli Business Systems Manager Adapter.

### **wdmdiscovery**

Runs a bulk or delta discovery on selected endpoints at selected gateways. The command should only be used if the Tivoli Business Systems Manager Adapter component is installed.

### **wdmmngcache**

Deletes or lists part or all of the contents of the endpoint cache.

For full details of the commands, see Appendix A, “Commands” on page 107.

---



---

# Part II. Web Health Console

<b>Chapter 7. Introduction to the Web Health Console.....</b>	<b>83</b>
Overview .....	83
Understanding Resource Health .....	83
Connecting the Web Health Console .....	84
The Request Manager Processor .....	84
Software Requirements .....	85
Installing the Web Health Console .....	85
Installing on Windows .....	85
Installing on UNIX .....	86
Troubleshooting the Installation .....	87
Enabling Secure Socket Layer Support .....	90
<b>Chapter 8. Using the Web Health Console.....</b>	<b>93</b>
Logging onto the Web Health Console .....	93
Logging on to the Web Health Console for the first time .....	94
Setting Your Preferences .....	94
Managing Endpoint Lists .....	94
Managing General Preferences .....	96
Managing Chart Preferences .....	96
Common Web Health Console Features .....	97
Using the Endpoint List View .....	98
Working with the Endpoints .....	99
Using the Resource Model List View .....	99
Using the Endpoints by Resource Model View .....	100
Working with the Endpoints .....	101
Working with the Endpoint Health View .....	101
Working with the Resource Models Frame .....	102
Working with the Resource Models .....	102
Indications Frame .....	103
Historical Data Frame .....	103
Working with the Chart Views .....	104
Online Data Graph .....	104
Historical Data Graph .....	105

---

# 7

## Introduction to the Web Health Console

---

This chapter provides an overview of the IBM Tivoli Monitoring Web Health Console (Web Health Console) and the information to install it.

The Web Health Console runs on Netscape 6.2 (or later) and Internet Explorer 6.x. You can use the Web Health Console for the following purposes:

- Checking, displaying, and analyzing the status and health of endpoints that have distributed resource monitors
- Displaying an endpoint's real-time and historical data logged to the IBM Tivoli Monitoring database
- Viewing online and historical data on endpoints as a follow-up to specific problems
- Starting and stopping the IBM Tivoli Monitoring engine and individual resource models on selected endpoints
- Removing a profile from the selected endpoint

This chapter contains the following information:

- A conceptual overview of the Web Health Console
- An explanation of how health is determined for a resource
- Connection information for the Web Health Console

### Overview

You can use the Web Health Console to check, display, and analyze the status and health of any endpoint with profiles and resource models. Status reflects the state of the endpoint displayed on the Web Health Console, such as running or stopped. Health is a numeric value determined by resource model settings. You can also use the Web Health Console to work with real-time or historical data from an endpoint that is logged to the IBM Tivoli Monitoring database.

You can use the diagnostic and monitoring capabilities of the Web Health Console to perform targeted analysis of problems associated with individual endpoints when an event is sent to the Tivoli Enterprise Console. Use the online and historical data to follow up specific problems with single endpoints.

### Understanding Resource Health

The Web Health Console obtains events and indications from endpoints. The Web Health Console displays the health of each potential problem as a numeric value between 100 (perfect health) and zero (with zero meaning that the conditions for the corresponding event

are met). Intermediate values show the percentage of occurrences currently registered with respect to the total number of occurrences needed to trigger an event. See Table 11

*Table 11. Health Determination Example*

Cycle	1	2	3	4	5
CPU%	55	73	54	63	68
Occurrences or Holes	H	O	H	O	O
Occurrence Count	0	1	1	2	3
Health %	100	75	75	50	25

Table 11 gives the health percentage changes in steps of 25% because four occurrences were required to trigger an event. If the indication required five occurrences, the health percentage would have changed by steps of 20%. Resource health is determined at the indication level and passed up to the endpoint. The lowest health of any indication in a resource model is shown as the health of that resource model, and the lowest health of any resource model installed on an endpoint is shown as the health of that endpoint. For example, if one indication on one resource model that is installed on an endpoint has a health of zero, the health of the endpoint is shown as zero. The required occurrences, cycle times, thresholds, and parameters for indications are defined when the resource model is created in the IBM Tivoli Monitoring Workbench. For information about this, see Part I, “Installing, Configuring, and Using” on page 1.

## Connecting the Web Health Console

You can connect the Web Health Console to any Tivoli management region server or managed node and configure it to monitor any or all of the endpoints that are found in that region (assuming that all gateways are interconnected).

To connect to the Web Health Console you need access to the server on which the Web Health Console server is installed and the IBM Tivoli Managed Region on which you want to monitor health. All user management and security is handled through the IBM Tivoli management environment. This includes creating users and passwords as well as assigning authority. For more information about how to log to the Web Haelst Console, refer to “Logging onto the Web Health Console” on page 93.

## The Request Manager Processor

The Web Health Console uses the request manager processor of Tivoli Monitoring. The request manager collects, stores, and manages all endpoint requests created and used by Tivoli Monitoring components (such as the Web Health Console and the heartbeat) and applications.

The request manager acts like a down-call concentrator: it receives all requests (from the various Web Health Console monitors and from the heartbeat processor) and concentrates them into one request to the endpoint. Once the endpoint sends data back, the request manager stores the data in a centralized cache on the gateway. At that point, the Web Health Console monitors and the heartbeat can retrieve the data from the gateway cache.

The following figure shows how the request manager works:

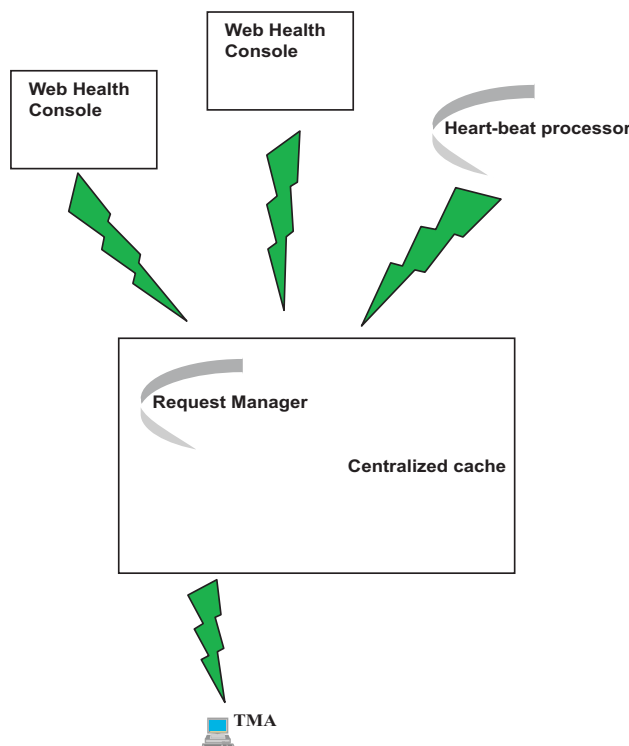


Figure 10. The request manager processor

To customize and use the request manager processor, see the corresponding parameters described in the **wdmconfig** command, Appendix A, “Commands” on page 107.

## Software Requirements

The Web Health Console runs on Netscape 6.2 (or later) and Internet Explorer 6.x.

The Web Health Console uses WebSphere Application Server, Advanced Edition Single Server 4.0.2, which is installed as part of the Web Health Console install. Refer to the following web address for the software prerequisites of WebSphere Application Server, Advanced Edition Single Server 4.0.2:

[http://www-3.ibm.com/software/webservers/appserv/doc/v40/prereqs/aes\\_v402.htm](http://www-3.ibm.com/software/webservers/appserv/doc/v40/prereqs/aes_v402.htm)

## Installing the Web Health Console

The following sections describe how to install the Web Health Console on the different operating systems.

### Installing on Windows

To install the Web Health Console on Windows:

1. Install Disk 2 on the CD-ROM drive and double-click the `setupwin32.exe` file.

**Note:** If the temporary directory does not have enough space, you could get an error. To avoid this, enter the following command:

```
setupwin32 -is:tempdir TMPdir
```

where *TMPdir* is the name of the temporary directory.

2. Follow the directions presented in the install dialogs. In particular:

- provide the Directory Name for the location on which you wish to install the Web Health Console Server. The Directory Name must contain no spaces.
- provide the User name under which the Web Server will run. This user must have Act As Operating System access.
- provide the Password for the User.

To install in "silent" mode via the command line, you must provide the following arguments:

**-silent**

**-P base\_install.installLocation="Directory Name"**

**-W user\_input.user="User Name"**

**-W user\_input.password="Password"**

You can also include the above arguments into a file and pass the file to the launcher using the options switch:

**-options options file**

## Installing on UNIX

To install the Web Health Console on UNIX:

1. Install Disk 1 on the CD-ROM drive and double-click one of the following files depending on the UNIX platform you are running:

- **setupaix.bin** on AIX
- **setuphp1020.bin** on HP-UX10.2
- **setuphp11x.bin** on HP-UX11.x
- **setupsolarisSparc.bin** on Sun Solaris
- **setuplinux.bin** on Linux

**Note:** If the temporary directory does not have enough space, you could get an error. To avoid this, enter the command (corresponding to your UNIX platform):

```
setupaix -is:tempdir TMPdir
```

where *TMPdir* is the name of the temporary directory.

2. Follow the directions presented in the install dialogs. In particular, provide the Directory Name for the location on which you wish to install the Web Health Console Server.

To install in "silent" mode via the command line, you must provide the following arguments to the setup command indicated before:

**-silent**

**-P base\_install.installLocation="Directory Name"**

You can also include the above arguments into a file and pass the file to the launcher using the options switch:

**-options options file**

**Note:** The UNIX installation will install the IBM HTTP Server into a standard location regardless of the directory that you specify for the install. The rest of the installation will go to the directory that you have specified. The standard locations are:

- AIX: /usr/HTTPServer
- Sun Solaris: /opt/IBMHTTPD
- Linux: /opt/IBMHTTPServer
- HP: /opt/HTTPServer
- UNIX: /opt/HTTPServer

## Troubleshooting the Installation

The Web Health Console installation installs three software components:

- WebSphere Application Server, Advanced Edition, Single Server, 4.0.2
- IBM HTTP Server
- Web Health Console

In order to install WebSphere Application Server, the Web Health Console installation invokes the WebSphere Application Server silent install. This means that, to debug the install, you may need to review several log files.

The primary log for the install is `AMW_WAS_Install.log`, which is stored in the temporary directory used by InstallShield. In addition, several other logs have useful information. The WebSphere Application Server install, which is executed by the Web Health Console, creates log files at the following location:

- **on UNIX:** /tmp/install.log
- **on Windows:**
  - during the install:
    - `install_dir/setup.log`: this is the IBM HTTP Server install log
    - `install_dir/StagingDir/setup.log`: this is the WebSphere Application Server install log
  - after the WebSphere Application Server completes:
    - `install_dir/logs/wssetup.log`

If the installation is not successful, see the following troubleshooting information:

- Make sure that WebSphere Application Server is running:
  - Access `http://hostname:9090/admin`
  - If this loads the WebSphere Application Server Admin console, then WebSphere Application Server is running correctly
  - If this does not load the WebSphere Application Server Admin console:
    - Check the WebSphere Application Server install log previously indicated
    - Make sure that the WebSphere Application Server application is running:
      - **on Windows:** check the Services Panel. Make sure that the service "WebSphere Application Server V4 - WebSphere for ITM" is running. If not, attempt to start it using the controls in the Services Panel.

- |                   ■ **on UNIX:** check for the WebSphere Application Server process using the **ps**  
|                   command on the java process under the WebSphere Application Server install  
|                   directory. If WebSphere Application Server is not running, try to start it using  
|                   *install\_dir/bin/startServer.sh*
- |           – Check the WebSphere Application Server run time logs at: *install\_dir/logs*
- |   ■ Make sure that the IBM HTTP Server is running:
  - |   • Check the installation logs:
    - |   – **on Windows:** look in *install\_dir/setup.log* and in  
|   *install\_dir/StagingDir/setup.log*
    - |   – **on UNIX:** look in */tmp/install.log*. The IBM HTTP Server install occurs  
|   approximately half way through the *install.log*. The command line used to  
|   install IBM HTTP Server is in this log. To find it, look for "IHS install  
|   command". Copy that command and attempt to run it manually on the command  
|   line.
  - |   • Check if the process is running:
    - |   – **on Windows:** check the Services Panel. Make sure that the service "IBM HTTP  
|   Server" is running. If not, attempt to start it using the controls in the Services  
|   Panel. If the service does not start, try to start the Server from the command line  
|   by running: *install\_dir/HTTP\_Server/Apache*.
    - |   ■ If the Server runs on the command line but not in a service, check the  
|   parameters of the service:
      - |   • Ensure the Login and Password are correct.
      - |   • Ensure the path is correct.
      - |   • If you are using Windows NT 4.0, attempt the following:
        - |   1. Run **regedit**.
        - |   2. Go to:  
|    HKEY\_LOCAL\_MACHINE/System/CurrentControlSet/Services/  
|    IBMHTTPService/Parameters
        - |   3. Set the value of *ConfigArgs* to be blank.
    - |   **Note:** Editing the Windows registry can corrupt the Windows  
|   operating system. Attempt the above instructions only if you are  
|   knowledgeable about the Windows Registry.
  - |   ■ If you see errors referring to the Security Library, check the following:
    - |   • Ensure that the IBM Key Management Utility is installed. If it is installed,  
|   it shows in Program Files/ibm/gsk5 on the drive where you installed  
|   the Web Health Console.
    - |   • Make sure that the environment variable *PATH* contains the path to the  
|   IBM Key Management Utility. For example: *PATH=c:\Program*  
|   *Files\ibm\gsk5\lib*.
  - |   – on UNIX: Check for the IBM HTTP Server process using the **ps** command on the  
|   http process. If IBM HTTP Server is not running, try to start it using *IBM HTTP*  
|   *Server install\_dir/bin/apachectl start*. If the IBM HTTP Server does not  
|   start, look at the log files in *IBM HTTP Server install\_dir/logs*.



- Check that the Web Health Console application is installed correctly:
  - If both WebSphere Application Server and IBM HTTP Server are running correctly but the Web Health Console is still not available, then the Web Health Console is not functioning. At this point do the following:
    1. Launch the WebSphere Application Server Admin Console:  
`http://hostname:9090/ADMIN`
    2. Go to Nodes → *yourNodeName* → Enterprise Applications.  
The application "DM Web Health Console" should be present and it should be running.
      - If the application is present but it is not running, try to start it. If it does not start, check the following:
        - Look in the WebSphere logs at *install\_dir*/logs.
        - Look in the Web Health Console logs at:  
`c:\Tivoli\AMW\logs\traceDMWebConsole.log` or  
`/opt/Tivoli/AMW/logs/traceDMWebConsole.log`
      - If the application is not present, check the following:
        - Check (for WebSphere Application Server) *install\_dir*/healthConsole. It should contain the file `dm.ear`. If the file is not there, then the installation has failed completely. Check the `AMW_WAS_Install.log` indicated before.
        - Check *install\_dir*/installedApps. It should contain a directory named `dm.ear`. If it does not contain that directory, then the configuration command failed to run:
          - Check the `AMW_WAS_Install.log` and *install\_dir*/appInstall.out or *install\_dir*/appInstall.err for details.
          - Attempt to run the following command manually:  
`install_dir/bin/SEAppInstall -install`  
`install_dir/healthConsole/dm.ear -interactive false`
        - Check *install\_dir*/config/server-cfg.xml. It should contain a reference to the Web Health Console. If it does not contain a reference, but the previous step worked, you probably overwrote the file when you logged into the Admin Console. To correct this, stop the WebSphere Application Server by running: `install_dir/bin/stopServer[.sh]` and execute again the second step in this list.
        - Check the *install\_dir*/config/plugin-cfg.xml. It should contain a reference to the string "dmwhc". If it does not contain a reference, then GenPluginCfg command failed. Look in *install\_dir*/appInstall2.out or *install\_dir*/appInstall2.err for details. Run the following command manually:  
`install_dir/bin/GenPlugIn -configFile install_dir/config/server-cfg.xml`
        - Restart the server, if you stopped it, by running:  
`install_dir/bin/startServer[.sh]`

## Enabling Secure Socket Layer Support

The Web Health Console installation inserts configuration parameters for Secure Socket Layer (SSL) support, but does not enable SSL. To enable SSL, you must first provide a key database containing the site specific keys and certificates, then uncomment the inserted configuration information.

This section shows you the steps to enable SSL.

**Note:** In the following directions, replace *IBM HTTP Server Location* with:

- AIX: /usr/HTTPServer
- Sun Solaris: /opt/IBMHTTPD
- Linux: /opt/IBMHTTPServer
- HP: /opt/HTTPServer
- UNIX: /opt/HTTPServer
- Windows: Web Health Console Server Location/HTTP\_Server

Follow these steps to enable SSL:

1. Create a directory on the server to hold your SSL key database files and certificates. In the following steps, this directory is referred to as `key_db_dir`.
2. **For Windows NT:** start the IBM Key Management utility by invoking **gsk5ikm.exe** from `[WEB_HEALTH_CONSOLE_INSTALL_DRIVE]\Program Files\ibm\gsk5\bin` or by selecting **Start>Programs>IBM HTTP Server>Start Key Management Utility**.  
**For AIX, Sun Solaris, HP/UX, and Linux:** start the IBM Key Management utility by invoking **gsk5ikm** from:
  - AIX: /usr/ibm/gsk5/bin
  - Linux: /usr/local/ibm/gsk5/bin
  - Sun Solaris: /opt/ibm/gsk5/bin
  - HP/UX: /opt/ibm/gsk5/bin
3. Within IBM Key Management, select **Key Database File>New**.
4. Complete the fields in the **New** dialog box, making the following selection entries:
  - **Key Database Type:** Select the **CMS Key database file**
  - **File Name:** Enter a base filename for the new key database files. Several files will be created using this base filename, including:  
`[key_database_name].kdb`, `[key_database_name].rdb`,  
`[key_database_name].sth`, and `[key_database_name].crl`.  
To use the default Web Health Console configuration, use *IBM HTTP SERVER LOCATION/key\_db* for the Location.
5. Click **OK**.
6. Complete the fields in the **Password Prompt** dialog box, making the following selection and entries:
  - Enter the password which will be used to encrypt and decrypt the key database
  - Reenter the password to confirm that it is typed correctly

- Select **Stash the password to a file**

7. If you have a server certificate from a Certificate Authority (for example, Verisign), you can click **Import** to import this certificate into your SSL key file. If not, create a new one by selecting **Create>New Self-Signed Certificate**.
8. Complete the fields in the **Create New Self-Signed Certificate** dialog box, making the following selection and entries:
  - **Key Label:** A label by which the key and certificate in the database will be identified.
  - **Common Name:** The fully qualified hostname of the server on which the Web Health Console is installed.
  - **Organization:** Enter an organization name. This information will be presented to the client as part of the credentials of the server.

Leave the remaining parameters at their default value, and click **OK**.

9. Generally speaking, SSL communication is run on a different port than the standard unencrypted communication typically used on port 80. By convention, the port used for SSL is port 443.

Edit the file *IBM HTTP Server Location/conf/httpd.conf*. At the end of this file, you will find the lines described hereafter. Uncomment these lines. If you did not use the default location and name for the key database, insert the appropriate information.

**For Windows NT:**

```
#LoadModule ibm_ssl_module
modules/IBMModuleSSL128.dll
#Listen 443
#SSLEnable
#Keyfile "IBM HTTP SERVER LOCATION\key_db\key.db"
```

**For AIX, Sun Solaris, HP/UX, and Linux:**

```
#LoadModule ibm_ssl_module
libexec/mod_ibm_ssl_128.so
#Listen 443
#SSLEnable
#Keyfile "IBM HTTP SERVER LOCATION\key_db\key.db"
```

10. **On Windows:**

Restart the IBM HTTP Server by selecting **IBM HTTP Server** in the Windows NT Services control panel, clicking **Stop**, and then clicking **Start**

**On UNIX:**

Restart the IBM HTTP Server by running *IBM HTTP Server Location/bin/apachectl restart*.

The IBM HTTP Web Server is now configured for SSL.



# 8

## Using the Web Health Console

---

This chapter describes how to use the Web Health Console, navigate the components, and interpret component contents.

You can use the Web Health Console to display real-time or historical data about resources on a specified system. The Web Health Console displays 24 hours of historical data.

The data in the Web Health Console is automatically refreshed every 5 minutes, by default. You can change the refresh rate of the Web Health Console on the **General** tab of the **Preferences** view. See “Managing General Preferences” on page 96 for more information.

### Logging onto the Web Health Console

Perform the following steps to log onto the Web Health Console:

1. Connect to `http://server_name/dmwhc`, where *server\_name* is the server on which you installed the Web Health Console server.



2. Supply the following information to display the **Login View**:

**User** Tivoli user ID

**Password**

The Password associated with the Tivoli user ID

**Host** The Name of the managed node to which you want to connect

3. Click **OK** to connect to the Web Health Console.

A message is displayed above the text entry fields if the login was unsuccessful, or you were redirected to this view because of a different error.

## Logging on to the Web Health Console for the first time

Use the following steps if you are logging on to the Web Health Console for the first time:

1. Log on to the Web Health Console.

The first time you log on to the Web Health Console, the **Preferences** view is displayed. You must populate the **Selected Endpoint** list before you can access any other Web Health Console views. When you log on subsequently, the endpoint list is automatically loaded.

2. Select the endpoints that you want to see. See “Managing Endpoint Lists” for more information.

3. Customize the Web Health Console to your specification. See “Managing General Preferences” on page 96 and “Managing Chart Preferences” on page 96 for more information.

## Setting Your Preferences

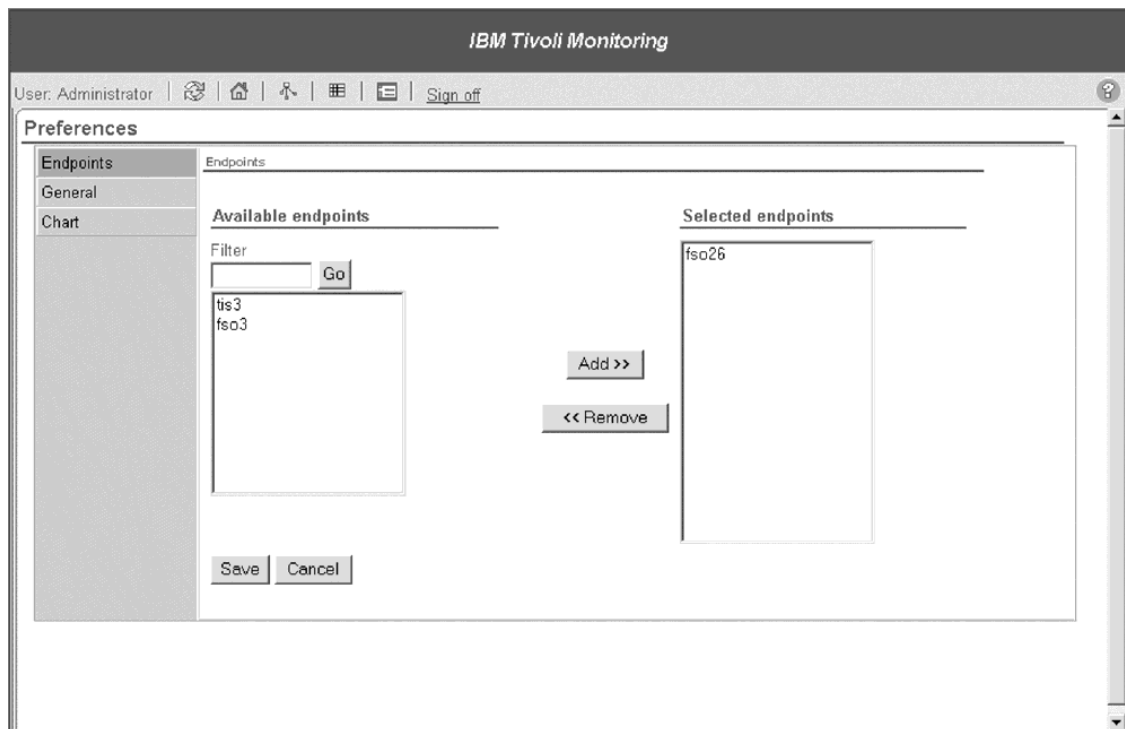
You can customize the Web Health Console views to display your preferred settings. This section explains how to manage the following preferences:

- “Managing Endpoint Lists” on page 94
- “Managing General Preferences” on page 96
- “Managing Chart Preferences” on page 96

## Managing Endpoint Lists

To maximize performance you should limit the number of endpoints that the Web Health Console monitors. Use **Preferences** view to specify only those endpoints in which you are

interested.



The first time you log into the Web Health Console the **Preferences** view is displayed. You must populate the **Selected Endpoint** list before you can access any other Web Health Console views. When you log on subsequently, the endpoint list is automatically loaded.

To populate the endpoint list:

1. Type the name of an endpoint filter in the **Filter** field.  
For example, type `abc*` to display all endpoints in the Tivoli managed region with names that start with `abc` and that have resource models installed. The filtering capability support is limited to the `(*)` regular expression character.
2. Click **Go**.
3. Click on the endpoint to select it from the **Available endpoints** list.  
Use Control-click or Shift-click to select multiple endpoints.
4. Click **Add >>**.  
Repeat these steps to add more endpoints to the list.

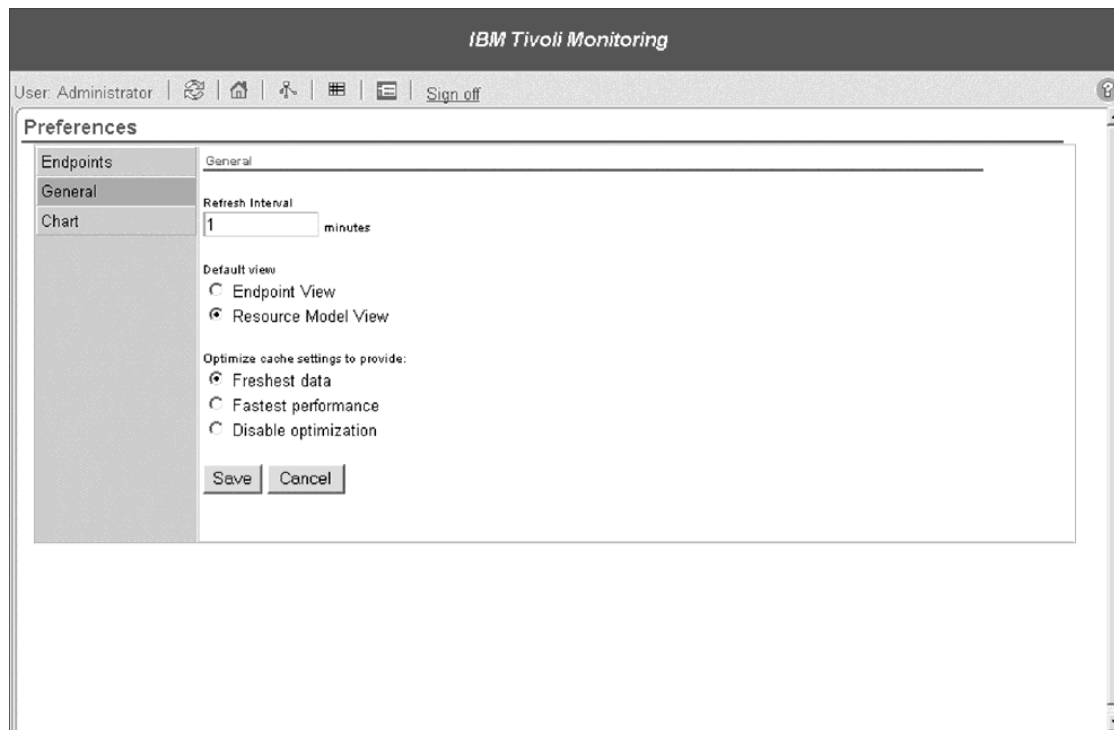
**Note:** You can apply a new filter at any time to update the **Available endpoints** list without affecting the **Selected endpoints**. You can move endpoints from the **Selected endpoints** list to the **Available endpoints** list by using the **<< Remove** button.

5. Click **Save** to save the **Selected endpoint** list or **Cancel** to cancel these changes.

After you populate your views, you can configure other preferences. See “Managing General Preferences” on page 96 and “Managing Chart Preferences” on page 96.

## Managing General Preferences

Perform the following steps to manage the **General Preferences** view:



1. Log on to display the **Preferences** view.
2. Click the **General** tab.
3. Select and define the following:

### **Refresh Interval**

The number of minutes between automatic updates of the data views. The default interval is 5 minutes. The interval you specify here does not control the rate at which the chart views refresh. Refer to “Managing Chart Preferences” on page 96 for more information.

### **Default view**

The view you want displayed whenever you log in to the Web Health Console. The default is the **Endpoint View**.

### **Optimize cache settings to provide:**

**Note:** This setting should be changed only by advanced users.

The optimization setting that provides the best service to you or no optimization. The default setting is **Freshest Data**.

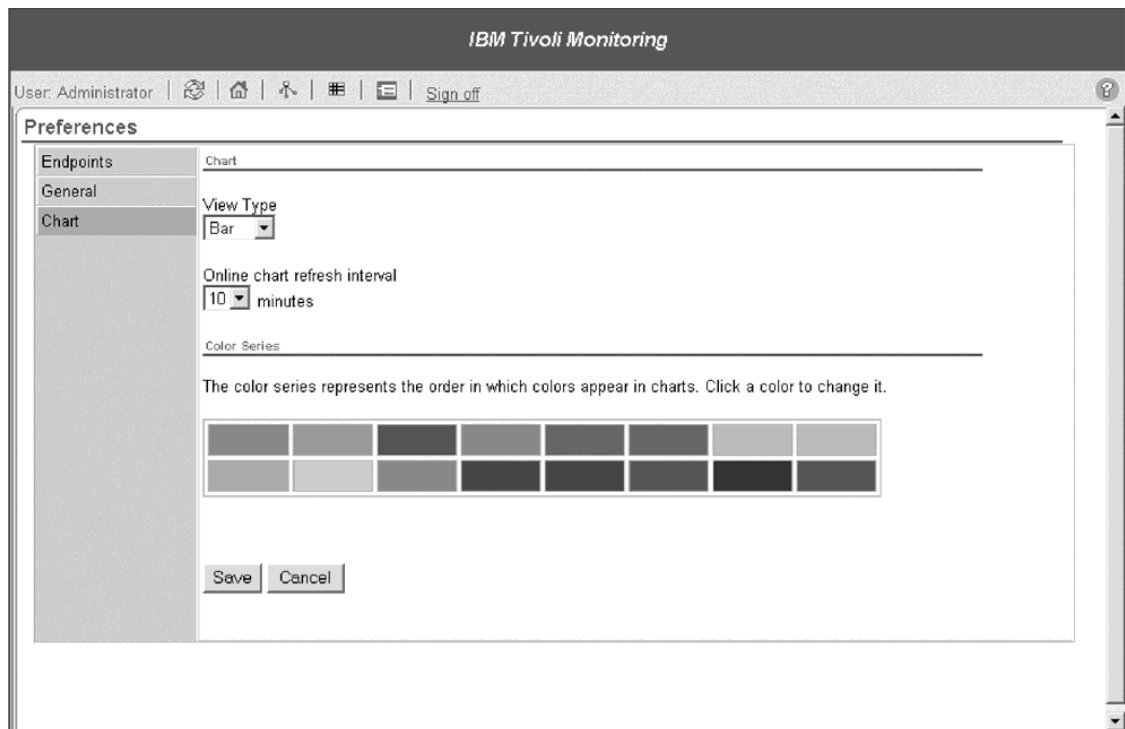
4. Click **Save** to implement changes you make to the **Preferences, General** view.

## Managing Chart Preferences

Use the **Chart** page of the **Preferences** view to select the type of chart that is displayed when you click **Graph** on the **Endpoint** view, the refresh interval for that chart, and the



colors in which that chart displays.



Perform the following steps to manage the **Preferences, Chart** view:

1. Log on to display the **Preferences** view.
2. Click the **Chart** tab.
3. Select a chart from the **View Type** drop-down list.
4. Select the refresh interval from the **Online chart refresh interval** drop-down list.
5. The color series represents the order in which colors appear in all of the charts. To change a color:
  - a. Click the color you want to change to display a color palette containing all of the possible choices.
  - b. Specify the color with which you want to replace the selected color.
  - c. Click **Save**.
6. Click **Save** to implement the changes.








## Common Web Health Console Features

The top portion of every view of the Web Health Console, except the **Login** view, contains a common menu banner.



This menu banner contains the following information:

*Table 12. Navigation Icons*




	The user name with which you logged onto this session of the Web Health Console
	The refresh icon button to manually refresh the current view
	The home icon button to return to the view that is set on the <b>General</b> tab of the <b>Preferences</b> view. The options are the <b>Endpoint List View</b> or the <b>Resource Model List View</b>
	The endpoint list icon button to open the <b>Endpoint List View</b>
	The resource model icon button to open the <b>Resource Model List View</b>
	The view/edit preferences icon button to open the <b>Preferences</b> view
	The sign off button to return to the <b>Login</b> view

The common navigation bar has fly-over help associated with each button.

As you drill down through the views, the Web Health Console displays a dynamic path in the title portion of the window. This path shows the views you went through to arrive at the currently displayed view. It also provides links to any of previous views. You can return to a previous view by clicking on the title of that view.

Health is displayed in the views as an exact health percentage and as an iconic representation of possible states of alert. See Table 13 for a description of the health alert icons:

*Table 13. Health alert icons*

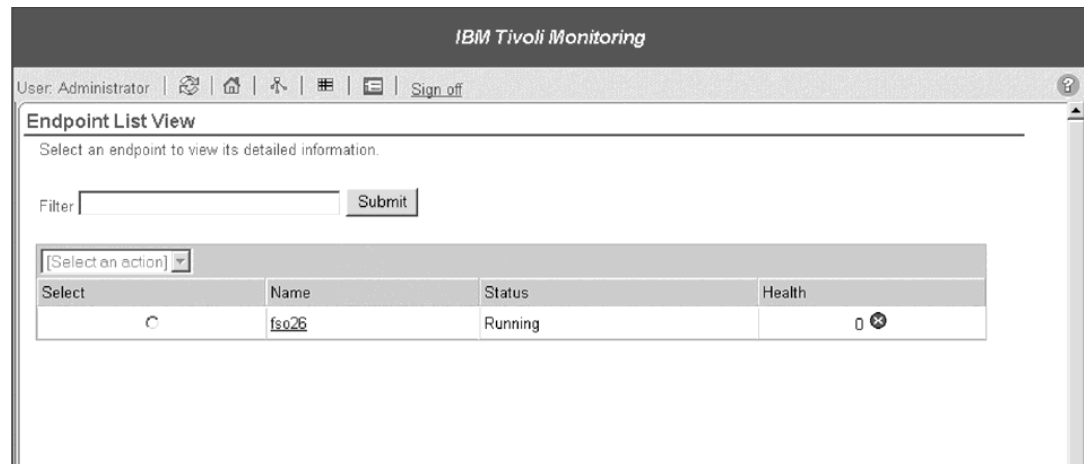
	The health of all of the resource models installed on the endpoint is at 100%.
	The health of at least one of the resource models installed on the endpoint is less than 100% but greater than 0.
	The health of at least one of the resource models installed on the endpoint is at 0.

## Using the Endpoint List View

The **Endpoint List View** shows the current health of all the endpoints specified in the **Endpoints** tab of the **Preferences** view. (See “Managing Endpoint Lists” on page 94 for more information.) The Web Health Console sorts the endpoints by health order, with the lowest health displayed first. For example, an endpoint with a health percentage of 20 is listed before an endpoint with a health percentage of 80. If there is a problem contacting an endpoint, the Web Health Console displays a message indicating the problem.

1. To display the health of a specific endpoint, or group of endpoints, type the appropriate information in the **Filter** field and click **Submit**.

For example, type `abc*` to display all endpoints in the Selected endpoints list with names that start with "abc" and that have resource models installed. The filtering capability support is limited to the (\*) regular expression character.



The **Endpoint List View** table provides information on the following:

**Status** The status of the specific endpoint being monitored, either Running or Could not be contacted.

**Health**

The lowest health of all the resource models installed on the endpoint. For example, if the endpoint has two resource models installed and one is at 20% health and the other is at 80% health, this column displays 20. The health is displayed as an exact health percentage and as an iconic representation of possible states of alert.

2. Click the endpoint name to display the **Endpoint** view for detailed information about each resource model running on that endpoint. Refer to “Working with the Endpoint Health View” on page 101 for detailed information.

—OR—

Click the radio button to select an endpoint and take action on that endpoint. See “Working with the Endpoints”.

## Working with the Endpoints

**Note:** These operations require the super, admin, or senior role.

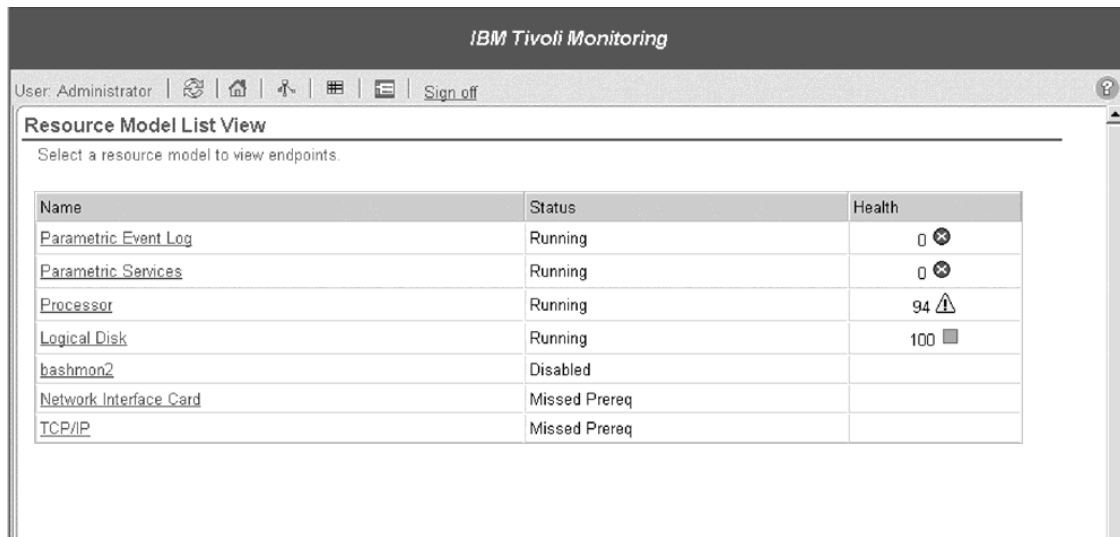
To start or stop the IBM Tivoli Monitoring engine from the **Endpoint List View**:

1. Click the radio button to select the endpoint.
2. Select **Start Engine** or **Stop Engine** from the **Select an action** drop-down list.

## Using the Resource Model List View

The **Resource Model List View** shows all of the resource models installed on the endpoints specified in the **Endpoint List** page of the **Preferences** view. (See “Managing Endpoint Lists” on page 94 for more information.) The Web Health Console sorts the resource models by health order, with the lowest health displayed first. For example, a resource model with a

health percentage of 20 is listed before a resource model with a health percentage of 80.



The **Resource Model List View** table provides information on:

**Status** The status of the endpoint associated with the resource model that is being displayed, either Running or Could not be contacted.

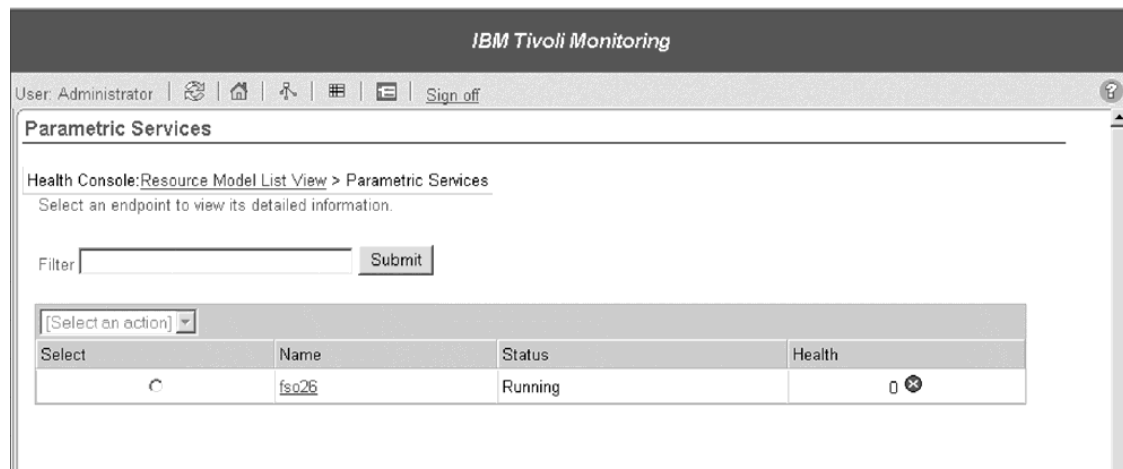
**Health**

The lowest health of all the endpoints on which this resource model is installed. For example, if the resource model is installed on two endpoints, one with a health percentage of 20 and the other with a health percentage of 80, this column displays 20. Health is displayed as an exact percentage and as an iconic representation of possible states of alert. See Table 13 on page 98 for information on the icons.

Click a resource model name to display the **Endpoint by Resource Model List View**. See “Using the Endpoints by Resource Model View” for more information.

## Using the Endpoints by Resource Model View

This view can be displayed only by clicking a resource model name in the **Resource Model List View**. Use the navigation bar at the top of this view to return to the **Resource Model List View**.



This view shows all of the endpoints specified in the **Endpoint List** page of the **Preferences** view that have the selected resource model installed. See “Managing Endpoint Lists” on page 94 for more information.

To display the health of the resource model on a specific endpoint, or group of endpoints, type the appropriate information in the **Filter** field and click **Submit**.

For example, type `abc*` to display all endpoints in the Selected endpoints list with names that start with “abc” and that have resource models installed. The filtering capability support is limited to the (\*) regular expression character.

## Working with the Endpoints

**Note:** These operations require the super, admin, or senior role.

You can start or stop the IBM Tivoli Monitoring engine from the **Endpoint List View**:

1. Click the radio button to select the endpoint.
2. Select **Start Engine** or **Stop Engine** from the **Select an action** drop-down list.

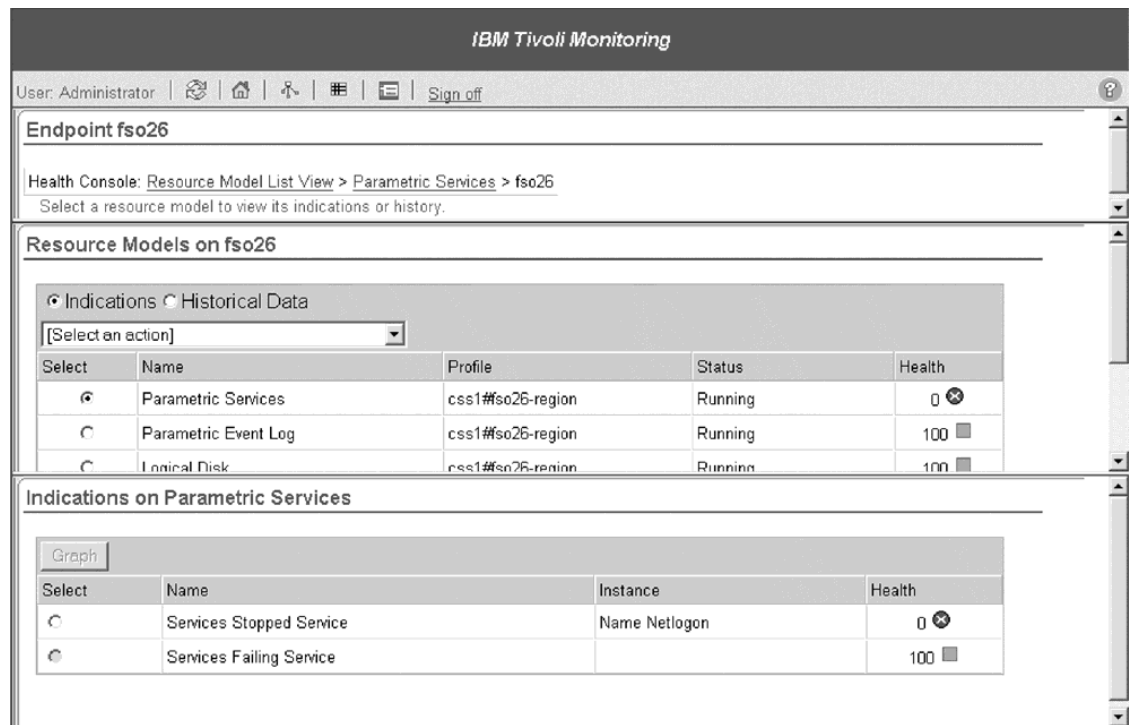
## Working with the Endpoint Health View

The **Endpoint Health** view is the most detailed view of the health of an endpoint. In this view, the following information is displayed:

- The health and status of all resource models installed on the endpoint
- The health of the indications that make up the resource model and historical data

The view is divided into three frames. The upper frame contains the common Web Health Console information. (See “Common Web Health Console Features” on page 97). The middle frame (**Resource Models** frame) displays a table of the resource models installed on this endpoint listed according to lowest health. The lower frame displays a table of the indications for the selected resource model (**Indications** frame) or the historical data controls

for the selected resource model (**Historical Data** frame).



## Working with the Resource Models Frame

The upper frame of the **Endpoint Health** view displays a table that includes the following information:

**Name** Resource model name

**Profile**

Name of the profile in which the resource model resides

**Status** Status of the resource model

**Health**

Health percentage of the resource model. The resource model health is the lowest health of any indication in the resource model. The Web Health Console sorts the endpoints by health order, with the lowest health displayed first.

Select **Indications** or **Historical Data** to determine the information displayed in the lower frame for the selected resource model. See “Indications Frame” on page 103 and “Historical Data Frame” on page 103 for more information.

## Working with the Resource Models

**Note:** These operations require the super, admin, or senior role.

From the **Endpoint Health** view you can perform the following actions on resource models:

**Start Resource Model**

To start the resource model selected.

**Stop Resource Model**

To stop the resource model selected.

**Start all Resource Models of the Same Profile**

To start all the resource models belonging to the profile selected.

**Stop all Resource Models of the Same Profile**

To stop all the resource models belonging to the profile selected.

**Remove Profile**

To delete the profile selected from the endpoint.

**Indications Frame**

This frame of the **Endpoint Health** view displays a table listing the indications for the selected resource model sorted by health.

Select	Name	Instance	Health
<input type="radio"/>	Services Stopped Service	Name Netlogon	0
<input type="radio"/>	Services Failing Service		100

From this table you can launch an online data graph for any indication with a health less than 100%. Select the desired indication and click **Graph**. This launches an online data graph in a separate window. See “Online Data Graph” on page 104 for more information.

**Historical Data Frame**

Using the historical data selection controls, you select instances and metrics from the selected resource model to create a chart of recent historical data from the endpoint. To create historical data, logging must be enabled for the resource model. With logged data, you can use the Historical Data Graph to identify specific instances of resource problems over the past one, six, twelve, or twenty-four hours.

To create a Historical Data Graph from the **Endpoint Health** view:

1. Select a resource model from the **Resource Model** list in the upper frame.
2. Click the **Historical Data** radio button to display the Historical Data selection information in the lower frame.
3. Click the **Resource** drop-down list and select a resource. The **Resource** drop-down list is the only option that is active when the frame opens.
4. Click the **Contexts** drop-down list and select a context. Each context identifies a logical grouping of problems related to the specified resource.
5. Select one or more instance from the **Instances** list. These identify specific instances of the selected indication.

6. Select one or more metric from the **Metrics** list. These are the metrics used to measure the selected indication.

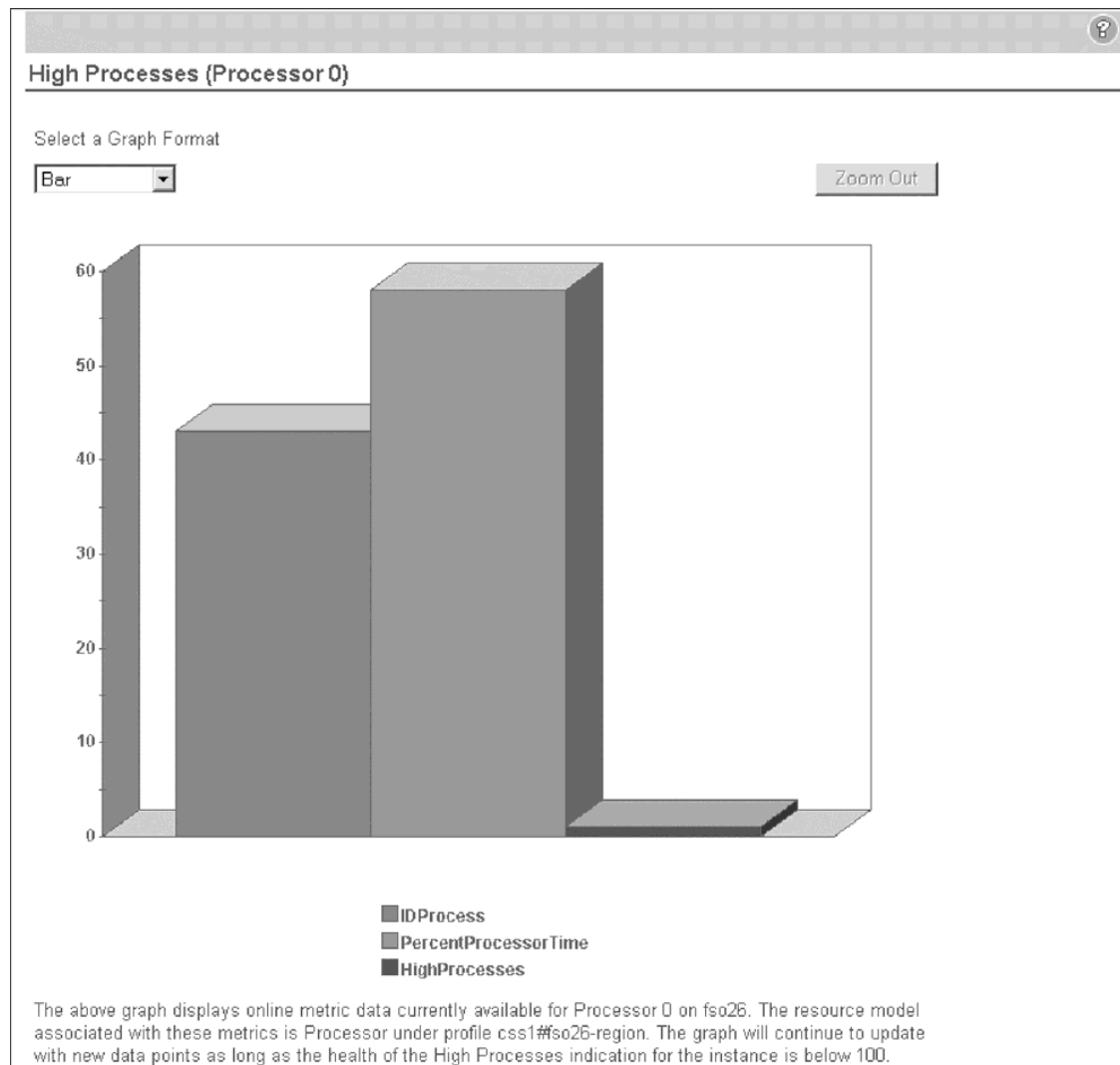
7. Click **Graph**.

For detailed description of the Historical Graph, see “Historical Data Graph” on page 105.

## Working with the Chart Views

The following sections provide you with a description of the chart views used by the Web Health Console.

### Online Data Graph



The Online Data Graph displays the current value of the metrics that make up the indication on a given instance of a resource; for example, Processor 0 on a single processor system. One data point is shown for each metric when the graph is launched. Use the **Table** format to display non-numeric metrics.

The **Online Data Graph** refresh rate and the format in which it is launched are specified on the **Graph View** tab of the **Preferences** view. You can change the style of graph displayed from the **Select a Graph Format** drop-down list.



Data is added to the graph at each refresh interval until the rollover interval for the graph is reached or the health of the instance returns to 100. The default rollover interval is 24 hours. When the health of the instance returns to 100, the graph displays a message. New data points are added if the health drops below 100 at a later time.

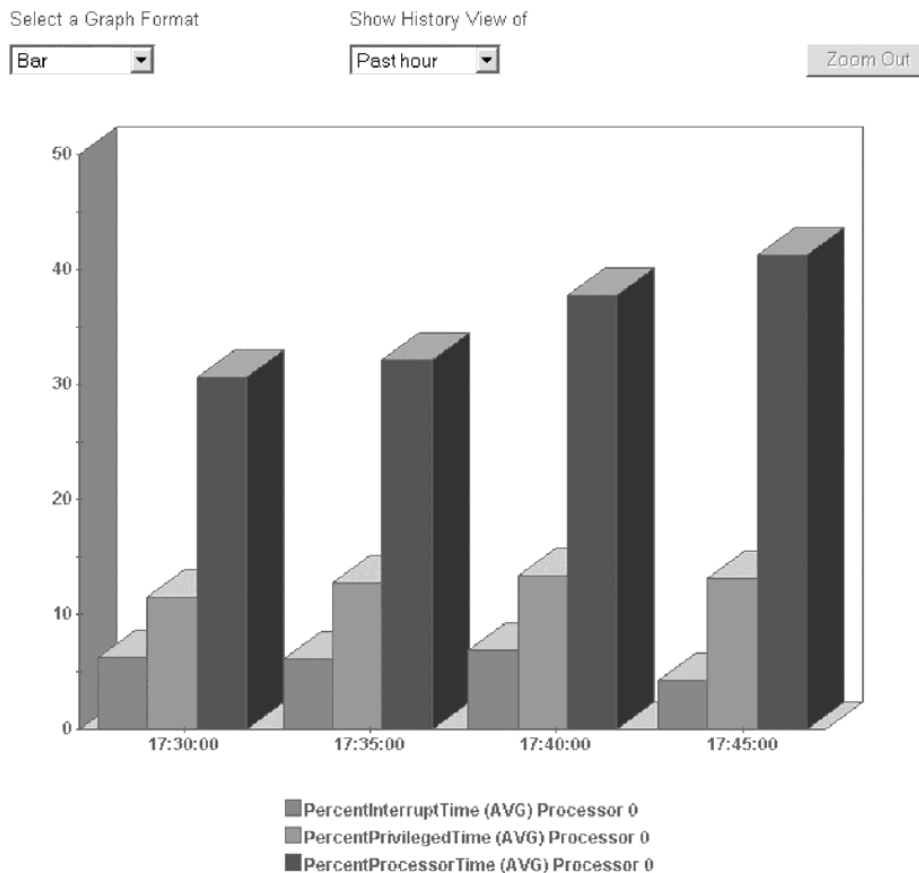
New data points are added to the right side of the graphs and the bottom of the table. When the rollover interval is reached, data points fall off the left side of the graphs and top of the table.

For non-table formats, the graph is divided into 10 zoom regions of equal size. To zoom in on a specific region of the graph, position the cursor on the desired area and click. The graph is refreshed and displays only the selected region. Click **Zoom Out** to return to the previous zoom level. For example, if you zoomed in three times you would have to click **Zoom Out** three times to return to the original graph view.

**Note:** The area on which you zoom remains at the center of the view until you zoom back out. This means that you do not see new data points being added if they are outside of the zoomed area.

## Historical Data Graph

### Processor CPU Usage



The above graph displays a history of Processor CPU Usage over the selected time period on fso26. The resource model which collected this data is Processor under profile css1##so26-region. The following metrics are shown in the graph: PercentInterruptTime (AVG), PercentPrivilegedTime (AVG), PercentProcessorTime (AVG). The values of the metrics are graphed for each of the following instances: Processor 0.

The **Historical Graph** displays recent historical data logged by a resource model on an endpoint. Metrics logged for multiple instances of a resource can be plotted on the same graph. The metrics displayed on the graph are specified in the **Historical Data** frame of the **Endpoint Health** View. See “Historical Data Frame” on page 103 for more information.

With logged data, you can use the **Historical Graph** to identify specific instances of resource problems over the past one, six, twelve, or twenty-four hours.

The graph launches in the format specified on the **Graph View** page of the **Preferences** view. You can change the style of graph displayed by using the **Select a Graph Format** drop-down list.

The **Historical Graph** can display data over the past one, six, twelve, or twenty-four hours. The default history view displays the last hour of data. You can change the time period from the **Show History View of** drop-down list. Use the Tivoli Enterprise Data Warehouse to view data older than 24 hours.

For non-table formats, the graph is divided into 10 zoom regions of equal size. To zoom in on a specific region of the graph, position the cursor on the desired area and click. The graph is refreshed and displays only the selected region. Click **Zoom Out** to return to the previous zoom level. For example, if you zoomed in three times you would have to click **Zoom Out** three times to return to the original graph view.



# Commands

This appendix describes the Tivoli Monitoring commands you can issue from the command line in the Tivoli environment, and how to use them.

## Upgraded Command Names

In Tivoli Monitoring, all commands have the prefix **wdm**. The following steps have been taken to assist the transition from the old command prefixes to the new prefixes:

- Five of the old command names have been included with this release only of the product as aliases for the new equivalent commands.
- Three of the old commands (wtmntaddrm, wtmntdefrm, wtmntrmrm) have been merged into one new command (wdmrm). The old commands have been made available with this release only, with their old unchanged syntax and options.
- Eight new commands have been added, for which aliases with the old prefix are not supplied.

Thus, you are recommended to switch to the new command names as soon as possible, and in case before any subsequent release of the product. Table 14 shows full details:

Table 14. New and old command names

New command names	Old command names, retained for this release	Obsolete merged commands, retained for this release
<b>wdmcmd</b>	wtmntcmd	
<b>wdmconfig</b>		
<b>wdmdiscovery</b>		
<b>wdmdistrib</b>		
<b>wdmdumpprf</b>		
<b>wdmeditprf</b>		
<b>wdmeng</b>	wtmnteng	
<b>wdmheartbeat</b>		
<b>wdmloadprf</b>		
<b>wdmlseng</b>	wtmntlseng	
<b>wdmmn</b>	wtmntmn	
<b>wdmmngcache</b>		
<b>wdmrm</b>		wtmntaddrm, wtmntdefrm, wtmntrmrm
<b>wdmtrceng</b>	wtmnttrceng	

The commands are described in this appendix using their new names only.

The commands in this appendix are presented in alphabetic order, and are also listed alphabetically in the index. However, what follows is a grouping of the available commands according to their primary function.

## Commands for Managing Profiles at the Tivoli Server

The following commands are available for managing profiles at the Tivoli server:

Command	Purpose	Page
<b>wdmistrib</b>	Distributes a profile to subscribers.	116
<b>wmdumpprf</b>	Writes the full details of a profile to the standard output.	118
<b>wdmeditprf</b>	Allows you to customize a profile, including all resource model details.	122
<b>wdmloadprf</b>	Loads and updates profiles at a Tivoli management region server.	133

## Command for Managing Default Resource Models at the Tivoli Server

The following command is available for managing default resource models at the Tivoli server:

Command	Purpose	Page
<b>wdmrm</b>	Adds, lists or removes a specified default resource model at the Tivoli management region server or managed node/gateway from where it is issued. It also adds the NLS catalog to an already installed default resource model.	144

## Command for Managing the Tivoli Business Systems Manager Adapter

The following command is available for managing the Tivoli Business Systems Manager Adapter:

Command	Purpose	Page
<b>wdmdiscovery</b>	Runs a bulk or delta discovery on selected endpoints at selected managed nodes/gateways. The command should only be used if the Tivoli Business Systems Manager Adapter component is installed.	114

## Commands for Managing Tivoli Monitoring at the Endpoints

The following commands are available for managing Tivoli Monitoring at the endpoints:

Command	Purpose	Page
<b>wdmcmd</b>	Stops or restarts Tivoli Monitoring on one or more endpoints from a managed node/gateway or server.	110
<b>wdmeng</b>	Stops or starts profiles or resource models at endpoints; also deletes profiles at endpoints.	129
<b>wdmlseng</b>	Returns a list and the status of all resource models that have been distributed on a specified endpoint.	135
<b>wdmtrceng</b>	Sets the trace parameters of the Tivoli Monitoring engine at the endpoint.	146

## Commands for Managing Tivoli Monitoring at Managed Nodes/Gateways

The following commands are available for managing the product at managed nodes/gateways:

Command	Purpose	Page
<b>wdmconfig</b>	Updates the configuration file of the Tivoli Business Systems Manager Adapter	111
<b>wdmheartbeat</b>	Stops or starts the 'heartbeat' monitoring of endpoints, changes its frequency and queries the status of the heartbeat processor.	131
<b>wdmmn</b>	Stops or starts selected Tivoli Monitoring processes on one or all managed nodes/gateways.	139
<b>wdmmngcache</b>	Deletes or lists part or all of the contents of the Tivoli Business Systems Manager Adapter cache.	141

## wdmcmd

Stops or restarts Tivoli Monitoring on one or more endpoints, from a managed node/gateway or server.

### Syntax

```
wdmcmd -restart {-p profile_manager#region | -e endpoint}
```

```
wdmcmd -stop {-p profile_manager#region | -e endpoint}
```

### Description

The **wdmcmd** command stops or restarts the Tivoli Monitoring endpoint engine on all the endpoints that are subscribed to the profile manager if the **-p** option is used, or only on the endpoints listed if the **-e** option is used.

### Options

**-e** *endpoint*

Specifies one or more endpoints by endpoint label: if more than one endpoint label is to be targeted, they should be separated by spaces.

**-p** *profile\_manager#region*

Specifies the profile manager whose subscribers are the target of the command. The profile manager name must always be fully qualified with the policy region name as shown in the syntax.

If you use this option in conjunction with the **-restart** option, all the profiles that were distributed to the target endpoints are restarted, including those belonging to other profile managers.

**-restart**

Restarts the endpoint engine on the endpoint or endpoints specified.

**-stop**

Stops the endpoint engine on the endpoint or endpoints specified.

### Authorization

Requires the super, senior or admin roles.

### Examples

1. The following example stops all the endpoints that belongs to the profile manager MyProfileManager.

```
wdmcmd -stop -p MyProfileManager
```

2. The following example restarts the endpoint engine on the endpoints **myEP1** and **myEP2**.

```
wdmcmd -restart -e myEP1 myEP2
```

### See Also

**wdmeng**

**wdmlseng**

**wdmtrceng**

## wdmconfig

Updates the configuration file of a managed node/gateway.

### Syntax

```
wdmconfig [-m {managed_node | @managed_nodes_file | all}]
          -D key=value [-D key=value] ...
```

### Description

The **wdmconfig** command operates on one or more selected managed nodes/gateways, and allows you to change the value of any of the configuration *keys* to the *value* you require.

### Options

**-D key=value [key=value]...**

Identifies one or more configuration *keys* to be updated and the new *value* that each key is to take. At least one key/value pair must be supplied. The keys that can be modified are as follows:

#### **transport.server.ip.address**

IP address or hostname of the CommonListener component of Tivoli Business Systems Manager, which listens out for messages from the systems being managed. For example: `transport.server.ip.address=193.202.74.21`

#### **transport.server.mqe.port**

Port number of the CommonListener component. For example:  
`transport.server.mqe.port=8082`

#### **adapter.working.dir**

Working directory that will be used by the adapter. The default, which you are recommended to use, is the Tivoli Monitoring middle layer directory (`$DBDIR/dmm1`)

#### **trace.filename**

Filename to which the trace messages from the adapter will be written. The default is `dm.trc`

#### **adapter.trace.enable**

Set this to **true** if you want to store all trace messages regarding the operations of the adapter. The messages are stored in the file identified in `trace.filename`. The default is **false**

#### **transport.trace.enable**

Set this to **true** if you want to store all messages regarding the transport of adapter-acquired data to the CommonListener. The messages are stored in the file identified in `trace.filename`. The default is **false**

#### **adapter.trace.level**

If you have enabled adapter trace messages, set this to **low**, **medium** or **high**, according to the level of details you require. The default is **low**

#### **transport.trace.level**

If you have enabled adapter trace messages, set this to **low**, **medium** or **high**, according to the level of details you require. The default is **low**

**transport.mqe.usefiller**

Set this to **true** if the managed node/gateway on which the adapter is installed is running Windows NT, 4.0, Service Pack 5; otherwise leave as the default value of **false**

**dmml\_trace\_size**

Specifies the size of the profile distribution trace in bytes; default is 500000 bytes.

**dmml\_trace\_level**

Specifies the level of the profile distribution trace from 0 (minimal) to 4 (verbose); the default is 1.

**heartbeat.send\_events\_to\_tbsm**

Set this to **true** if you want to send heartbeat events to the Tivoli Business Systems Manager; otherwise leave as the default value of **false**.

**heartbeat.send\_events\_to\_tec**

Set this to **true** if you want to send heartbeat events to the Tivoli Enterprise Console server; otherwise leave as the default value of **false**.

**heartbeat.send\_events\_to\_notice**

Set this to **true** if you want to send heartbeat events to the Tivoli Monitoring notice group; otherwise leave as the default value of **false**.

**heartbeat.reboot\_engine\_if\_down**

Set this to **true** if you want heartbeat to restart the engine that was stopped abnormally. A heartbeat event is sent, if you configured sending events.

**heartbeat.tec\_server**

If you have set **heartbeat.send\_events\_to\_tec** to **true**, enter here the name of the Tivoli Enterprise Console server.

**request\_manager.threads**

Specifies the number of threads that the Request Manager uses to handle internal requests. It is approximately the number of endpoints that can be managed simultaneously. The value can be tuned depending on the workload of the managed node on which the Request Manager runs. The default value is 10.

**request\_manager.automatic\_cancel\_frequency**

Specifies the time interval (in seconds) after which the Request Manager checks if the applications are using the requests that they have submitted. When an application is not using the requests, the Request Manager cancels the requests submitted by the application. The default value is 600 seconds.

**request\_manager.request\_expiration\_period**

Specifies the number ("x") of periods allowed for an application to retrieve data. If an application does not receive data during "x" periods, then the Request Manager cancels the request for that application. The default value is 3 periods. This means that, for an application which submitted a request with a refresh time of 10 minutes, if the application is not getting any data for 10\*3=30 minutes, then the request gets canceled.

**tbsma.jre\_root**

This parameter is set during the installation of the Tivoli Business Systems Manager Adapter (see "Installing Tivoli Business Systems Manager Adapter" on page 73, and you do not normally need to change it manually. However, if, for example, you want to install the adapter on a group of gateways using one instance



of the install action/command, you will then need to change this parameter on any gateways in the group that have JRE installed at a location different to that supplied on the Install Options dialog.

Set this parameter to the complete path of the root directory of Java Runtime Environment, 1.3.0 (excluding the directory /bin).

**Note:** In a Windows NT Workstation, if the install target path contains a directory with spaces in the name, the directory name must be specified between single quotes, as in this example:

```
D:\'Program Files'\jre
```

**Note:** No validation is performed on the values entered against a given key, so you must ensure yourself that they are valid.

**-m** {*managed\_node* | @*managed\_nodes\_file* | **all**}

Specifies the managed nodes/gateways on which the product configuration will be updated. If this option is not used the product configuration is updated on the managed node/gateway from which the command is being run.

The options are as follows:

*managed\_node*

The name of the managed node/gateway on which the command is to be performed.

@*managed\_nodes\_file*

A file containing managed node/gateway names separated by the CR/LF character; the @ symbol is an indicator that a file name follows but the symbol should not form part of the filename.

**all** All managed nodes/gateways in the Tivoli region

## Authorization

Requires the super or senior roles.

## Examples

The following example updates the configuration at all managed nodes/gateways identified in the file `my_nodes1.txt` with the ip address and port of the server where the Tivoli Business Systems Manager's CommonListener is to be found

```
wdmconfig -m @my_nodes1.txt -D transport.server.ip.address=193.202.74.21
-D transport.server.mqe.port=4068
```

## See Also

**wdmdiscovery**

**wdmheartbeat**

**wdmmngcache**

## wdmdiscovery

Runs a bulk or delta discovery on selected endpoints at selected managed nodes/gateways. The command should only be used if the Tivoli Business Systems Manager Adapter component is installed.

### Syntax

```
wdmdiscovery [-m {managed_node | @managed_nodes_file | all}] -b
  [{-e {endpoint | @endpoints_file} | -a}]
```

```
wdmdiscovery [-m {managed_node | @managed_nodes_file | all}] -d
  [{-e {endpoint | @endpoints_file} | -a}]
```

### Description

The **wdmdiscovery** command operates on a selected managed node/gateway, a list of managed nodes/gateways contained in a file or all managed nodes/gateways, and sends a bulk or a delta discovery of an endpoint, a list of endpoints contained in a file or all endpoints, to the Tivoli Business Systems Manager CommonListener.

### Options

- b** The discovery is to be bulk, in which all information on all endpoints identified by the -**e** parameter is returned
- d** The discovery is to be delta, in which information is only returned on endpoints where a change of status has taken place since the previous discovery. The status in this case is the presence or absence of the endpoint in the network. Thus, the delta discovery reports only endpoints that have come alive or have been switched off or are for some other reason unavailable in the network.

**{-e {*endpoint* | @*endpoints\_file*} | -a}**

Specifies the endpoints for which information is required. If this option is not used the discovery will be carried out on all endpoints attached to the managed node/gateway from which the command is being run.

The options are as follows:

-*e endpoint*

An endpoint label.

-**e** @*endpoints\_file*

A file containing endpoint labels separated by the CR/LF character; the @ symbol is an indicator that a file name follows but the symbol should not form part of the filename.

-**a** All endpoints attached to the defined managed nodes/gateways that gave an *alive* status at the most recent heartbeat monitoring.

**-m {*managed\_node* | @*managed\_nodes\_file* | **all**}**

Specifies the managed nodes/gateways on which the discovery will be carried out. If this option is not used the discovery is carried out on the managed node/gateway from which the command is being run.

The options are as follows:

*managed\_node*

The name of the managed node/gateway on which the command is to be performed.

**@managed\_nodes\_file**

A file containing managed node/gateway names separated by the CR/LF character; the @ symbol is an indicator that a file name follows but the symbol should not form part of the filename.

**all** All managed nodes/gateways in the Tivoli region

## Authorization

Requires the super, senior or admin roles.

## Examples

1. The following example carries out a bulk discovery on all *alive* endpoints of the managed node/gateway **MyManagedNode**:  

```
wdmdiscovery -m MyManagedNode -b -a
```
2. The following example carries out a bulk discovery on endpoint **MyEndpoint** of the managed node/gateway **MyManagedNode**:  

```
wdmdiscovery -m MyManagedNode -b -e MyEndpoint
```
3. The following example carries out a delta discovery on all endpoints of the managed node/gateway **MyManagedNode**:  

```
wdmdiscovery -m MyManagedNode -d
```

## See Also

**wdmconfig**

**wdmheartbeat**

**wdmmngcache**

---

## wmdmistrib

Distributes a profile to subscribers.

### Syntax

```
wmdmistrib -p profile_name [-D MDist2_property=value]... [-e] [-w] [-i] [-J  
JRE_location_dir] [-d] [-R] [-I][-s subscribers_file] [subscriber...]
```

### Description

This command distributes a profile to subscribers. Four types of subscribers can be specified (see option **-s**).

Tivoli Monitoring uses Multiplexed Distribution (MDist2) to perform asynchronous profile data transfers. For details on MDist2, see “Profile Distribution using MDist2” on page 56.

### Options

**-p** *profile\_name*

Specifies the name of the profile that is to be distributed.

**-R** Specifies that the profile is distributed recursively to all level of subscribers. The default value is **false**, meaning that the profile is delivered only to the next level of subscribers.

**-I** If not specified, the default allows distributions only to the current subscribers of the profile manager to which the profile belongs.

**-d** If specified, then the data is disposable, meaning that it can be removed from the repeater depot closer to the endpoint after distribution.

**-e** If specified, then error messages should be written to the log file.

**-w** If specified, then warning messages should be written to the log file.

**-i** If specified, then informational messages should be written to the log file.

**-D** *MDist2\_property=value*

Sets one or more properties to configure MDist2 for the current distribution; a separate **-D** must be specified for each property to be configured. Only the following MDist2 properties are supported:

**label** Specifies a description string for the distribution. The default value is the string *profile\_name(operation)*, where *operation* is **install**.

**pri** Specifies the priority level, which is the order in which distributions are handled by repeaters: **h** (highest priority), **m** (medium priority), or **l** (low priority). The default value is **m** (medium priority).

**send\_timeout**

Specifies the length of time in seconds a repeater will wait for a target system to receive a block of data. This timeout is used to detect network or endpoint failures. This attribute is initially set using the **wmdist -s** command (the default is 300 seconds). You can override the **wmdist -s** setting by specifying a different value here.

**result\_timeout**

Specifies the length of time in seconds a repeater will wait for the distribution method at the endpoint to return the distribution results.

**deadline**

The date on which a distribution expires, that is, when it fails for unavailable target systems. It is specified in the format "*mm/dd/yyyy hh:mm*"

**-J "JRE\_location\_dir"**

The complete path of the directory where Java Runtime Environment (JRE) is located on the CD-ROM (excluding the platform directory). The path can be included within double quotes.

**-s subscribers\_file**

A file containing the profile's subscriber names, separated by the CR/LF character. Four types of subscriber can be specified, in the following format:

- Endpoint::*label*
- ProfileManager::*label*
- AppProxy::*label*
- AppSvcs::*label*

where *label* is the name of the endpoint, profile manager, application proxy, or application service. Endpoint is the default if no subscriber type is specified.

**subscriber**

The name of any subscriber to which the profile is to be distributed. If more than one subscriber is specified, separate the names by blanks. If no subscribers are specified then the profile is distributed to all current subscribers of the profile manager to which the profile belongs. This option must be specified as the last one in the command syntax.

**Authorization**

Requires the admin, super, or senior role.

**Examples**

1. The following example distributes the profile myProfile to the application proxy myProxy and to the endpoint myEndpoint:

```
wdmdistrib -p myProfile AppProxy::myProxy myEndpoint
```

2. The following example distributes the profile myProfile to the endpoint myEndpoint. The JRE from the CD-ROM drive X in the directory \Tools\Jre will be installed at the endpoint if not present:

```
wdmdistrib -p myProfile -J "X:\Tools\Jre" myEndpoint
```

3. The following example distributes the profile myProfile, recursively, to the subscribers listed in the subscribersFilename file in the directory myDir on drive D and also to the profile manager myProfileManager:

```
wdmdistrib -p myProfile -R -l -s "D:\myDir\subscribersFilename"  
ProfileManager::myProfileManager
```

**See Also**

**wdmconfig**

**wmdist** (see the *Tivoli Management Framework Reference* manual)

## wdmdumpprf

Writes the full details of a profile to the standard output.

### Syntax

**wdmdumpprf -P** *profile#region* [-x]

### Description

The **wdmdumpprf** command writes the full details of the selected profile to the standard output, in Tivoli Management Framework format or XML format. The output can then be saved and edited and reloaded as a new or amended profile using the **wdmloadprf** command.

An example output from this command of a profile containing a single resource model (in XML format), is as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Tmw2kProfile [
  <!ELEMENT Tmw2kProfile (GenProfileInfo, ResModelInfo)*>
  <!--ATTLIST Tmw2kProfile Version          CDATA #REQUIRED-->
  <!ELEMENT GenProfileInfo (#PCDATA)>
  <!--ATTLIST GenProfileInfo ProfileName   CDATA #REQUIRED-->
  <!--ATTLIST GenProfileInfo EnableTBSMfeeding CDATA #REQUIRED-->
  <!--ATTLIST GenProfileInfo Tec          CDATA #REQUIRED-->
  <!--ATTLIST GenProfileInfo TecEventServer CDATA #REQUIRED-->
  <!--ATTLIST GenProfileInfo TecServerPort  CDATA #REQUIRED-->
  <!ELEMENT ResModelInfo (SchedulingInfo,LoggingInfo,Property)*>
  <!--ATTLIST ResModelInfo Enabled         CDATA #REQUIRED-->
  <!--ATTLIST ResModelInfo ModelName      CDATA #REQUIRED-->
  <!--ATTLIST ResModelInfo Zipfile        CDATA #REQUIRED-->
  <!--ATTLIST ResModelInfo Platform       CDATA #REQUIRED-->
  <!--ATTLIST ResModelInfo MinorVersion   CDATA #REQUIRED-->
  <!--ATTLIST ResModelInfo MajorVersion   CDATA #REQUIRED-->
  <!--ATTLIST ResModelInfo CycleTime      CDATA #REQUIRED-->
  <!ELEMENT SchedulingInfo (String)*>
  <!--ATTLIST SchedulingInfo StartDate    CDATA #REQUIRED-->
  <!--ATTLIST SchedulingInfo StopDate     CDATA #REQUIRED-->
  <!ELEMENT String (#PCDATA)>
  <!ELEMENT LoggingInfo (DataLoggingInfo, EventsLoggingInfo)>
  <!ELEMENT DataLoggingInfo (#PCDATA)>
  <!--ATTLIST DataLoggingInfo EnableLogging CDATA #REQUIRED-->
  <!--ATTLIST DataLoggingInfo LoggingPeriod CDATA #REQUIRED-->
  <!--ATTLIST DataLoggingInfo AggregateData CDATA #REQUIRED-->
  <!--ATTLIST DataLoggingInfo AggregationPeriod CDATA #REQUIRED-->
  <!--ATTLIST DataLoggingInfo WantMin       CDATA #REQUIRED-->
  <!--ATTLIST DataLoggingInfo WantMax      CDATA #REQUIRED-->
  <!--ATTLIST DataLoggingInfo WantAvg      CDATA #REQUIRED-->
  <!ELEMENT EventsLoggingInfo (#PCDATA)>
  <!--ATTLIST EventsLoggingInfo EnableLogging CDATA #REQUIRED-->
  <!--ATTLIST EventsLoggingInfo LoggingPeriod CDATA #REQUIRED-->
  <!ELEMENT Property (#PCDATA)>
  <!--ATTLIST Property Name                CDATA #REQUIRED-->
  <!--ATTLIST Property Value              CDATA #REQUIRED-->
  <!ELEMENT Parameter (String)*>
  <!--ATTLIST Parameter Name              CDATA #REQUIRED-->
  <!--ATTLIST Parameter Type              CDATA #REQUIRED-->
  <!ELEMENT EventAggregInfo (Action*,Task*)>
  <!--ATTLIST EventAggregInfo Name        CDATA #REQUIRED-->
  <!--ATTLIST EventAggregInfo Occurrences CDATA #REQUIRED-->
  <!--ATTLIST EventAggregInfo Holes       CDATA #REQUIRED-->
  <!--ATTLIST EventAggregInfo SendToTBSM CDATA #REQUIRED-->
  <!--ATTLIST EventAggregInfo Severity    CDATA #REQUIRED-->
  <!--ATTLIST EventAggregInfo SendTec     CDATA #REQUIRED-->
  <!--ATTLIST EventAggregInfo ExecuteAction CDATA #REQUIRED-->
  <!ELEMENT Action (#PCDATA)>
```

```

    <!ATTLIST Action Methname      CDATA #REQUIRED>
    <!ATTLIST Action Retry         CDATA #REQUIRED>
  <!ELEMENT Task (#PCDATA)>
    <!ATTLIST Task Name           CDATA #REQUIRED>
    <!ATTLIST Task Lib            CDATA #REQUIRED>
    <!ATTLIST Task Args           CDATA #REQUIRED>
  ]>
<Tmw2kProfile Version = "41000">
  <GenProfileInfo
    ProfileName = "myprofile#ucase11i-region"
    EnableTBSMfeeding = "FALSE"
    Tec = "None"
    TecEventServer = ""
    TecServerPort = "5529"/>
  <ResModelInfo
    ModelName = "TMW_Processor"
    Enabled = "TRUE"
    Zipfile = "TMW_Processor.zip"
    Platform = "w32-ix86"
    MajorVersion = "1"
    MinorVersion = "0"
    CycleTime = "60">
    <SchedulingInfo
      StartDate = "0"
      StopDate = "0">
    </SchedulingInfo>
    <LoggingInfo>
      <DataLoggingInfo
        EnableLogging = "FALSE"
        LoggingPeriod = "720"
        AggregateData = "TRUE"
        AggregationPeriod = "15"
        WantMin = "FALSE"
        WantMax = "FALSE"
        WantAvg = "TRUE">
      </DataLoggingInfo>
      <EventsLoggingInfo
        EnableLogging = "FALSE"
        LoggingPeriod = "300">
      </EventsLoggingInfo>
    </LoggingInfo>
    <Property
      Name = "HighCPUUsageProcess"
      Value = "40">
    </Property>
    <Property
      Name = "HighInterruptsSec"
      Value = "600">
    </Property>
    <Property
      Name = "HighCPUUsageUserPriv"
      Value = "70">
    </Property>
    <Property
      Name = "HighCPUUsageInterrupt"
      Value = "20">
    </Property>
    <Property
      Name = "HighCPUUsage"
      Value = "80">
    </Property>
    <Property
      Name = "HighPercentUsageDelta"
      Value = "5">
    </Property>
    <Property
      Name = "TotalCPUsMod"

```

```
    Value = "10">
  </Property>
  <EventAggregInfo
    Name = "TMW_ProcessorBusy"
    Occurrences = "20"
    Holes = "1"
    SendToTBSM = "FALSE"
    Severity = "CRITICAL"
    SendTec = "TRUE"
    ExecuteAction = "FALSE">
  </EventAggregInfo>
  <EventAggregInfo
    Name = "TMW_HighProcesses"
    Occurrences = "10"
    Holes = "2"
    SendToTBSM = "FALSE"
    Severity = "CRITICAL"
    SendTec = "TRUE"
    ExecuteAction = "FALSE">
  </EventAggregInfo>
  <EventAggregInfo
    Name = "TMW_HWKeepingCPUBusy"
    Occurrences = "15"
    Holes = "3"
    SendToTBSM = "FALSE"
    Severity = "WARNING"
    SendTec = "TRUE"
    ExecuteAction = "FALSE">
  </EventAggregInfo>
  <EventAggregInfo
    Name = "TMW_CPUCantKeepUpWithHW"
    Occurrences = "15"
    Holes = "3"
    SendToTBSM = "FALSE"
    Severity = "CRITICAL"
    SendTec = "TRUE"
    ExecuteAction = "FALSE">
  </EventAggregInfo>
  <EventAggregInfo
    Name = "TMW_BusyHardware"
    Occurrences = "15"
    Holes = "3"
    SendToTBSM = "FALSE"
    Severity = "WARNING"
    SendTec = "TRUE"
    ExecuteAction = "FALSE">
  </EventAggregInfo>
  <EventAggregInfo
    Name = "TMW_HighPercentUsageDelta"
    Occurrences = "10"
    Holes = "3"
    SendToTBSM = "FALSE"
    Severity = "MINOR"
    SendTec = "FALSE"
    ExecuteAction = "FALSE">
  </EventAggregInfo>
</ResModelInfo>
</Tmw2kProfile>
```

## Options

### **-P** *profile#region*

Defines the profile to be written to the standard output. The profile name must always be fully qualified with the policy region name in the syntax shown.



**Note:** To obtain a listing of the complete names of all the profiles that are on an endpoint, issue the following command:

```
wdmlseng -e endpoint
```

- x Dumps the profile in XML format. The default format is Tivoli Management Framework.

## Authorization

Requires the super, senior or admin roles.

## Examples

The following example writes the full details of the profile **MyProfile** in the region **MyRegion** to the standard output.

```
wdmdumpprf -P MyProfile#MyRegion
```

## See Also

**wdmeditprf**

**wdmloadprf**

## wdmeditprf

Allows you to edit the definition of a profile, including all resource model details.

### Syntax

```

wdmeditprf -P profile#region -add resource_model
wdmeditprf -P profile#region -add resource_model -c cycle_time
wdmeditprf -P profile#region -add resource_model
  -t threshold threshold_value [-t threshold threshold_value]...
wdmeditprf -P profile#region -add resource_model -e event [-o occurrences] [-h holes]
  [-SendTec | -NoSendTec] [-SendTBSM | -NoSendTBSM] [-severity severity_type]
  [-e event [-o occurrences] [-h holes] [-SendTec | -NoSendTec]
  [-SendTBSM | -NoSendTBSM] [-severity severity_type]...]
wdmeditprf -P profile#region -add resource_model -Log [-LogEnable | -LogDisable]
  [-p period] [-Agg {yes | no}] [-ap aggregation_period] [-Min | -NoMin]
  [-Max | -NoMax] [-Avg | -NoAvg]
wdmeditprf -P profile#region -add resource_model
  -AddPar parameter_name parameter_value
  [-AddPar parameter_name parameter_value]...
wdmeditprf -P profile#region -add resource_model -enable | -disable
wdmeditprf -P profile#region -edit resource_model -c cycle_time
wdmeditprf -P profile#region -edit resource_model
  -t threshold threshold_value [-t threshold threshold_value]...
wdmeditprf -P profile#region -edit resource_model -e event [-o occurrences] [-h holes]
  [-SendTec | -NoSendTec] [-SendTBSM | -NoSendTBSM] [-severity severity_type]
  [-e event [-o occurrences] [-h holes] [-SendTec | -NoSendTec]
  [-SendTBSM | -NoSendTBSM] [-severity severity_type]...]
wdmeditprf -P profile#region -edit resource_model -Log [-LogEnable | -LogDisable]
  [-p period] [-Agg {yes | no}] [-ap aggregation_period] [-Min | -NoMin]
  [-Max | -NoMax] [-Avg | -NoAvg]
wdmeditprf -P profile#region -edit resource_model
  -AddPar parameter_name parameter_value
  [-AddPar parameter_name parameter_value]...
wdmeditprf -P profile#region -edit resource_model
  -DelPar parameter_name parameter_value
  [-DelPar parameter_name parameter_value]...
wdmeditprf -P profile#region -edit resource_model -enable | -disable
wdmeditprf -P profile#region -list
wdmeditprf -P profile#region -print resource_model [-t] [-e] [-Log] [-c] [-par]
  [-schedule]
wdmeditprf -P profile#region -remove resource_model
wdmeditprf -P profile#region -TBSM {yes | no}
wdmeditprf -P profile#region -Tec {secure -S server | unsecure -S server -p port | no}

```

## Description

The **wdmeditprf** command lets you change various attributes of a profile:

- Resource models can be added with default values or with value supplied by you
- Resource model details can be edited
- Profiles or selected resource model details can be printed
- The destination monitors for events generated by the profile can be defined

Resource models can be added using all of the default values supplied, which are documented in *IBM Tivoli Monitoring: Resource Model Reference*. Alternatively you can add a model with one or more values modified to suit your circumstances. You can also edit any of the details of an existing resource model. The various options shown in the *Syntax* section, above, and defined in the *Options* section, below, are non-exclusive; thus you could issue a single command to add a resource model modifying the default values of, for example, the cycle time and an event. You could then issue a separate command to edit, for example, the logging details and two thresholds.

## Options

The options to the **wdmeditprf** command must be specified in the correct sequence:

1. **-P** *profile#region*
2. **-add**, **-edit**, **-list**, **-print**, **-remove**, **-TBSM** or **-Tec**
3. *resource\_model* (if appropriate)
4. Arguments to the **-add** or **-edit** options: these can be entered in any order

The option details are as follows:

### **-P** *profile#region*

The profile on which the actions of this command are to be performed.

### **-add** *resource\_model*

Adds the named resource model to the profile. See *IBM Tivoli Monitoring: Resource Model Reference* for details of the correct resource model names to use.

The command may optionally be followed by one or more of the resource model arguments, that will modify the default values of the named resource model; only the values that need changing need to be specified. Details of these arguments are given following the **-Tec** option.

### **-edit** *resource\_model*

Specifies that you want to edit the named resource model.

The command may optionally be followed by one or more of the resource model arguments, that will modify the default values of the named resource model; only the values that need changing need to be specified. Details of these arguments are given following the **-Tec** option.

### **-list**

Lists a summary of the profile details to the standard output. The output format is as follows:

Resource Model	Enable
DMXFileSystem	YES
DMXMemory	YES
DMXProcess	YES
TMW_ParamEventLog	NO

**-print** *resource\_model*

Prints basic information about the named resource model. Additional information can be obtained by specifying one or more of the following arguments:

- c**     Additionally prints cycle time information.
- e**     Additionally prints event information.
- Log**   Additionally prints data logging information.
- par**    Additionally prints parameter information (for parametric resource models).
- schedule**  
          Additionally prints schedule information.
- t**     Additionally prints threshold information.

**-remove** *resource\_model*

Removes the named resource model from the profile.

**-TBSM**

Specifies whether events for this profile are to be sent to the Tivoli Business Systems Manager. The options are:

- yes**     Specifies that events will be sent to the Tivoli Enterprise Console server.
- no**     Specifies that no events will be sent to the Tivoli Enterprise Console server. If this option is selected it will ignore any requests to “Send to TBSM” that you specify or may have specified for individual indications.

**-Tec**

Specifies whether events for this profile are to be sent to the Tivoli Enterprise Console server, and identifies the server. The options are:

**secure -S** *server*

Specifies that the Tivoli Enterprise Console server is a secure server, and provides its server name.

**unsecure -S** *server -p* *port*

Specifies that the Tivoli Enterprise Console server is an unsecure server, and provides its server name and port number.

- no**     Specifies that no events will be sent to the Tivoli Enterprise Console server. If this option is selected it will ignore any requests to “Send to TEC” that you specify or may have specified for individual indications.

**Arguments to the -add and -edit options**

The **-add** and **-edit** options can take any of the following arguments, in any order. Only the specified values will be changed:

**-AddPar** *parameter\_name* *parameter\_value*

Defines the values of the parameters for parametric resource models. Each parameter is of one of four types. Details of the parameter names and types for each parametric resource model can be found in the *IBM Tivoli Monitoring: Resource Model Reference*. The parameter values are added using the **-AddPar** option according to the parameter type, as follows:

■ **Numeric:**

A list of numeric values. The **-AddPar** option appends the value supplied in *parameter\_value* to the existing values in the numeric parameter.

■ **String:**

A list of strings. The **-AddPar** option appends the string supplied in *parameter\_value* to the existing values in the string parameter.

- **Boolean list:**

A boolean list of pre-defined non-exclusive values. The **-AddPar** option switches the boolean list value supplied in *parameter\_value* to the *true* state. To switch it to the *false* state use the **-DelPar** option.

- **Choice list:**

A choice list of pre-defined exclusive alternatives. The **-AddPar** option selects the choice list value supplied in *parameter\_value*; the previously selected value will automatically be deselected when the command is performed.

**-c** *cycle\_time*

Changes the default value of the cycle time of a resource model. The value is supplied as a number of seconds.

**-DelPar** *parameter\_name parameter\_value*

Changes the values of the parameters for parametric resource models. Each parameter is of one of four types. Details of the parameter names and types for each parametric resource model can be found in the *IBM Tivoli Monitoring: Resource Model Reference*. The parameter values are changed using the **-DelPar** option according to the parameter type, as follows:

- **Numeric:**

A list of numeric values. The **-DelPar** option deletes the value supplied in *parameter\_value* from the existing values in the numeric parameter.

- **String:**

A list of strings. The **-DelPar** option deletes the string supplied in *parameter\_value* from the existing values in the string parameter.

- **Boolean list:**

A boolean list of pre-defined non-exclusive values. The **-DelPar** option switches the boolean list value supplied in *parameter\_value* to the *false* state. To switch it to the *true* state use the **-AddPar** option.

- **Choice list:**

This type of parameter cannot be modified with the **-DelPar** option; use the **-AddPar** instead.

**-disable**

Disables the resource model for the defined profile.

**-e** *event*

Changes the default value for a named event. Any number of events can be defined. The specific values that can be changed are the following:

- **-h** *holes:*

Defines the number of consecutive holes (cycles where an indication does not happen) that can interrupt the count of consecutive occurrences without zeroing that count.

- **-o** *occurrences*

Defines the number of consecutive occurrences (cycles where an indication happens) that are needed to trigger the event.

- **-NoSendTBSM**

Defines that the event is not to be sent to Tivoli Business Systems Manager.

- **-NoSendTec**

- The event is not to be sent to the Tivoli Enterprise Console server.
  - **-SendTBSM**  
Defines that the event is to be sent to Tivoli Business Systems Manager.
  - **-SendTec**  
Defines that the event is to be sent to the Tivoli Enterprise Console server.
  - **-severity severity\_type**  
The severity type of the event must be one of the following: FATAL, CRITICAL, WARNING, HARMLESS, MINOR.
- enable**  
Enables the resource model for the defined profile.
- Log** Changes the default values for the data logging details. The specific values that can be changed are the following:
- **-Agg {yes | no}**  
Determines whether logging data will be aggregated (**yes**) or not (**no**).
  - **-ap aggregation\_period**  
Defines the period for which data is aggregated, in the format HH:MM; max 24:00 hours.
  - **-Avg**  
Defines that the average of the values encountered during the aggregation period will be logged.
  - **-LogDisable**  
Defines that the data logging feature is to be disabled.
  - **-LogEnable**  
Defines that the data logging feature is to be enabled.
  - **-Max**  
Defines that the highest value encountered during the aggregation period will be logged.
  - **-Min**  
Defines that the lowest value encountered during the aggregation period will be logged.
  - **-NoAvg**  
Defines that the average of the values encountered during the aggregation period will not be logged (see option **-Avg**)
  - **-NoMax**  
Defines that the highest value encountered during the aggregation period will not be logged (see option **-Max**)
  - **-NoMin**  
Defines that the lowest value encountered during the aggregation period will not be logged (see option **-Min**)
  - **-p period**  
Defines the historical duration of data in the data logging database, in the format HH:MM; max 24:00 hours.
- t threshold threshold\_value**  
Defines the value for a named threshold; any number of thresholds can be defined.

## Authorization

Requires the super, senior or admin roles.

## Examples

- The following example adds the resource model **MyResourceModel** to the profile **MyProfile** in the region **MyRegion**, using all of the default values:
 

```
wdmeditprf -P MyProfile#MyRegion -add MyResourceModel
```
- The following example adds the resource model **MyResourceModel** to the profile **MyProfile** in the region **MyRegion**, changing a selection of the defaults (the changed values are shown on separate lines for clarity, but would normally be concatenated in a single string):
 

```
wdmeditprf -P MyProfile#MyRegion -add MyResourceModel
-t MyThreshold 60
-e MyEvent1 -o 5 -h 2 -SendTec -severity WARNING -NoSendTBSM
-Log -LogEnable -Agg yes -ap 00:30 -Min -NoMax -Avg -p 12:00
-c 120
-enable
```
- The following example edits the parametric resource model **MyParamEventLog** in the profile **MyProfile** in the region **MyRegion**, changing a selection of the defaults:
 

```
wdmeditprf -P MyProfile#MyRegion -edit MyParamEventLog
-AddPar EventID 2034 -AddPar Source Win2K -DelPar Source WinNT
-AddPar EventSeverity Warning -DelPar EventSeverity Information
-AddPar FilterType and
```
- The following example lists a summary of the profile **MyProfile** in the region **MyRegion** on the standard output:
 

```
wdmeditprf -P MyProfile#MyRegion -list
```
- The following example prints basic information about the resource model **MyResourceModel** in the profile **MyProfile** in the region **MyRegion**:
 

```
wdmeditprf -P MyProfile#MyRegion -print MyResourceModel
```
- The following example prints information about the resource model **MyResourceModel** in the profile **MyProfile** in the region **MyRegion**, including information on the cycle time, the events, the data logging and the parameters:
 

```
wdmeditprf -P MyProfile#MyRegion -print MyResourceModel -c -e -Log -par
```
- The following example removes the resource model **MyResourceModel** from the profile **MyProfile** in the region **MyRegion**:
 

```
wdmeditprf -P MyProfile#MyRegion -remove MyResourceModel
```
- The following example enables the sending of events to Tivoli Business Systems Manager for the profile **MyProfile** in the region **MyRegion**:
 

```
wdmeditprf -P MyProfile#MyRegion -TBSM yes
```
- The following example disables the sending of events to Tivoli Business Systems Manager for the profile **MyProfile** in the region **MyRegion**:
 

```
wdmeditprf -P MyProfile#MyRegion -TBSM no
```
- The following example defines the secure Tivoli Enterprise Console server **EventServer** for the profile **MyProfile** in the region **MyRegion**:
 

```
wdmeditprf -P MyProfile#MyRegion -Tec secure -S EventServer
```
- The following example defines the insecure Tivoli Enterprise Console server **tecinserv** at port **8080** for the profile **MyProfile** in the region **MyRegion**:

```
wdmeditprf -P MyProfile#MyRegion -Tec unsecure  
-S tecinserv.rome.tivoli.com -p 8080
```

12. The following example disables the sending of events to the Tivoli Enterprise Console server for the profile **MyProfile** in the region **MyRegion**:

```
wdmeditprf -P MyProfile#MyRegion -Tec no
```

## See Also

**wmdumpprf**

**wmloadprf**



## wdmeng

Stops or starts profiles or resource models at endpoints; also deletes profiles at endpoints.

### Syntax

```
wdmeng -e endpoint [-p profile#region] resource_model -start
```

```
wdmeng -e endpoint [-p profile#region] resource_model -stop
```

```
wdmeng -e endpoint -p profile#region -start
```

```
wdmeng -e endpoint -p profile#region -stop
```

```
wdmeng -e endpoint -p profile#region -delete
```

### Description

The **wdmeng** command allows you to stop and start the Tivoli Monitoring engine for a specific profile or resource model at a specified endpoint; it also allows you to delete a profile at a specific endpoint.

### Options

#### *endpoint*

Defines the endpoint on which the command is to be performed; only one endpoint can be specified in each use of the command.

#### **-delete**

Deletes the named profile.

**Note:** To delete a resource model at an endpoint, you should disable the resource model in the profile at the Tivoli server using the **wdmeditprf** command and distribute the new profile to the endpoint, using this command to delete the old profile.

#### **-p profile#region**

Defines the profile on which the command's actions will be carried out. If the profile is not specified, the command is performed on all profiles at the named endpoint. The profile name must always be fully qualified with the policy region name in the syntax shown.

**Note:** To obtain a listing of the complete names of all the profiles that are on an endpoint, issue the following command:

```
wdmlseng -e endpoint
```

#### *resource model*

Defines the resource model to be started or stopped.

#### **-start**

Starts the named resource model or profile.

#### **-stop**

Stops the named resource model or profile.

### Authorization

Requires the super, senior or admin roles.

### Examples

1. The following example starts the resource model **MyResourceModel** on the endpoint **MyEndpoint** in all profiles.

```
wdmeng -e MyEndpoint MyResourceModel -start
```

2. The following example stops the resource model **MyResourceModel** on the endpoint **MyEndpoint** in all profiles.

```
wdmeng -e MyEndpoint MyResourceModel -stop
```

3. The following example starts the resource model **MyResourceModel** in the profile **MyProfile** in policy region **MyRegion** on the endpoint **MyEndpoint**.

```
wdmeng -e MyEndpoint -p MyProfile#MyRegion MyResourceModel -start
```

4. The following example stops the resource model **MyResourceModel** in the profile **MyProfile** in policy region **MyRegion** on the endpoint **MyEndpoint**.

```
wdmeng -e MyEndpoint -p MyProfile#MyRegion MyResourceModel -stop
```

5. The following example starts the profile **MyProfile** in policy region **MyRegion** on the endpoint **MyEndpoint**.

```
wdmeng -e MyEndpoint -p MyProfile#MyRegion -start
```

6. The following example stops the profile **MyProfile** in policy region **MyRegion** on the endpoint **MyEndpoint**.

```
wdmeng -e MyEndpoint -p MyProfile#MyRegion -stop
```

7. The following example deletes the profile **MyProfile** in policy region **MyRegion** on the endpoint **MyEndpoint**.

```
wdmeng -e MyEndpoint -p MyProfile#MyRegion -delete
```

## See Also

**wdmcmd**

**wdmlseng**

**wdmtrceng**

## wdmheartbeat

Stops or starts the ‘heartbeat’ monitoring of endpoints, changes its frequency and queries the status of the heartbeat processor.

### Syntax

```
wdmheartbeat [-m {managed_node | @managed_nodes_file | all}] [-s frequency | -t] [-q]
```

### Description

The **wdmheartbeat** command operates on selected managed nodes/gateways and starts or stops the ‘heartbeat’, which is a periodic monitoring by the Tivoli Business Systems Manager Adapter component at a managed node/gateway of all attached endpoints to determine certain basic status information. In addition, this command can also be used to change the frequency of the heartbeat, in other words the period elapsing between monitoring requests.

### Options

**-m** {*managed\_node* | @*managed\_nodes\_file* | **all**}

Specifies the managed nodes/gateways on which the heartbeat command is to be implemented. If this option is not used the heartbeat command is implemented on the managed node/gateway from which the command is being run.

The options are as follows:

*managed\_node*

The name of the managed node/gateway on which the command is to be performed.

@*managed\_nodes\_file*

A file containing managed node/gateway names separated by the CR/LF character; the @ symbol is an indicator that a file name follows but the symbol should not form part of the filename.

**all** All managed nodes/gateways in the Tivoli region

**-q** Queries the status of the heartbeat processor. An example of the output is as follows:

```
Processing ManagedNode mcrudele...
HeartBeat processor status: STARTED, time interval: 60
Processing ManagedNode dmw2k2...
HeartBeat processor status: STARTED, time interval: 60
Warning: DM_Advanced_Edition feature not installed
on the Managed Node 'boccaccio'. Skipping...
```

**-s** *frequency*

This option starts the heartbeat on the defined managed nodes/gateways with the frequency indicated in seconds. If the heartbeat is already running, this option changes the heartbeat frequency to that specified.

**-t** This option stops the heartbeat with immediate effect on the defined managed nodes/gateways. The heartbeat must be running for this command to take effect.

### Authorization

Requires the super, senior or admin roles.

### Examples

1. The following example stops the heartbeat on the managed node/gateway **MyManagedNode**:

```
wdmheartbeat -m MyManagedNode -t
```

2. The following example starts the heartbeat on the managed node/gateway **MyManagedNode**, the monitoring taking place every 180 seconds:

```
wdmheartbeat -m MyManagedNode -s 180
```

3. The following example queries the status of the heartbeat on all managed node/gateways in the Tivoli management region:

```
wdmheartbeat -m all -q
```

## See Also

**wdmconfig**

**wdmdiscovery**

**wdmmngcache**

## wdmloadprf

Loads new and updates old profiles at a Tivoli management region server (Tivoli server).

### Syntax

```
wdmloadprf -f profile_filename -g profile_manager#region [-P profile#region]
[-k | -m | -s] [-x]
```

### Description

The **wdmloadprf** command makes new profiles available on the Tivoli server. If the command identifies an existing profile the command will update the profile, requiring you to choose whether to merge the new resource models with the existing ones, substitute the existing resource models for the new ones or keep the existing resource models.

If you use this option to move profiles between Tivoli servers, you should make sure that the resource models defined in the profile have already been loaded onto the receiving Tivoli server (using the **wdmrm** command), before loading the profile.

The input is a file created by using the **wmdumpprf** command.

### Options

**-f** *profile\_filename*

Identifies the file containing the new profile. The name should include the full path from the \$BINDIR directory. The file should be in Tivoli Management Framework format, unless the **-x** option is used to load a file in XML format (this option available only in Windows systems).

**-g** *profile\_manager#region*

Identifies the profile manager to which the new profile is to belong or to which the profile to be updated belongs. The profile manager name must always be fully qualified with the policy region name in the syntax shown.

**-P** *profile#region*

Defines the profile to be added or updated. If the profile is not specified, the profile name contained in the XML file will be used. The profile name must always be fully qualified with the policy region name in the syntax shown.

**Note:** To obtain a listing of the complete names of all the profiles that are on an endpoint, issue the following command:

```
wdmlseng -e endpoint
```

[**-k** | **-m** | **-s**]

Defines how the model is updated. If none of the following options is defined, a new profile is created from the input information, provided that the supplied profile name does not already exist, in which case the command fails. The mutually exclusive updating options are as follows:

**-k** If a profile already exists with the supplied name, the **-k** option leaves all resource models in the existing profile as they are, only changing the profile parameters that are external to the resource models.

**-m** If a profile already exists with the supplied name, the **-m** option performs the following actions on the resource models in the existing profile, by comparing the names of the resource models contained therein:

- A resource model in the XML file that does not match a resource model in the existing profile, is added to the existing profile.

- A resource model in the XML file that matches a resource model in the existing profile, overwrites the resource model in the existing profile.
  - A resource model in the existing profile that does not match a resource model in the XML file is not changed.
- s If a profile already exists with the supplied name, the –s option substitutes the entire profile with that defined in the XML file.
- x This option can only be used on Windows systems. It determines that the file to be loaded is in XML format, instead of the default Tivoli Management Framework format.

## Authorization

Requires the super, senior or admin roles.

## Examples

1. The following example loads the profile stored in the file **MyXMLFile** in the **MyProfiles** directory and adds it to the profile manager **MyProfileManager** in the region **MyRegion**. The profile name will be determined from the contents of the XML file:  

```
wdmloadprf -f MyProfiles\MyXMLFile -g MyProfileManager#MyRegion
```
2. The following example loads the profile stored in the file **MyXMLFile** in the directory **MyProfiles** and adds it to the profile manager **MyProfileManager** in the region **MyRegion**, naming it **MyProfileName**:  

```
wdmloadprf -f MyProfiles\MyXMLFile -g MyProfileManager#MyRegion  
-P MyProfileName#MyRegion
```
3. The following example updates an existing profile identified in the file **MyXMLFile** in the **MyProfiles** directory and belonging to the profile manager **MyProfileManager** in the region **MyRegion**, and merges its resource models with those contained in the XML file:  

```
wdmloadprf -f MyProfiles\MyXMLFile -g MyProfileManager#MyRegion -M
```
4. The following example updates the profile **MyProfileName** in the region **MyRegion**, and belonging to the profile manager **MyProfileManager** by substituting its contents with those stored in the file **MyXMLFile** in the directory **MyProfiles**:  

```
wdmloadprf -f MyProfiles\MyXMLFile -g MyProfileManager#MyRegion  
-P MyProfileName#MyRegion -s
```
5. The following example updates the profile **MyProfileName** in the region **MyRegion**, and belonging to the profile manager **MyProfileManager** by keeping its resource models, only updating the profile parameters external to the resource models with those contained in the file **MyXMLFile** in the directory **MyProfiles**:  

```
wdmloadprf -f MyProfiles\MyXMLFile -g MyProfileManager#MyRegion  
-P MyProfileName#MyRegion -k
```

## See Also

**wdmdumpprf**

**wdmeditprf**

## wdmlseng

Returns a list and the status of all resource models that have been distributed on a specified endpoint.

### Syntax

```
wdmlseng -e endpoint [-verbose] [-xml]
```

```
wdmlseng -e endpoint -p profile#region [-verbose] [-xml]
```

```
wdmlseng -e endpoint -p profile#region -r resource_model [-verbose] [-xml]
```

```
wdmlseng -e endpoint -p profile#region -r resource_model -i indication [-verbose] [-xml]
```

### Description

This command returns a list of all the resource models that have been distributed on the specified endpoint. The status of each resource model is also displayed. Some status codes are standard; others are defined by the user in the Tivoli Monitoring Workbench, in which case they are accompanied by a user-defined *reason code*.

The status can be one of the following:

#### Disabled

The resource model has been distributed but not started.

#### Error

An error has occurred with the Tivoli Monitoring engine, possibly due to a failure in the resource monitoring agent.

#### Failed *reason code*

A resource model has failed.

#### Failed after recovery *reason code*

A resource model has failed after the recovery operation has tried running it three times (see “Recovering”).

#### Failing *reason code*

The endpoint engine tries to run the resource model once every three minutes, indefinitely.

#### Missing prerequisites

One or more prerequisites of the resource model is not being met. For example, if you distribute the Print resource model to a Windows NT system, this status is generated because the resource model works only with Windows 2000.

#### Not compiled

The resource model has not been compiled.

#### Recovering *reason code*

The endpoint engine tries to run the resource model once every cycle, for three cycles only, after which the status is changed to *failed after recovery*.

#### Retrying *reason code*

The endpoint engine tries to run the resource model three times in each cycle, indefinitely.

#### Running

The resource model is running.

**Scheduled**

The resource model has been scheduled to be started.

**Stopped**

The resource model has been stopped. It is compiled and enabled and is not in an error state but it is not running and is not scheduled.

**Unable to start** *reason code*

A resource model has been unable to start.

Several examples of the output are given

1. The first shows the output received when sending a request at endpoint level (with just the endpoint specified) and without the verbose option:

```
Forwarding the request to the engine...
The following profiles are running:
myprofile#ucaselli-region
  TMW_Processor           :Running
[ucaselli][y:/w32-ix86/cdrom]
```

2. The second output example shows the same endpoint with the verbose setting:

```
Forwarding the request to the engine...
The following profiles are running:
myprofile#ucaselli-region
  TMW_Processor           :Running
  TMW_ProcessorBusy       100 %
  TMW_HighProcesses       100 %
  TMW_CPUCantKeepUpWithHW 100 %
  TMW_HWKeepingCPUBusy    100 %
  TMW_BusyHardware        100 %
  TMW_HighPercentUsageDelta 100 %
[ucaselli][y:/w32-ix86/cdrom]
```

3. The third output example shows the same endpoint with the verbose setting in XML format:

```
Forwarding the request to the engine...
The following profiles are running:
<Profiles>
  <Profile Name="myprofile#ucaselli-region">
    <RMs>
      <RM Name="TMW_Processor" Status="Running">
        <Indications>
          <Indication Name="TMW_ProcessorBusy" Health="100">
            </Indication>
          <Indication Name="TMW_HighProcesses" Health="100">
            </Indication>
          <Indication Name="TMW_CPUCantKeepUpWithHW" Health="100">
            </Indication>
          <Indication Name="TMW_HWKeepingCPUBusy" Health="100">
            </Indication>
          <Indication Name="TMW_BusyHardware" Health="100">
            </Indication>
          <Indication Name="TMW_HighPercentUsageDelta" Health="100">
            </Indication>
        </Indications>
      </RM>
    </RMs>
  </Profile>
</Profiles>
```

**Options****-e** *endpoint*

Specifies the endpoint on which the command is to be performed.



**-i** *indication*

Specifies the indication on which the command is to be performed. If the indication is not specified the command is performed on all indications in the specified resource model.

**-p** *profile#region*

Defines the profile on which the command is to be performed. If the profile is not specified, the command will be performed on all profiles at the endpoint. The profile name must always be fully qualified with the policy region name in the syntax shown.

**Note:** To obtain a listing of the complete names of all the profiles that are on an endpoint, issue the following command:

```
wdmlseng -e endpoint
```

**-r** *resource\_model*

Specifies the resource model on which the command is to be performed. If the resource model is not specified the command is performed on all resource models in the specified profile.

**-verbose**

Displays all indications for each resource model. The worst instance key (if it exists) and the status (consolidation level) of each indication are also displayed. The consolidation level indicates the completion status of the generated occurrences that are required to trigger the particular event.

If a profile is specified, all indications are displayed for each resource model of that profile.

If a resource model is also specified, all instances are displayed for each indication of that resource model.

If an indication is also specified, all categories (the indication's string property value) and metrics (the indication's numeric property value) data are displayed for each instance.

**-xml**

Displays the output in XML format.

## Authorization

Requires the super, senior, admin or user roles.

## Examples

1. This example shows the command necessary to list all the resource models at the endpoint **MyEndpoint** with full details of all indications in XML format:
 

```
wdmlseng -e MyEndpoint -verbose -xml
```
2. This example shows the command necessary to list all the resource models in the profile **MyProfile** at the endpoint **MyEndpoint** with full details of all indications in XML format:
 

```
wdmlseng -e MyEndpoint -p MyProfile -verbose -xml
```
3. This example shows the command necessary to list the resource model **MyResourceModel** in the profile **MyProfile** at the endpoint **MyEndpoint** with full details of all indications in XML format:
 

```
wdmlseng -e MyEndpoint -p MyProfile -r MyResourceModel -verbose -xml
```

4. This example shows the command necessary to list the indication **MyIndication** in the resource model **MyResourceModel** in the profile **MyProfile** at the endpoint **MyEndpoint** with full details of all indications in XML format:

```
wdmIseng -e MyEndpoint -p MyProfile -r MyResourceModel -i MyIndication  
-verbose -xml
```

## See Also

**wdmcmd**

**wdmeng**

**wdmtrceng**

## wdmmn

Stops or starts selected Tivoli Monitoring processes on one or all managed nodes/gateways.

### Syntax

```
wdmmn -start [-m {managed_node | all}] [-t] [-r]
```

```
wdmmn -stop [-m {managed_node | all}] [-b] [-h] [-t] [-r]
```

### Description

This command starts or stops the various Tivoli Monitoring processes running on managed nodes/gateways. The processes that can be started and stopped are:

- Task engine
- Tivoli Business Systems Manager Adapter
- Heartbeat engine
- The request manager processor

You can either specify individual managed nodes/gateways or all managed nodes/gateways on which Tivoli Monitoring is installed.

### Options

**-b** Stops the Tivoli Business Systems Manager Adapter.

**-h** Stops the heartbeat engine.

**-m** {*managed\_node* | **all**}

Identifies the managed node/gateway on which the command is to be run. If it is not supplied, the command will be performed on the managed node/gateway from which it is issued. If it is supplied you must identify a specific managed node/gateway, or all managed nodes/gateways using the **all** argument.

**-start**

Starts the defined process or processes. If no **-r** or **-t** argument is supplied, the task engine is started (**-t**).

**-stop**

Stops the defined process or processes. If no **-b**, **-h**, **-r**, or **-t** argument is supplied, the task engine is stopped (**-t**).

**-t** Starts or stops the task engine. This is the default value.

**-r** Starts or stops the request manager.

### Authorization

Requires the super, senior or admin roles.

### Examples

1. The following example stops the Tivoli Monitoring heartbeat engine on the managed node/gateway from which it is issued.

```
wdmmn -stop -h
```

2. The following example starts the Tivoli Monitoring task engine on the managed node/gateway from which it is issued.

```
wdmmn -start
```

3. The following example starts the task engine on managed node/gateway **MyManagedNode**.  
`wdmmn -start -m MyManagedNode -t`
4. The following example stops the Tivoli Business Systems Manager Adapter on all the managed nodes/gateways.  
`wdmmn -stop -m all -b`
5. The following example stops the heartbeat engine on all the managed nodes/gateways.  
`wdmmn -stop -m all -h`
6. The following example stops the Tivoli Monitoring task engine and the Tivoli Business Systems Manager Adapter on all the managed nodes/gateways.  
`wdmmn -stop -m all -t -b`
7. The following example stops the heartbeat engine and the Tivoli Business Systems Manager Adapter on all the managed nodes/gateways.  
`wdmmn -stop -m all -h -b`
8. The following example stops the Tivoli Monitoring task engine and the heartbeat engine on all the managed nodes/gateways.  
`wdmmn -stop -m all -t -h`
9. The following example stops all processes on all the managed nodes/gateways.  
`wdmmn -stop -m all -t -h -b -r`

## See Also

None.

## wdmmngcache

Deletes or lists part or all of the contents of the gateway endpoint cache.

### Syntax

```
wdmmngcache [-m {managed_node | @managed_nodes_file | all}] -I [-v]
```

```
wdmmngcache [-m {managed_node | @managed_nodes_file | all}]
  -d {all | discovered | dead | endpoint | @endpoints_file}
```

### Description

The **wdmmngcache** command operates on a selected managed node/gateway, a list of managed nodes/gateways contained in a file or all managed nodes/gateways, and either lists or deletes all or part of the contents of the cache maintained by the Tivoli Business Systems Manager Adapter.

If the delete option is used, endpoint details are marked for deletion but are not deleted until the next discovery action. Thus, you are recommended to carry out a discovery prior to using the list option of this command, and after using the delete option.

The output of the list option is available in two formats. A sample of the output without using the **-v** argument is as follows, showing details of the heartbeat status of the endpoints in the cache:

```
Processing ManagedNode mpulp...
Processing ManagedNode dmw2k2...
Endpoint                                     | Status
-----+-----
mpulp-ep                                     Alive
dmw2k2-ep                                     Alive
```

A sample of the output using the **-v** argument is as follows, showing the Tivoli Business Systems Manager status of the endpoints in the cache:

```
Processing ManagedNode mcrudele...
Processing ManagedNode dmw2k2...
Warning: DM_Advanced_Edition feature not installed
on the Managed Node 'boccaccio'. Skipping...
Endpoint           | HB status           | TBSM status
-----+-----+-----
dmw2k2-ep          DMEngineOff         Not discovered
mcrudele-ep        DMEngineOff         Not discovered
```

### Options

**-m** {*managed\_node* | *@managed\_nodes\_file* | **all**}

Specifies the managed nodes/gateways on which the command is to be carried out. If this option is not used the command is carried out on the managed node/gateway from which it is being run.

The options are as follows:

*managed\_node*

The name of the managed node/gateway on which the command is to be performed.

*@managed\_nodes\_file*

A file containing managed node/gateway names separated by the CR/LF character; the @ symbol is an indicator that a file name follows but the symbol should not form part of the filename.

**all** All managed nodes/gateways in the Tivoli region

- l This option lists the contents of the cache at the defined managed nodes/gateways. It has one optional argument:
  - v Provides details of the Tivoli Business Systems Manager status of the endpoints in the cache. If this argument is not supplied, the command provides details of the heartbeat status of the endpoints in the cache.
- d {**all** | **discovered** | **dead** | *endpoint* | *@endpoints\_file*}  
Deletes the cache appertaining to the defined endpoints. You should note that the deleted endpoint details remain visible using the -l option until the next discovery request is made to the managed node/gateway to which the endpoint is attached. The endpoint definition can be one of the following:
  - all** All endpoints attached to the managed nodes/gateways
  - discovered**  
All endpoints revealed as active by any discovery commands
  - dead**  
All endpoints revealed as inactive by any discovery commands
  - endpoint*  
An endpoint label.
  - @endpoints\_file*  
A file containing endpoint labels separated by the CR/LF character; the @ symbol is an indicator that a file name follows but the symbol should not form part of the filename.

## Authorization

Requires the super, senior or admin roles.

## Examples

1. The following example lists the cache at the managed node/gateway **MyManagedNode**:  

```
wdmmngcache -m MyManagedNode -l
```
2. The following example lists the status of the Tivoli Business Systems Manager Adapter at all gateways/managed nodes in the Tivoli management region:  

```
wdmmngcache -m all -l -v
```
3. The following example deletes the cached details of all endpoints at all managed nodes/gateways in the Tivoli region:  

```
wdmmngcache -m all -d all
```
4. The following example deletes the cached details of endpoints revealed by a discovery at all managed nodes/gateways in the Tivoli region:  

```
wdmmngcache -m all -d discovered
```
5. The following example deletes the cached details of endpoints not revealed by a discovery at all managed nodes/gateways in the Tivoli region:  

```
wdmmngcache -m all -d dead
```
6. The following example deletes the cached details of endpoint **MyEndpoint1** at the managed node/gateway from which the command is given:  

```
wdmmngcache -d MyEndpoint1
```
7. The following example deletes the cached details of all endpoints listed in the file **MyEndpointList** at the managed node/gateway from which the command is given:  

```
wdmmngcache -d @MyEndpointList
```

**See Also****wdmconfig****wdmdiscovery****wdmheartbeat**

---

## wdmrm

Adds, lists or removes a specified default resource model at the Tivoli management region server or managed node/gateway from where it is issued. It also adds the NLS catalog to an already installed default resource model.

### Syntax

**wdmrm -add** *resource\_model\_tarfile*

**wdmrm -addcat** *resource\_model* [-f *catalog\_file* -l *locale*]

**wdmrm -list**

**wdmrm -remove** *resource\_model*

### Description

This command adds, lists or removes the specified resource model at the Tivoli server. In the case of the remove option it removes the resource model from the Tivoli Name Registry (TNR) on the server and from all profiles to which the resource model has previously been added. It does not remove the resource model from the copies of the profiles at the endpoints.

The command also adds the NLS catalog to the resource model, so that when it is subsequently distributed to endpoints, the resource model has all the necessary NLS information to support the display of localized events on the Tivoli Enterprise Console server.

The list option produces output similar to the following for each resource model:

```
Resource -> DMXCpu
  NLS name       : CPU
  product_id     : none
  major_version  : 1
  minor_version  : 0
  platform       : aix4-r1\hpux10\linux-ix86\linux-s390\solaris2
  message catalog : DMXCpu
  zip file       : DMXCpu.zip
```

### Options

#### **-add**

Adds the resource model in the specified *resource\_model\_tarfile* to the Tivoli server.

#### **-addcat**

Adds the specified NLS catalog to the specified resource model at the Tivoli server; if the resource model does not exist the command fails. If no NLS catalog is specified, the command searches the *msg\_cat* directory and updates the specified resource model with the details of any NLS catalogs that it finds, that are not already added to the resource model (*locales* that are absent are added, those that are present are upgraded).

**[-f *catalog\_file* -l *locale*]**

Defines a specific NLS catalog that is to be added to the resource model. If this option is not specified, the resource model will be updated with the details of all NLS catalogs in the *msg\_cat* directory.

#### **-list**

Lists all resource models defined at the Tivoli server.



**-remove**

Removes the specified resource model from the Tivoli server, removing it also from all profile definitions at the server that contain the resource model.

*resource\_model*

Specifies the resource model to remove, or to which you want to add NLS catalog details.

*resource\_model\_tarfile*

Specifies the name of the file that is produced by Tivoli Workbench when you build a resource model package.

**Authorization**

On a Windows system, this command can only be run from the bash Shell. It requires the super, admin or senior roles.

**Examples**

1. The following example adds a resource model created by the user, and saved in a file called **MyResModel.tar**, to the list of available resource models.  

```
wdmrm -add MyResModel.tar
```
2. The following example updates the resource model called **MyResModel**, with all the NLS catalog details in the msg\_cat directory:  

```
wdmrm -addcat MyResModel
```
3. The following example adds the locale **MyLocale** in the NLS catalog **MyCatalog** to the resource model called **MyResModel**:  

```
wdmrm -addcat MyResModel -f MyCatalog -l MyLocale
```
4. The following example returns a list of all the resource models that are defined at the Tivoli management region.  

```
wdmrm -list
```
5. The following example removes the resource model **MyResModel** from the resource model registry and all profiles in the Tivoli region (except copies of profiles at endpoints):  

```
wdmrm -remove MyResModel
```

**See Also**

None.

## wdmtrceng

Sets the trace parameters of the Tivoli Monitoring engine at the endpoint.

### Syntax

```
wdmtrceng -e endpoint logfile_name trace_level size
```

### Description

This command sets the trace parameters of the engine on the endpoint. For more details about the traces, see “Logs and Traces” on page 179.

### Options

#### *endpoint*

Specifies the endpoint where the command is to be executed.

#### *trace\_level*

The value set for the trace level of the endpoint’s engine. The *level* can be one of the following:

- 0** Only errors are stored in the log file. This is the default value. For performance reasons, Tivoli recommends that you do not change this value unless you are experiencing problems that you want to report to Customer Support.
- 1** Only warnings and errors are stored in the log file.
- 2** All steps of the monitoring process are stored in the log file.
- 3** Verbose mode: all operations performed during monitoring are stored in the log file.

#### *logfile\_name*

- For Windows endpoints, the name, with its complete path, of the logfile where the output of the trace is to be saved.
- For UNIX or Linux endpoints, this option to set the log name cannot be used; the log names are as described in “Non-Windows Endpoints Logs” on page 187.

#### *size*

The maximum size of the logfile, specified in bytes. Specify **-1** to reset the size to the default value (5 MB for Windows, 2.5MB for UNIX/Linux).

### Authorization

Requires the super, senior or admin roles.

### Examples

1. The following example sets the logfile **MyLogFile** to trace level *verbose mode* and a size of 1 MB on the endpoint **MyEP1**.

```
wdmtrceng -e MyEP1 C:\Log\MyLogFile 3 1000000
```

2. The following example resets the default trace level on the endpoint **MyEP1**.

```
wdmtrceng -e MyEP1 "" -1 -1
```

**See Also**

**wdmcmd**

**wdmeng**

**wdmlseng**

## **wtmntaddrm**

This is an obsolete command that has now been merged into the new command **wdmrm**. The command has been retained in this version of the product to provide continuity for customers with scripts that used the old command. It will not be included in any future release, so customers are advised to convert scripts to use the new command.

For details of the new command **wdmrm** see “wdmrm” on page 144.

### **See Also**

**wtmntdefrm**

**wtmntrmrm**

---

## wtmntdefrm

This is an obsolete command that has now been merged into the new command **wdmrm**. The command has been retained in this version of the product to provide continuity for customers with scripts that used the old command. It will not be included in any future release, so customers are advised to convert scripts to use the new command.

For details of the new command **wdmrm** see “wdmrm” on page 144.

### See Also

**wtmntaddrm**

**wtmnttrrm**

## **wtmntrmrm**

This is an obsolete command that has now been merged into the new command **wdmrm**. The command has been retained in this version of the product to provide continuity for customers with scripts that used the old command. It will not be included in any future release, so customers are advised to convert scripts to use the new command.

For details of the new command **wdmrm** see “wdmrm” on page 144.

### **See Also**

**wtmntaddrm**

**wtmntdefrm**

# B

## Error, Warning, and Information Messages

---

This appendix explains the messages that can be issued by the server, client and gateway when Tivoli Monitoring is running.

Messages are listed in ascending numeric order.

### Identifying a Message

Messages are of different type but are all identified in the same way. The following example shows a typical message and explains its identifying components.

Identity	Message
AMW0001E	You must specify a label for this profile.

**AMW** This prefix identifies the message as belonging to Tivoli Monitoring

**0005** The unique serial number of the message

**E** Is the type of message and may be:

- I** **Information messages** provide feedback about something that has happened in the product or system that may be important. These messages also give guidance when you are requesting a specific action from the product.
- W** **Warning messages** call your attention to an exception condition that is not necessarily an error but may cause problems if not attended to.
- E** **Error messages** indicate that an action cannot be completed because of a user or system error. These error messages always require user response.

### Notation

Some messages, especially information and warning messages, are multi-purpose. The same basic text can contain different strings such as different command names or application names, according to the way the application was behaving when the message was generated. These messages are shown in the following sections with the string identity displayed in italics at the appropriate part of the message.

---

## Messages

The following messages can be displayed.

---

- AMW0001E**     **A label was not specified for the profile.**  
**Explanation:** A profile was created, without specifying a label.  
**User Response:** Specify a label for the profile.
- 
- AMW0002E**     **Could not create the profile database. An error occurred with the profile manager.**  
**Explanation:** The profile entry in the Tivoli database could not be created. This is a serious error that indicates that something is wrong with the Tivoli Object Database.  
**System Action:** Uninstall and reinstall the product.  
**User Response:** Contact a system administrator to reinstall the product.
- 
- AMW0003E**     **Cannot find the IBM Tivoli Monitoring default profile. Errors might have occurred during installation.**  
**Explanation:** During installation, a default policy region, a default profile manager, and a default profile are created. The default profile is used as a template to create all the other profiles. If it is deleted, it is not possible to create other profiles.  
**System Action:** Uninstall and reinstall the product.  
**User Response:** Contact a system administrator to reinstall the product.
- 
- AMW0004E**     **A name for the resource model was not specified. The record could not be added to the database.**  
**Explanation:** A resource model was added to a profile without specifying a name.  
**User Response:** Specify the resource model name and repeat the operation.
- 
- AMW0005E**     **An IBM Tivoli Monitoring internal application error occurred while getting the values from the Tivoli Object database.**  
**Explanation:** The object database is corrupted.  
**User Response:** Delete the profile and create a new one.
- 
- AMW0006E**     **Could not insert the resource model in the database. The resource model name was not specified.**  
**Explanation:** A resource model was added, without specifying a name.  
**User Response:** Set the field for the resource model name and repeat the operation.
- 
- AMW0007E**     **The function convert\_entry\_to\_nmval failed. An internal application error occurred while trying to insert the resource model information in the profile database.**  
**Explanation:** It is not possible to add new resource models to the profile database.  
**System Programmer Response:** If the problem persists, uninstall and reinstall the product.  
**User Response:** Close the IBM Tivoli Monitoring graphical interface, wait for 30 seconds and then reopen it. If the problem persists, contact a system administrator to uninstall and reinstall the product.
-



- 
- AMW0008E**     **The function CCMS\_Database\_add\_record failed. The resource model record is already in the profile database for the target profile.**
- Explanation:** It is not possible to have multiple instances of the same resource model in a single profile. If you receive this error while moving or copying a resource model to another profile, the target profile already contains the resource model. To overwrite the resource model in the target profile, delete the old instance in the target profile before you move or copy the resource model.
- If you receive this error while adding a resource model (with or without default values) to a profile, the profile database is corrupted.
- User Response:** If you receive this error while moving or copying records from a source profile to a target profile, you must delete the instances of the same resource models in the target profile. If you receive this error while adding a resource model to a profile, delete the profile and create a new one.
- 
- AMW0009I**     **Added an entry for *resource\_model* to the *name* profile.**
- Explanation:** You entered a new entry for *resource\_model* to *profile\_name*.
- User Response:** None.
- 
- AMW0010I**     **Finished adding entries to the *profile\_name* profile. Number of entries entered: *number\_of\_entries*.**
- Explanation:** The operation of adding entries has been completed successfully.
- User Response:** None.
- 
- AMW0011E**     **You are not authorized to retrieve a policy or validation record.**
- Explanation:** The operation could not be performed. You are not authorized to retrieve the content of a policy or of a validation record.
- User Response:** None.
- 
- AMW0012E**     **The list of keys is empty.**
- Explanation:** Internal error. The list of keys that corresponds to the records that internally identify the resource models is empty. The profile database may be corrupted.
- User Response:** Delete the corrupted profile. If the problem recurs on other newly created profiles, contact your Tivoli Customer Support.
- 
- AMW0013E**     **Cannot retrieve the record with the key *key\_number*. An IBM Tivoli Distributed internal application error occurred.**
- Explanation:** Internal error. A serious error occurred, because the record that corresponds to the *key\_number* could not be retrieved. The profile database may be corrupted.
- User Response:** Delete the corrupted profile. If the problem recurs on other newly created profiles, contact your Tivoli Customer Support.
- 
- AMW0014E**     **The key *record\_key* is already in use. An IBM Tivoli Monitoring internal application error occurred.**
- Explanation:** A serious error occurred due to a corruption of the profile database.
- User Response:** Delete the corrupted profile. If the problem occurs in other profiles, contact your Tivoli Customer Support.
-

- AMW0015I**      **Removed the following entries from the *profile\_name* profile:**  
**Explanation:** The specified entries were removed from *profile\_name*.  
**User Response:** None.
- 
- AMW0016E**      **Cannot retrieve default record with the key *key\_number*. An IBM Tivoli Monitoring internal application error occurred.**  
**Explanation:** Internal Error. A serious error occurred. It was not possible to retrieve the default record corresponding to the key. The profile database may be corrupted.  
**User Response:** Delete the corrupted profile. If the problem recurs on newly created profiles, contact your Tivoli Customer Support.
- 
- AMW0017E**      **Cannot retrieve validation record with key: *key\_number*. A Distributed Monitoring internal application error occurred.**  
**Explanation:** Internal Error. A serious error occurred. It was not possible to retrieve the validation record corresponding to the key. The profile database may be corrupted.  
**User Response:** Delete the corrupted profile. If the problem recurs on newly created profiles, contact your Tivoli Customer Support.
- 
- AMW0018I**      **The following records could not be moved or copied: *record1* {*record2 record3*}.**  
**Explanation:** The records listed could not be moved or copied.  
**User Response:** Repeat the operation. If the problem recurs, contact your Tivoli Customer Support.
- 
- AMW0019I**      **Removed the *profile\_name* profile.**  
**Explanation:** The specified profile was removed.  
**User Response:** None.
- 
- AMW0020E**      **First select the entry you want to remove.**  
**Explanation:** You tried to remove an entry without selecting it.  
**User Response:** Select the entry you want to remove first. Then repeat the remove operation.
- 
- AMW0021E**      **First select the entry you want to lock or unlock.**  
**Explanation:** You tried to lock or unlock an entry without selecting it.  
**User Response:** Select the entry you want to lock or unlock first. Then repeat the remove operation.
- 
- AMW0022E**      **You cannot currently edit this entry. This entry is currently being edited by another user.**  
**Explanation:** Internal error. You tried to edit an entry that is currently being edited by another user.  
**User Response:** None.
-

- 
- AMW0025E**     **An IBM Tivoli Monitoring internal application error occurred. Contact Tivoli Customer Support for assistance.**
- Explanation:** Something failed during the creation of the default profile. This error can happen during product installation.
- System Programmer Response:** This is a serious error. Contact your Tivoli Customer Support for assistance.
- 
- AMW0026E**     **An application error occurred while distributing the profile. Contact your systems administrator for assistance.**
- Explanation:** This error is generated when a distribution fails for reasons other than a CCMS distribution failure, such as an error with an endpoint or with the oserv.
- System Programmer Response:** Verify that the subscribers to the distribution are defined correctly, in particular that the endpoints are running. If this does not resolve the problem, contact your systems administrator for assistance.
- 
- AMW0027E**     **There are no records in the specified profile. The profile database is empty.**
- Explanation:** When a profile is created, the default record, the validation record, and a record that holds general information are created. This error indicates that either the database is corrupted or that there was a failure during profile creation.
- User Response:** Create a new profile, open it and verify if the problem persists. If the new profile is created correctly, this indicates that the profile database associated with the first profile is corrupted. If the problem persists, contact the Tivoli Customer Support.
- 
- AMW0028E**     **Could not retrieve the records associated with the profile, because you do not have read access rights to the profile database.**
- Explanation:** The user who attempted to open the profile does not have the permissions to access the profile database.
- User Response:** Contact a system administrator to verify that the user has the correct access rights. If the user has the correct access rights and still cannot access the database, contact the Tivoli Customer Support because a critical error condition has occurred.
- 
- AMW0030I**     **Added the general record for the *profile\_name* profile.**
- Explanation:** The general record was added to the specified profile.
- User Response:** None.
- 
- AMW0031I**     **The record was deleted from the *profile\_name* profile.**
- Explanation:** The record was deleted from the specified profile.
- User Response:** None.
- 
- AMW0032E**     **Insert a numeric value.**
- Explanation:** The operator attempted to insert an alphanumeric value in a text field that accepts only numeric values
- User Response:** Insert a numeric value and repeat the operation.
-

**AMW0033E**     **You must enter a location for the Tivoli Enterprise Console server.**

**Explanation:** You specified in the properties dialog to send Tivoli Enterprise Console server events but did not specify the location of the Tivoli Enterprise Console server.

**User Response:** Open the IBM Tivoli Monitoring profile, select the Edit → Properties menu and enter the location of the Tivoli Enterprise Console server.

---

**AMW0034E**     **The IBM Tivoli Monitoring profile is currently being used by another user.**

**Explanation:** The IBM Tivoli Monitoring profile is currently being used by another user.

**User Response:** None.

---

| **AMW0037E**     **A fatal error occurred during parsing.**

| **Explanation:** The command failed while parsing the resource model.

| **User Response:** Use the workbench to create the resource model again, making sure that all the internal names of the resource model are alphanumeric and do not include blanks.

---

**AMW0038E**     **Transaction error.**

**Explanation:** A transaction error occurred. The command wdmrm cannot be completed because it is not possible to start a new transaction.

**User Response:** None.

---

**AMW0039I**     **Zip and message catalog files cannot be removed because they are being used by another resource.**

**Explanation:** Another resource is currently using the files you are trying to remove, therefore they cannot be removed.

**User Response:** Determine which resources are locking the files. First release them, then try again.

---

**AMW0040E**     **Resource *resource\_name* does not exist.**

**Explanation:** The resource you specified does not exist.

**User Response:** Enter a correct resource name.

---

**AMW0041E**     **You tried to add an older version of the resource *resource\_name*.**

**Explanation:** You tried to add a version of the specified resource that is older than the current one.

**User Response:** Check the version number of the resource you are trying to add. Perform the add operation with the newer version.

---

**AMW0044E**     **An internal error occurred in the parameters dialog.**

**Explanation:** An error occurred due to incorrect parameters.

**User Response:** Enter the correct parameters.

---

---

AMW0045E	<p><b>You must enter an element to add to the list.</b></p> <p><b>Explanation:</b> The add operation failed because you did not specify the element to be added to the list.</p> <p><b>User Response:</b> Specify the element to be added to the list.</p>
AMW0046E	<p><b>Unable to perform the operation. An error occurred.</b></p> <p><b>Explanation:</b> The command wdmrm cannot complete successfully because it is unable to determine the installation directory for the product. Check that the Tivoli environment has been set correctly.</p> <p><b>User Response:</b> None.</p>
AMW0047I	<p><b>The resource model already has this status.</b></p> <p><b>Explanation:</b> The resource model already has the status to which you tried to change it.</p> <p><b>User Response:</b> None.</p>
AMW0048E	<p><b>The resource model has not been distributed.</b></p> <p><b>Explanation:</b> You tried to perform an operation on a resource model that has not been distributed.</p> <p><b>User Response:</b> First distribute a profile containing the resource model, and then repeat the operation.</p>
AMW0049E	<p><b>The profile has not been distributed.</b></p> <p><b>Explanation:</b> You tried to perform an operation on a profile that has not been distributed.</p> <p><b>User Response:</b> First distribute the profile and then repeat the operation.</p>
AMW0050E	<p><b>The profile cannot be deleted. It has not been distributed.</b></p> <p><b>Explanation:</b> You tried to remove a profile from an endpoint to which it has not been distributed.</p> <p><b>User Response:</b> None.</p>
AMW0051E	<p><b>The engine is not running or is unreachable.</b></p> <p><b>Explanation:</b> The operation you are trying to perform on the endpoint cannot be executed, either because the engine is not running on the endpoint, or because the endpoint is not reachable.</p> <p><b>User Response:</b> Check that the Tivoli Distributed Monitoring (Advanced Edition) endpoint engine and the Tivoli Management Agent at the endpoint are both running, and retry the operation.</p>
AMW0052E	<p><b>The schedule for this entry is not correct. Ensure that the stop date is later than the start date and at least one scheduling rule exists.</b></p> <p><b>Explanation:</b> The values entered to schedule this rule are not correct.</p> <p><b>User Response:</b> Ensure that the stop date is later than the start date and that at least one scheduling rule has been defined.</p>
AMW0053E	<p><b>The specified schedule rules do not overlap with the specified date range. At least one day from one rule must fall within the scheduled start and stop dates.</b></p> <p><b>Explanation:</b> The specified schedule rules do not overlap with the specified date range.</p> <p><b>User Response:</b> Ensure that at least one day in one of the rules falls within the specified date range.</p>

---

**AMW0054E**      **An internal error occurred within IBM Tivoli Monitoring. Contact your Tivoli Customer Support for assistance.**

**Explanation:** The scheduling rule is incorrect. Check the consistency of the parameters.

**User Response:** Contact your Tivoli customer Support for assistance.

---

**AMW0055E**      **A rule name was not specified.**

**Explanation:** You did not specify any rule name.

**User Response:** Specify a rule name.

---

**AMW0056E**      **The rule name contains invalid characters.**

**Explanation:** You entered an invalid character in the rule name.

**User Response:** Enter the rule name again, using only valid characters. Do not use the equals (=) character.

---

**AMW0057E**      **A schedule cycle was not specified.**

**Explanation:** You did not specify any schedule cycle.

**User Response:** Specify a schedule cycle.

---

**AMW0058E**      **You must select an element to delete from the list.**

**Explanation:** The delete operation failed because you did not specify the element to be deleted from the list.

**User Response:** Specify the element to be deleted from the list and retry the operation.

---

**AMW0059E**      **The schedule rule has an invalid stop time. Ensure that the stop time is later than the start time.**

**Explanation:** You entered an invalid stop time.

**User Response:** Ensure that the stop time is later than the start time.

---

**AMW0060E**      **An internal error occurred in the Parameters dialog.**

**Explanation:** An internal error occurred in the Parameters dialog.

**User Response:** None.

---

**AMW0061E**      **The resource model *resource\_model1* cannot be installed, because it redefines the event *event* which is already used by the resource model *resource\_model2*.**

**Explanation:** As message.

**User Response:** None.

---

**AMW0062E**      **The historical period must be greater than zero.**

**Explanation:** The historical period determines the period of time that data is maintained in the logger database. For the logger to record any data, the historical period must have a value. Check the name of the profile and attempt the operation again.

**User Response:** Select a value greater than zero in the Historical Period drop-down list.

---

- 
- AMW0063E**     **The historical period must be greater than the aggregation period, and the aggregation period must be greater than zero.**
- Explanation:** During aggregation periods, data are collected, and minimum, average, and maximum values obtained. A historical period must be long enough to hold data from one or more aggregation periods. If a historical period is shorter than an aggregation period, it cannot obtain the data to average.
- User Response:** Select a new value for the historical period or for the aggregation period, so that the requirements outlined in the explanation are met.
- 
- AMW0064E**     **The Tmw2Profile *profile\_name* does not exist.**
- Explanation:** The system could not find the profile that you tried to edit.
- User Response:** Check the name of the profile and attempt the operation again.
- 
- AMW0065E**     **The resource model *resource\_model* does not exist.**
- Explanation:** The system could not find the resource model that you tried to add to the profile.
- User Response:** Check the name of the resource model and attempt the operation again. You can display a list of resource models by using the following command:
- ```
wdmrm -list
```
- 
- AMW0066E**     **The following resource model does not belong to the following profile. Resource Model: *resource\_model*  
Profile: *profile\_name***
- Explanation:** You tried to delete or edit a resource model. However, the system could not find the resource model in the specified profile.
- User Response:** Check the name of the resource model. You can list all the resource models in a profile by using the following command:
- ```
wdmeditprf -P profile_name -list
```
- Alternatively, add the resource model to the profile with the following command:
- ```
wdmeditprf -P profile_name -add resource_model
```
- 
- AMW0067E**     **The cycle time is missing.**
- Explanation:** The cycle time was not included in the following command:
- ```
wdmeditprf -P profile_name -add resource_model -c cycle_time
```
- User Response:** Enter the command again, making sure you include a value for the cycle time.
- 
- AMW0068E**     **The cycle time must be an integer, expressed numerically.**
- Explanation:** The cycle time must be a positive integer. Typically, this has a value in the range 60 - 300.
- User Response:** Enter the command again, making sure you include a value in the correct format.
- 
- AMW0069E**     **The logging period is missing.**
- Explanation:** The logging period was not included in the following command:
- ```
wdmeditprf -P profile_name -add resource_model -Log -p logging_period
```
- User Response:** Enter the command again, making sure you include values for the logging period in the format hh:mm.
-

**AMW0070E**     **The logging period must be in the format hh:mm.**

**Explanation:** The logging period was incorrectly specified in the following command:

```
wdmeditprf -P profile_name -add resource_model -Log -p logging_period
```

The logging period must be specified in hours and minutes, in the format hh:mm. For example, 22:15.

**User Response:** Enter the command again, using the correct syntax.

---

**AMW0071E**     **The aggregation setting was not specified.**

**Explanation:** The aggregation option was included in the following command, but the corresponding value was not included:

```
wdmeditprf -P profile_name -add resource_model -Log -Agg
```

**User Response:** Enter the command again, including the aggregation option and the value yes or the value no.

---

**AMW0072E**     **The logging aggregation period is missing.**

**Explanation:** The aggregation period option was included in the following command, but the corresponding value was not included:

```
wdmeditprf -P profile_name -add resource_model -Log -ap aggregation_period
```

**User Response:** Enter the command again, including the aggregation period option and the corresponding value. This is expressed in hours and minutes, in the format hh:mm. For example, 22:15.

---

**AMW0073E**     **The logging aggregation period must be in the format hh:mm.**

**Explanation:** The aggregation period option was included in one of the following commands, but the corresponding value was specified in an incorrect format:

```
wdmeditprf -P profile_name -add resource_model -Log -ap aggregation_period
```

```
wdmeditprf -P profile_name -edit resource_model -Log -ap aggregation_period
```

**User Response:** Enter the command again, including the aggregation period option and the corresponding value in the correct format. This is expressed in hours and minutes, in the format hh:mm. For example, 01:20.

---

**AMW0074E**     **The event name is missing.**

**Explanation:** The event option was included in one of the following commands, but the corresponding value was not included:

```
wdmeditprf -P profile_name -add resource_model -e event_name
```

```
wdmeditprf -P profile_name -edit resource_model -e event_name
```

**User Response:** Enter the command again, including the event option and a value for the event name.

---

**AMW0075E**     **The value for the number of occurrences is missing.**

**Explanation:** The occurrence option was included in one of the following commands, but the corresponding value was not included:

```
wdmeditprf -P profile_name -add resource_model -e event_name -o occurrence_value
```

```
wdmeditprf -P profile_name -edit resource_model -e event_name -o occurrence_value
```

**User Response:** Enter the command again, including the occurrence option and a suitable numeric value. Typical values are 1 - 10.

---



- 
- AMW0076E**     **The value for the number of holes is missing.**
- Explanation:** A series of potential event triggers causes an event. These triggers are called occurrences. The number of occurrences required to cause the event is specified by the occurrence value. However, gaps (or holes) in the series of occurrences can also be specified. For example, if a value of 1 is specified for the number of holes, and a value of 3 is specified for the occurrence value, then the following does not cause an event:
- ```
potential_trigger_1, potential_trigger_2, hole_1, hole_2, potential_trigger_3
```
- However, the following does cause an event:
- ```
potential_trigger_1, potential_trigger_2, hole_1, potential_trigger_3
```
- The occurrence option was included in one of the following commands, but the corresponding value was not included:
- ```
wdmeditprf -P profile_name -add resource_model -e event_name -o occur_value -h holes_value
wdmeditprf -P profile_name -edit resource_model -e event_name -o occur_value -h holes_value
```
- User Response:** Enter the command again, including the holes option together with a suitable numeric value. Typical values are 0 - 5.
- 
- AMW0077E**     **Event *event\_name* does not exist.**
- Explanation:** You entered a command specifying an event that the system cannot find in the resource model.
- User Response:** Enter the command again, specifying an existing event. You can list all events available for a resource model with the following command:
- ```
wdmeditprf -P profile_name -print resource_model -e
```
- 
- AMW0078E**     **The threshold name is missing.**
- Explanation:** The threshold option was included in one of the following commands, but the corresponding name was not included:
- ```
wdmeditprf -P profile_name -add resource_model -t threshold_name threshold_value
wdmeditprf -P profile_name -edit resource_model -t threshold_name threshold_value
```
- User Response:** Enter the command again, specifying a threshold name. You can list all threshold names available for a resource model by using the following command:
- ```
wdmeditprf -P profile_name -print resource_model -t
```
- 
- AMW0079E**     **The threshold value is missing.**
- Explanation:** The threshold option was included in one of the following commands, but the corresponding value was not included:
- ```
wdmeditprf -P profile_name -add resource_model -t threshold_name threshold_value
wdmeditprf -P profile_name -edit resource_model -t threshold_name threshold_value
```
- User Response:** Enter the command again, specifying a threshold value.
- 
- AMW0080E**     **Threshold *threshold\_name* does not exist.**
- Explanation:** You entered a command specifying a threshold name that the system cannot find in the resource model.
- User Response:** Enter the command again, specifying an existing threshold name. You can list all threshold names available for a resource model by using the following command:
- ```
wdmeditprf -P profile_name -print resource_model -t
```
-

**AMW0081E**     **The Tivoli Enterprise Console server *server\_name* is not available.**

**Explanation:** You entered a command specifying a Tivoli Enterprise Console server that the system cannot find. This could be because the system does not exist, or because there is a serious error with the database.

**User Response:** Enter the command again, specifying an existing Tivoli Enterprise Console. Check your database.

---

**AMW0082E**     **The Tivoli Enterprise Console server name is missing.**

**Explanation:** The Tivoli Enterprise Console server option was included in one of the following commands, but the corresponding name was not included:

```
wdmeditprf -P profile_name -Tec secure -S TEC_server_name
```

```
wdmeditprf -P profile_name -Tec unsecure -S TEC_server_name -p port_number
```

**User Response:** Enter the command again, specifying a name for the Tivoli Enterprise Console server.

---

**AMW0083E**     **The Tivoli Enterprise Console server port number is missing.**

**Explanation:** The Tivoli Enterprise Console server option was included in the following command, but the corresponding port number was not included:

```
wdmeditprf -P profile_name -Tec unsecure -S TEC_server_name -p port_number
```

**User Response:** Enter the command again, specifying a number for the Tivoli Enterprise Console server port. This a numeric value.

---

**AMW0084E**     **The string specifying the Tivoli Enterprise Console delivery type is incorrect. Available types are "secure", "unsecure" and "none".**

**Explanation:** The Tivoli Enterprise Console server option was included in one of the following commands, but the corresponding string for the delivery type was not entered correctly:

```
wdmeditprf -P profile_name -Tec secure -S TEC_server_name
```

```
wdmeditprf -P profile_name -Tec none
```

```
wdmeditprf -P profile_name -Tec unsecure -S TEC_server_name -p port_number
```

**User Response:** Enter the command again, specifying one of the following values:

secure

unsecure

none

---

**AMW0085E**     **The parameter *parameter\_name* is not allowed.**

**Explanation:** You entered the **wdmeditprf** command specifying a parameter or option that the system does not recognize.

**User Response:** Check the command syntax and enter the command again.

---

**AMW0086E**     **It is not possible to start this transaction.**

**Explanation:** This is a Tivoli Management Environment Framework error. The command is unable to start a new transaction.

**User Response:** Contact your Tivoli Customer Support.

---

- 
- AMW0087E**     **The following resource model already belongs to the following profile. Resource Model:** *resource\_model* **Profile:** *profile\_name*
- Explanation:** You tried to add a resource model to a profile by using a command of the type:  
`wdmeditprf -P profile_name -add resource_model`
- However, the profile already contained the resource model.
- User Response:** If you want to add a resource model to the profile specified, you cannot. You must either specify a different resource model or a different profile. However, if you performed the command because you wanted to change a value, run the command again. For instance:  
`wdmeditprf -P profile_name -edit resource_model -t threshold_name threshold_value`
- where *threshold\_name* is the name of a threshold associated with editing a resource model, and *threshold\_value* is an integer value indicating a threshold.
- 
- AMW0088E**     **The severity identifier is missing.**
- Explanation:** The severity indicator option was included in one of the following commands, but the corresponding severity value was not included:  
`wdmeditprf -P profile_name -add resource_model -e event_name -severity severity_type`  
`wdmeditprf -P profile_name -edit resource_model -e event_name -severity severity_type`
- User Response:** Enter the command again, specifying one of strings shown above for the severity type. If you enter an empty string, you must enclose it with quotes.
- 
- AMW0089E**     **The following resource model cannot be distributed to the endpoint, because the resource model does not support the interp. Resource Model:** *resource\_model* **Endpoint:** *endpoint\_name* **Interp:** *interp\_name*
- Explanation:** Some code in a resource model is platform-specific. Resource models are logically contained in profiles. Some profiles, therefore, can be distributed only to some platforms.
- User Response:** Perform the action again, specifying a different resource model or endpoint.
- 
- AMW0091E**     **An error occurred during Xerces-c initialization. Exception Message:** *exception\_message*
- Explanation:** An external error occurred in the Xerces linked library while running the `wdmloadprf` command. The exception message can sometimes help to resolve this problem.
- User Response:** The user response is determined by the message.
- 
- AMW0092E**     **An error occurred during parsing. Exception Message:** *exception\_message*
- Explanation:** An external error occurred in the Xerces linked library while running the `wdmloadprf` command. The exception message can sometimes help to resolve this problem.
- User Response:** The user response is determined by the message.
- 
- AMW0093E**     **An error occurred during parsing.**
- Explanation:** An external error occurred in the Xerces linked library while running the `wdmloadprf` command.
- User Response:** Check the XML file for indications as to what caused this error.
-

**AMW0094I**     **A warning message was generated. File:** *file\_name* **Line in File:** *file\_line* **Column in File:** *file\_column*  
**Warning Message:** *message*

**Explanation:** A warning message was generated due to an error found at the indicated place in the indicated file. A warning message is less serious than an error message, and the system continues to function.

**User Response:** Check the XML file for indications as to what caused this warning. The warning message can offer additional information.

---

**AMW0095E**     **An error message has been generated. File:** *file\_name* **Line in File:** *file\_line* **Column in File:** *file\_column*  
**Warning Message:** *message*

**Explanation:** An error message was generated due to an error found at the indicated place in the indicated file. An error message is more serious than a warning message. The system cannot function.

**User Response:** Check the XML file for indications as to what caused this error. The error message can offer additional information.

---

**AMW0096E**     **A fatal message has been generated. File:** *file\_name* **Line in File:** *file\_line* **Column in File:** *file\_column*  
**Warning Message:** *message*

**Explanation:** A fatal error message was generated due to a fatal error found at the indicated place in the indicated file. A fatal error message is more serious than an error message. The system cannot function.

**User Response:** Check the XML file for indications as to what caused this error. The error message can offer additional information.

---

**AMW0097E**     **XML declaration not found.**

**Explanation:** A mistake was made in the XML syntax of the file you tried to load.

**User Response:** Check the XML file for indications as to what caused this error.

---

**AMW0098E**     **DTD not found.**

**Explanation:** A mistake was made in the XML syntax of the file you tried to load.

**User Response:** Check the XML file for indications as to what caused this error.

---

**AMW0099E**     **Element not found.**

**Explanation:** A mistake was made in the XML syntax of the file you tried to load.

**User Response:** Check the XML file for indications as to what caused this error.

---

**AMW0100E**     **The DTD defined in the specified XML file is not supported.**

**Explanation:** A mistake was made in the XML syntax of the file you tried to load.

**User Response:** Check the XML file for indications as to what caused this error.

---

**AMW0101E**     **A valid number greater than zero is required for the *attribute\_name* attribute.**

**Explanation:** A mistake was made in the XML syntax of the file you tried to load.

**User Response:** Check the XML file for indications as to what caused this error.

---

---

|                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>AMW0102E</b> | <p><b>The profile version that you are importing is too old.</b></p> <p><b>Explanation:</b> XML files contain a version number indicating the version of the profile. You tried to load a file with a profile version that was out of date, and could not run with your version of the product.</p> <p><b>User Response:</b> Generate a new XML file, using the latest version of the product.</p>                                                                                                                                                                                                                   |
| <hr/>           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>AMW0103E</b> | <p><b>The profile version that you are importing is too new.</b></p> <p><b>Explanation:</b> XML files contain a version number indicating the version of the profile. You tried to load a file with a profile version that was more recent than your version of the product.</p> <p><b>User Response:</b> Update your software. Alternatively, generate a replacement XML file, using the older version of the product.</p>                                                                                                                                                                                          |
| <hr/>           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>AMW0104E</b> | <p><b>The profile name is missing in both the XML file and in the command line options.</b></p> <p><b>Explanation:</b> Your XML file does not contain a string indicating the name of the profile.</p> <p><b>User Response:</b> You must either include the string in the XML file or specify it on the command line.</p>                                                                                                                                                                                                                                                                                            |
| <hr/>           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>AMW0105E</b> | <p><b>Only the value 'TRUE' or 'FALSE' is allowed for the <i>attribute_name</i> attribute.</b></p> <p><b>Explanation:</b> A value other than TRUE or FALSE was specified for the attribute indicated. The element is not indicated in the message.</p> <p><b>User Response:</b> Check the XML file for indications as to what caused this error.</p>                                                                                                                                                                                                                                                                 |
| <hr/>           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>AMW0106E</b> | <p><b>Only the value 'Unsecure', 'Secure' or 'None' is allowed for the <i>attribute_name</i> attribute.</b></p> <p><b>Explanation:</b> A value other than Unsecure, Secure, or None was specified for the attribute indicated. The element is not indicated in the message.</p> <p><b>User Response:</b> Check the XML file for indications as to what caused this error.</p>                                                                                                                                                                                                                                        |
| <hr/>           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>AMW0107W</b> | <p><b>The Tivoli Enterprise Console server <i>TEC_server_name</i> is not available in this region. Events to the Tivoli Enterprise Console are disabled.</b></p> <p><b>Explanation:</b> You included syntax in your XML file specifying a Tivoli Enterprise Console server that the system cannot find. This could be because the system does not exist, or because there is a serious error with the database.</p> <p><b>User Response:</b> Edit your XML file, specifying an existing Tivoli Enterprise Console. Check your database.</p>                                                                          |
| <hr/>           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>AMW0108W</b> | <p><b>A Tivoli Enterprise Console server for the non-Tivoli connection was not specified. Events to the Tivoli Enterprise Console are disabled.</b></p> <p><b>Explanation:</b> The Tivoli Enterprise Console server name is missing in the XML file.</p> <p><b>User Response:</b> Edit your XML file, specifying a value for the Tivoli Enterprise Console server.</p>                                                                                                                                                                                                                                               |
| <hr/>           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>AMW0109E</b> | <p><b>Resource model <i>resource_model</i> is not installed in this Tivoli management region.</b></p> <p><b>Explanation:</b> The XML file tried to perform an operation on a resource model that the system could not find in the Tivoli management region specified.</p> <p><b>User Response:</b> Edit your XML file, specifying a different value for the resource model. You can list all available resource models by using the following command:</p> <pre>wdmrm -list</pre> <p>Alternatively, add a new resource model by using the following command:</p> <pre>wdmrm -add <i>resource_model_tarfile</i></pre> |

---

**AMW0110E**     **The zip file associated with the resource model *resource\_model* in the XML file is not the one defined in the resource model.**

**Explanation:** The resource model associates a zip file with itself. The syntax of the XML file attempted to associate a different zip file with the resource model.

**User Response:** Edit your XML file appropriately. You can list the zip file for each resource model by using the following command:

```
wdmrm -list
```

---

**AMW0111E**     **The list of supported platforms for resource model *resource\_model* is not correct.**

**Explanation:** A resource model can run only on chosen platforms. This is defined by a list in each resource model. The list of platforms in the XML file was different to the list in the resource model.

**User Response:** Edit your XML file appropriately. You can list the platforms for each resource model by using the following command:

```
wdmrm -list
```

---

**AMW0112E**     **The version of the resource model *resource\_model* included in the XML file you are importing is not the same as that of the resource model currently installed.**

**Explanation:** The version of the resource model included in the syntax of the XML file must match that of the resource model you have installed. It does not.

**User Response:** Check the XML file to find out whether the resource model version is earlier or later than that of the one you have installed. Update your resource model, or install an older version, as appropriate. Alternatively, do not import this XML file.

---

**AMW0113E**     **Resource model *resource\_model* does not have logging capabilities.**

**Explanation:** This message can refer to the `wdmeditprf` or the `wdmloadprf` command. You tried to set logging capabilities but the resource model you specified is not able to perform this function.

**User Response:** If you used the `wdmeditprf` (edit profile) command, try the command again without the `-Log` option. If you used the `wdmloadprf` (load profile) command, edit and reload the XML file.

---

**AMW0114E**     **The aggregation period cannot be greater than the logging period, and both values must be lower than 1440.**

**Explanation:** Same as message.

**User Response:** Edit your XML file appropriately.

---

**AMW0115E**     **Threshold *threshold\_name* does not belong to resource model *resource\_model*.**

**Explanation:** Your XML file specifies a threshold name that the system cannot find in the resource model.

**User Response:** Edit your XML file appropriately. You can list all threshold names available for a resource model by using the following command:

```
wdmeditprf -P profile_name -print resource_model -t
```

---

- 
- AMW0116E**      **Threshold *threshold\_name* does not belong to resource model *resource\_model*.**
- Explanation:** All thresholds must be specified for all resource models specified in the XML file. This was not done.
- User Response:** Edit your XML file appropriately. You can list all threshold names available for a resource model by using the following command:
- ```
wdmeditprf -P profile_name -print resource_model -t
```
- 
- AMW0117E**      **Parameter *parameter\_name* does not belong to resource model *resource\_model*.**
- Explanation:** This message can refer to:
- The **wdmeditprf** command
  - The **wdmloadprf** command (this reads the XML file)
  - XML syntax used in the XML file to specify a parameter value to a resource model
- User Response:** Enter the command again. Alternatively, edit your XML file appropriately. You can list all parameter names available for a resource model by using the following command:
- ```
wdmeditprf -P profile_name -print resource_model -par
```
- 
- AMW0118E**      **Parameter *parameter\_name* of resource model *resource\_model* was not found in the XML file.**
- Explanation:** All parameters must be specified for all resource models specified in the XML file. This was not done.
- User Response:** Edit your XML file appropriately. You can list all parameter names available for a resource model by using the following command:
- ```
wdmeditprf -P profile_name -print resource_model -par
```
- 
- AMW0119E**      **The type of parameter *parameter\_name*, of resource model *resource\_model*, is incorrect.**
- Explanation:** The resource model associates each parameter name with a parameter type. The parameter type and parameter name included in the XML file must correspond to this definition.
- User Response:** Edit your XML file appropriately. The resource model documentation lists all parameter names and their types.
- 
- AMW0120E**      **One value must be specified for the choice list parameter *parameter\_name*, of the resource model *resource\_model*.**
- Explanation:** The XML file specified either no value, or several values for the indicated choice list parameter.
- User Response:** Edit your XML file appropriately, so that the choice list parameter has one value. You can list all parameter names available for a resource model by using the following command:
- ```
wdmeditprf -P profile_name -print resource_model -par
```
- 
- AMW0121E**      **The value *value* is not allowed for the parameter *parameter\_name* of the resource model *resource\_model*.**
- Explanation:** This message can be caused by running the **wdmloadprf** or **wdmeditprf** command. A parameter value was specified that is not permitted for the parameter type of the indicated parameter.
- User Response:** If the message was caused by running the **wdmloadprf** command, edit your XML file appropriately, and run the command again. If the message was caused by running the **wdmeditprf** command, enter the command again, using the correct syntax. The documentation for the resource model contains listings of all available values.
-

- AMW0122E**     **The event *event\_name* does not belong to resource model *resource\_model*.**
- Explanation:** The XML contains syntax specifying an event that the system cannot find in the resource model.
- User Response:** Edit the XML file appropriately, specifying an existing event. You can list all events available for a resource model by using the following command:
- ```
wdmeditprf -P profile_name -print resource_model -e
```
- 
- AMW0123E**     **Event *event\_name* of resource model *resource\_model* was not found in the XML file.**
- Explanation:** All events must be specified for all resource models specified in the XML file. This was not done.
- User Response:** Edit your XML file appropriately. You can list all events available for a resource model by using the following command:
- ```
wdmeditprf -P profile_name -print resource_model -e
```
- 
- AMW0124E**     **Valid severity types for an event are: HARMLESS, MINOR, WARNING, CRITICAL, FATAL, or an empty string.**
- Explanation:** This message can be caused by the severity indicator option being included in one of the following commands, but the corresponding severity type having an incorrect value:
- ```
wdmeditprf -P profile_name -add resource_model -e event_name -severity severity_type  
wdmeditprf -P profile_name -edit resource_model -e event_name -severity severity_type
```
- Alternatively, the message can be caused by an error in the XML file, when using the **wdmloadprf** command.
- User Response:** If this message was caused by the **wdmeditprf** command, enter the command again, specifying one of strings shown above for the severity type. If you enter an empty string, you must enclose it with quotes. If this message was caused by the **wdmloadprf** command, edit the XML file appropriately.
- 
- AMW0125E**     **Built-in action *builtin\_action* is not allowed for the event *event\_name* of resource model *resource\_model*.**
- Explanation:** This message is caused by an error in the XML file, when using the **wdmloadprf** command. Only some actions are permitted for a given event.
- User Response:** Edit the XML file appropriately.
- 
- AMW0126E**     **Unable to access the Tivoli name registry in the database.**
- Explanation:** For some reason, the area of the database containing the Tivoli names cannot be accessed. This is a critical error, and indicates a database malfunction.
- User Response:** Contact your Tivoli Customer Support.
- 
- AMW0127E**     **Profile *profile\_name* already exists. Specify one of the flags -k, -s, or -m to update it.**
- Explanation:** When loading a profile with the **wdmloadprf** command, you specified a profile that already exists in the policy region. However, you can only do this if you specify one of the following flags:
- k
  - s
  - m
- User Response:** Run the command again, including one of the required flags. See the man pages for more information.
-



- 
- AMW0128W**    **Profile *profile\_name* does not exist. Options -k, -s, or -m are ignored.**
- Explanation:** When loading a profile with the **wdmloadprf** command, you specified a profile that does not exist in the policy region. The profile is loaded, but any options specified by the flags -k, -s, and -m are not valid, and cannot be performed.
- User Response:** None.
- 
- AMW0129W**    **Task *task\_name* from the task library *task\_library* will not be set in response to the event *event\_name*. The task cannot be found.**
- Explanation:** When loading a profile with the **wdmloadprf** command, syntax in the XML file specified a task that cannot be associated with the event specified, because it does not exist.
- User Response:** Edit the XML file appropriately, or create the task.
- 
- AMW0130W**    **The arguments specified for the task *task\_name* from the task library *task\_library* are not necessary and will be ignored. (The task is in response to the event *event\_name*.)**
- Explanation:** When loading a profile with the **wdmloadprf** command, syntax in the XML file specified arguments for a task that cannot accept arguments.
- User Response:** No response is required, but you can edit the XML file to avoid displaying this message.
- 
- AMW0131W**    **A wrong number of arguments was specified for the task *task\_name* from the task library *task\_library*. (This task is in response to the event *event\_name*.) The task is not imported.**
- Explanation:** When loading a profile with the **wdmloadprf** command, syntax in the XML file specified too many or too few arguments for a task. Tasks require a fixed number of arguments.
- User Response:** Edit the XML file appropriately.
- 
- AMW0132W**    **Invalid scheduling settings were specified for the resource model *resource\_model*. The resource model will run without scheduling.**
- Explanation:** When loading a profile with the **wdmloadprf** command, syntax in the XML file specified scheduling settings for a resource model.
- User Response:** Edit the XML file appropriately.
- 
- AMW0133E**    **The parameter name and value are missing.**
- Explanation:** When you use the **wdmeditprf** command with the -AddPar flag, you must include a name and value for the parameter. This was not done.
- User Response:** Enter the command correctly.
- 
- AMW0134E**    **The parameter value is missing.**
- Explanation:** When you use the **wdmeditprf** command with the -AddPar flag, you must include a name and value for the parameter. The name was included, but the value was not.
- User Response:** Enter the command correctly.
-

**AMW0135E**     **The parameter value must be specified in numerals.**

**Explanation:** When you use the **wdmeditprf** command with the **-AddPar** flag, you must include a name and value for the parameter. The name was included correctly, but the value was not correctly specified. You can use only numeric characters (and the decimal point) for the value.

**User Response:** Enter the command correctly.

---

**AMW0136E**     **Parameter *parmaeter\_name* does not exist.**

**Explanation:** When you use the **wdmeditprf** command with the **-AddPar** or **-DelPar** flag, the parameter name specified must belong to the resource model.

**User Response:** Enter the command correctly. You can list all parameter names available for a resource model by using the following command:

```
wdmeditprf -P profile_name -print resource_model -par
```

---

**AMW0137E**     **Parameter *parameter\_name* does not contain the value *value* that you tried to delete.**

**Explanation:** The system could not find the parameter value that you tried to delete.

**User Response:** Enter the command correctly. You can list all parameter names and values for a resource model by using the following command:

```
wdmeditprf -P profile_name -print resource_model -par
```

---

**AMW0138E**     **Command syntax is incorrect.**

**Explanation:** An error was made in typing a command.

**User Response:** Use the associated man page to check the syntax of the command you entered. Enter the command again.

---

**AMW0139E**     **Invalid value for parameter *parameter\_name*.**

**Explanation:** When using the command **wdmeditprf** with the **-AddPar** flag, an incorrect value for a parameter was specified.

**User Response:** Enter the command again. The resource model documentation lists valid values for all parameters.

---

**AMW0140E**     **You can select, but cannot delete a choice parameter value. The choice parameter for which you tried to delete a value was *parameter\_name*.**

**Explanation:** Choice parameters allow you to select one value from a list. You cannot delete or add values (the **wdmeditprf** command with the **-Add** flag is used to select a value).

**User Response:** Enter the command again, using the **-Add** flag to select a value.

---

**AMW0141E**     **The **-x** flag of the **wdmloadprf** command is available only on **w32-ix86** platforms.**

**Explanation:** The **-x** flag is currently supported only on **w32-ix86** platforms.

**User Response:** None.

---

- 
- AMW0142E**     **The installation failed. The system cannot open the file or directory *file\_or\_directory\_name*.**
- Explanation:** The system could not open the file or directory indicated. This could be because the file or directory does not exist, or because you do not have the necessary access permissions.
- User Response:** Make sure the file exists, and that you have the required access rights.
- 
- AMW0143E**     **The installation failed. An error occurred while extracting files from the tar file *tar\_filename*.**
- Explanation:** This message is associated with the **wdmrm** command. This command opens the resource model tar file, containing the code, message catalogs, and configuration files for the resource model. There could be an error in the tar file due to a transmission error.
- User Response:** Perform the command again on a new copy of the tar file.
- 
- AMW0144E**     **The installation failed. An error occurred while creating the tar file *tar\_filename*.**
- Explanation:** This message is associated with the **wdmrm** command. The command creates a zip file that contains a tar file (although this action is transparent to the user). An error has occurred during this process.
- User Response:** Make sure you have enough disk space. Enter the command again.
- 
- AMW0145E**     **You must configure the Tivoli environment appropriately, before you can run this command.**
- Explanation:** You tried to run a Tivoli command, without first setting up the Tivoli command line interface.
- User Response:** Run the `setup_env` script to set up the Tivoli command line interface.
- 
- AMW0146E**     **The downcall DMAE\_EngineInit failed on the endpoint *endpoint\_name* It could be unreachable.**
- Explanation:** The product could not find the endpoint. For example, maybe there was a network problem, or the endpoint could be switched off.
- User Response:** Check the name of the endpoint and its availability in the network, before retrying the distribution.
- 
- AMW0147E**     **It is not possible to download dynamic dependencies to the endpoint *endpoint\_name*.**
- Explanation:** There is something wrong with the integrity of the resource model. It could be, for example, that the resource model was corrupted in transmission to the endpoint, or the integrity of the resource model on the server is damaged in some way.
- User Response:** List and check the details of the resource model on the server, before retrying the distribution.
- 
- AMW0148E**     **The CCMS push failed for the endpoint *endpoint\_name*.**
- Explanation:** There is something preventing the resource model from starting its processing at the endpoint. For example, perhaps the prerequisites at the endpoint (operating system level, ODBC, JRE) have not been satisfied.
- User Response:** Check that the prerequisites for the endpoint are satisfied (see “Prerequisite Software” on page 20), before retrying the distribution.
- 
- AMW0149E**     **Setting the boot method for the endpoint *endpoint\_name* failed.**
- Explanation:** Something has gone wrong with the setting up of the endpoint to restart the resource model automatically after being rebooted.
- User Response:** Retry the distribution. If the problems persists and cannot be resolved, contact your Tivoli Customer Support.
-

- |  |                  |   |
|--|------------------|---|
|  | <b>AMW0170E</b>  | <i>target_name</i> is not a valid target for distribution.  |
|  |                  | <b>Explanation:</b> An error occurred during the target validation of the distribution procedure.   |
|  |                  | <b>User Response:</b> Verify that each target of the distribution is a target of a valid type for the distribution of a Tivoli Monitoring profile. All targets must be valid. |
|  |                  |   |
|  | <b>AMW0171E</b>  | Cannot parse properly <i>property_name</i> .  |
|  |                  | <b>Explanation:</b> An invalid or not supported option has been specified on the command line.  |
|  |                  | <b>User Response:</b> See Appendix A, “Commands” on page 107 for supported parameters of this command.  |
|  |                  |   |
|  | <b>AMW0172E</b>  | <i>subscriber_name</i> is not subscriber of <i>profile_name</i> .   |
|  |                  | <b>Explanation:</b> An error occurred during the distribution subscribers validation, if the lenient option is not specified.   |
|  |                  | <b>User Response:</b> Verify that each target of the profile distribution is a valid subscriber for the Tivoli Monitoring profile. All targets must be valid.                 |
|  |                  |   |
|  | <b>AMW0173E</b>  | The profile <i>profile_name</i> does not have subscribers.  |
|  |                  | <b>Explanation:</b> A profile has been distributed but no valid target of the distribution has been specified.  |
|  |                  | <b>User Response:</b> Specify at least one valid target..   |
|  |                  |   |
|  | <b>AMW0174W</b>  | IBM Tivoli Monitoring not installed on the Managed Node <i>managed_node_name</i> .  |
|  |                  | <b>Explanation:</b> The product is not installed on the specified managed node.   |
|  |                  | <b>User Response:</b> None.   |
|  |                  |   |
|  | <b>AMW2007W</b>  | An internal error has occurred while performing <i>operation_name</i> .   |
|  |                  | <b>Explanation:</b> An internal error has occurred while performing an operation.   |
|  |                  | <b>User Response:</b> Try to perform the operation again.   |
|  |                  |   |
|  | <b>AMW2008EI</b> | The profile <i>profile_name</i> failed to start.  |
|  |                  | <b>Explanation:</b> The profile failed to start because of an internal error.   |
|  |                  | <b>User Response:</b> Distribute the profile again.   |
|  |                  |   |
|  | <b>AMW2009W</b>  | The profile <i>profile_name</i> is already active.  |
|  |                  | <b>Explanation:</b> You are trying to start a profile already started.  |
|  |                  | <b>User Response:</b> None.   |
|  |                  |   |
|  | <b>AMW2011W</b>  | The profile <i>profile_name</i> is not active.  |
|  |                  | <b>Explanation:</b> You are trying to stop a profile that has not been started.   |
|  |                  | <b>User Response:</b> None.   |
-

---

AMW2012W	<b>The resource model <i>resource_model</i> is already active in the profile <i>profile_name</i>.</b> <b>Explanation:</b> The resource model is already active in the profile. <b>User Response:</b> None.
AMW2015I	<b>The resource model <i>resource_model</i> of the profile <i>profile_name</i> has failed to start.</b> <b>Explanation:</b> The resource model failed to start because of an internal error. <b>User Response:</b> None.
AMW2016E	<b>The resource model <i>resource_model</i> of the profile <i>profile_name</i> has failed to start.</b> <b>Explanation:</b> The resource model failed to start because of an internal error. <b>User Response:</b> Distribute the profile again.
AMW2017E	<b>The resource model <i>resource_model</i> of the profile <i>profile_name</i> has failed to stop.</b> <b>Explanation:</b> The resource model failed to stop because of an internal error. <b>User Response:</b> Try to stop the resource model again.
AMW2019E	<b>The settings for the profile <i>profile_name</i> have not been saved.</b> <b>Explanation:</b> The profile settings have not been saved because of an internal error. <b>User Response:</b> Distribute the profile again.
AMW2021E	<b>Unable to connect the local database, Web address: <i>url_name</i>.</b> <b>Explanation:</b> The connection to the local database has failed. <b>User Response:</b> Restart the UNIX engine.
AMW2023E	<b>Unable to clean data of profile <i>profile_name</i>.</b> <b>Explanation:</b> The cleaning query has failed to remove data for the profile. <b>User Response:</b> Contact Customer Support for assistance.
AMW2025E	<b>The resource model initialization for the profile <i>profile_name</i>, resource model <i>resource_model</i>, has failed.</b> <b>Explanation:</b> The resource model initialization has failed because of an internal error. <b>User Response:</b> Distribute the profile again.
AMW2027W	<b>The start is deferred because the resource model is disabled. Profile <i>profile_name</i>, resource model <i>resource_model</i>.</b> <b>Explanation:</b> The resource model has been previously disabled. <b>User Response:</b> Enable the resource model and distribute the profile again.

---

| AMW2028E      **The resource model *resource\_model* of the profile *profile\_name* has not been stopped.**

|                    **Explanation:** The resource model failed to be stopped because of an internal error.

|                    **User Response:** Try to stop the resource model again.

---

| AMW2030E      **The profile *profile\_name* cannot run on the *interpreter\_name* interpreter.**

|                    **Explanation:** The machine interpreter is not a supported one.

|                    **User Response:** Distribute the profile on a workstation with a supported interpreter.

---

| AMW2031E      **The engine failed to start with error code: *error\_code*.**

|                    **Explanation:** The engine failed to start because of an internal failure.

|                    **User Response:** Contact Customer Support for assistance.

---

| AMW2032E      **The dependencies for the profile *profile\_name* cannot be downloaded and installed.**

|                    **Explanation:** An internal failure occurred during the downloading of engine dependencies.

|                    **User Response:** Contact Customer Support for assistance.

---

| AMW2033E      **I/O error on file *file\_name*. The error number is *error\_number*.**

|                    **Explanation:** A run-time error occurred during an attempt to open the file.

|                    **User Response:** Contact Customer Support for assistance.

---

| AMW2034E      **The distribution expired.**

|                    **Explanation:** The distribution status changed to expired state.

|                    **User Response:** Contact Customer Support for assistance.

---

| AMW2035E      **The distribution was canceled.**

|                    **Explanation:** The distribution status changed to canceled state.

|                    **User Response:** Contact Customer Support for assistance.

---

| AMW2036E      **Fatal error in file *file\_name* at line *file\_line*.**

|                    **Explanation:** an internal error occurred during the downloading of segments.

|                    **User Response:** Contact Customer Support for assistance.

---

| AMW2037E      **Cannot unzip file *file\_name*.**

|                    **Explanation:** an internal failure occurred during the downloading of engine dependencies.

|                    **User Response:** Contact Customer Support for assistance.

---

---

AMW2037E	<b>Cannot unzip file <i>file_name</i>.</b>
	<b>Explanation:</b> an internal failure occurred during the downloading of engine dependencies.
	<b>User Response:</b> Contact Customer Support for assistance.

---

AMW2039E	<b>Java failure on the endpoint. Java Runtime Environment is not installed or was not found. The engine could not be started.</b>
	<b>Explanation:</b> Java Runtime Environment (JRE) is not installed, or the link to it is not correct.
	<b>User Response:</b> Install Java Runtime Environment and run the DMLinkJre task. If you have already installed a supported version of JRE, ensure the the link to it was correctly created.

---

AMW2040E	<b>Java failure on the endpoint. The current version of Java Runtime Environment is not supported. The engine could not be started.</b>
	<b>Explanation:</b> The supported version of Java Runtime Environment (JRE) is 1.3.0. Previous versions are not supported.
	<b>User Response:</b> Install Java Runtime Environment, Version 1.3.0.

---

AMW2041E	<b>Java failure on the endpoint. The endpoint's operating system is not supported. The engine could not be started.</b>
	<b>Explanation:</b> The endpoint runs an operating system that is not supported.
	<b>User Response:</b> Contact Customer Support for assistance.

---

AMW2042E	<b>Java failure on the endpoint: Java Runtime Environment was not started. The engine could not be started.</b>
	<b>Explanation:</b> Java Runtime Environment (JRE) did not start.
	<b>User Response:</b> Contact Customer Support for assistance.

---

AMW2043E	<b>A Windows Management Instrumentation failure on the endpoint. Either Windows Management Instrumentation is not installed, or it is failing.</b>
	<b>Explanation:</b> Either Windows Management Instrumentation could not be installed, or it is failing.
	<b>User Response:</b> Install Windows Management Instrumentation. If it is already installed, restart it.

---

AMW2044E	<b>Database connection failure on the endpoint. Either the ODBC driver is missing or is an old version. The engine will work but without logging.</b>
	<b>Explanation:</b> The engine is not able to connect to the local database because the ODBC driver is missing, or is an old version.
	<b>User Response:</b> Update the ODBC driver of the machine. The ODBC driver for Microsoft Access 2000 must be installed on the endpoint.

---

AMW2045E	<b>General failure on the endpoint. The engine could not be started.</b>
	<b>Explanation:</b> The engine did not start.
	<b>User Response:</b> Contact Customer Support for assistance.

---

- |          |   |
|----------|---|
| AMW4400E | <p><b>The message properties file is missing or corrupt. An exception error was detected:</b> <i>I/O_exception</i>.</p> <p><b>Explanation:</b> This occurs when the JLog file cannot be found. The properties file should be in the Java classpath.</p> <p><b>User Response:</b> Ensure that the properties file exists. Check the syntax of the properties file. If necessary, extract the original log properties file from the util.jar file.</p>  |
| <hr/>    |   |
| AMW4401E | <p><b>The message properties file is missing or is corrupted. An exception error was detected:</b> <i>I/O_exception</i>.</p> <p><b>Explanation:</b> This occurs when the JLog file cannot be found. The properties file should be in the Java classpath. The log system could not load the message bundle in order to translate this string.</p> <p><b>User Response:</b> Ensure that the properties file exists. Check the syntax of the properties file. If necessary, extract the original log properties file from the util.jar file.</p>               |
| <hr/>    |   |
| AMW4402E | <p><b>An error occurred during the log system initialization. The log properties file might be corrupted. An exception error was detected:</b> <i>log_exception</i>.</p> <p><b>Explanation:</b> The logging system encountered an error when it tried to load the properties from the log properties file.</p> <p><b>User Response:</b> Check the syntax of the properties file. If necessary, extract the original log properties file from the util.jar file.</p>   |
| <hr/>    |   |
| AMW4403E | <p><b>An error occurred during the log system initialization. The log properties file might be corrupted. An exception error was detected:</b> <i>log_exception</i>.</p> <p><b>Explanation:</b> The logging system encountered an error when it tried to load the properties from the log properties file. The log system could not additionally load the message bundle to translate this string.</p> <p><b>User Response:</b> Check the syntax of the properties file. If necessary, extract the original log properties file from the util.jar file.</p> |
| <hr/>    |   |
| AMW4404E | <p><b>The message logger is no longer valid. No log messages can be written.</b></p> <p><b>Explanation:</b> An attempt was made to log a message where no logger existed. This may be a result of problems occurring during the log system initialization.</p> <p><b>User Response:</b> Check that the log system initialized correctly. Respond to any messages in the standard out stream that pertain to the log system.</p>   |
| <hr/>    |   |
| AMW4405E | <p><b>The message logger is no longer valid. No log messages can be written.</b></p> <p><b>Explanation:</b> An attempt was made to log a message where no logger existed. This may be a result of problems occurring during the log system initialization. This message also could not be translated, indicating that the translation bundle is missing.</p> <p><b>User Response:</b> Check that the log system initialized correctly. Respond to any messages in the standard out stream that pertain to the log system.</p>                               |
| <hr/>    |   |
| AMW4406E | <p><b>The trace logger is no longer valid. No trace messages can be written.</b></p> <p><b>Explanation:</b> An attempt was made to log a trace message where no logger existed. This may be a result of problems occurring during the log system initialization.</p> <p><b>User Response:</b> Check that the log system initialized correctly. Respond to any messages in the standard out stream that pertain to the log system.</p>   |
-

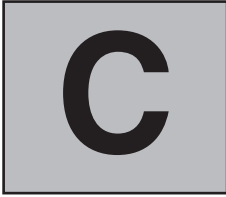


---

AMW4407E	<p><b>The trace logger is no longer valid. No trace messages can be written.</b></p> <p><b>Explanation:</b> An attempt was made to log a trace message where no logger existed. This may be a result of problems occurring during the log system initialization. also, this message could not be translated, which indicates that the translation bundle is missing.</p> <p><b>User Response:</b> Check that the log system initialized correctly. Respond to any messages in the standard out stream that pertain to the log system.</p>
<hr/>	
AMW4410E	<p><b>A connection could not be made to the Tivoli management region during login. A connection exception error was detected: <i>connection_exception_error</i>.</b></p> <p><b>Explanation:</b> An error occurred during the login process. The Web application server was unable to setup a connection with the Tivoli managed node.</p> <p><b>User Response:</b> Ensure the Tivoli managed environment is running. Check the connection between the Tivoli Monitoring Web Health Console application server and the managed node specified for the connection. Check the user ID and password that were specified.</p>
<hr/>	
AMW4411E	<p><b>The application server was unable to initialize its internal data structures during login. The following exception error was raised: <i>cache_exception</i>.</b></p> <p><b>Explanation:</b> An error occurred during the login process. The server was not able to set up the caching data structures.</p> <p><b>User Response:</b> Ensure the machine has enough memory and disk space. Check the status of the Web application server. Restart the Tivoli Monitoring Web Health Console.</p>
<hr/>	
AMW4412E	<p><b>An unknown error occurred during login. The following exception error was raised: <i>exception_error</i>.</b></p> <p><b>Explanation:</b> An error occurred during the login process. No more information is known.</p> <p><b>User Response:</b> Ensure the Tivoli managed environment is running. Check the connection between the Tivoli Monitoring Web Health Console application server and the managed node specified for the connection. Check the user ID and password that were specified.</p>
<hr/>	
AMW4413E	<p><b>The login was unsuccessful for the user <i>userid</i> on the managed node <i>managed_node</i>.</b></p> <p><b>Explanation:</b> An error occurred during the login process. The Web application server was unable to set up a connection with the Tivoli managed node. This may be the result of an incorrect user ID, password, or managed node.</p> <p><b>User Response:</b> Ensure the Tivoli Managed Environment is running. Check the connection between the Tivoli Monitoring Web Health Console application server and the managed node specified for the connection. Check the user ID and password that were specified.</p>
<hr/>	
AMW4414E	<p><b>The managed node <i>managed_node</i> is unreachable.</b></p> <p><b>Explanation:</b> An error occurred during the login process. The managed node was unreachable or was unable to respond.</p> <p><b>User Response:</b> Ensure the Tivoli Managed Environment is running. Check the connection between the Tivoli Monitoring Web Health Console application server and the managed node specified for the connection.</p>

---

- |          |   |
|----------|---|
| AMW4415E | <p><b>An error occurred when the user <i>userid</i> tried to log on to the managed node <i>managed_node</i>.</b></p> <p><b>Explanation:</b> An error occurred during the login process. The Web application server was unable to set up a connection with the Tivoli managed node.</p> <p><b>User Response:</b> Ensure the Tivoli managed environment is running. Check the connection between the Tivoli Monitoring Web Health Console application server and the managed node specified for the connection.</p> |
| AMW4416E | <p><b>The user ID or password is invalid for <i>userid</i> on the managed node <i>managed_node</i>.</b></p> <p><b>Explanation:</b> The user ID or password was invalid when the user attempted to log into the Tivoli management region.</p> <p><b>User Response:</b> Check the user name and password. Make sure the user has the authority to perform the requested action.</p>   |
| AMW4417W | <p><b>The default preferences could not be loaded from <i>preferences_file</i>.</b></p> <p><b>Explanation:</b> An I/O exception occurred while the default preferences were being loaded from the classpath.</p> <p><b>User Response:</b> This is a critical error only if the user did not set any preferences set yet. Check to ensure the classpath is correct and that the jar files shipped with the application have not been corrupted.</p>  |
| AMW4418E | <p><b>The default preferences could not be loaded from <i>preferences_file</i>. The file was not found. Default preferences will be used.</b></p> <p><b>Explanation:</b> A FileNotFoundException occurred when the user preferences were loaded. Check to ensure the file exists in the path specified.</p> <p><b>User Response:</b> Log off with this user and log on again. The preferences will be reset to the defaults.</p>  |
| AMW4419E | <p><b>The user preferences could not be loaded from <i>preferences_file</i>. An error occurred while reading the file. Default preferences will be used.</b></p> <p><b>Explanation:</b> An IOException occurred when the user preferences were loaded. Check to ensure the file exists in the path specified.</p> <p><b>User Response:</b> Log off with this user and log on again. The preferences will be reset to the defaults.</p>  |
| AMW4420I | <p><b>The user preferences could not be loaded from <i>preferences_file</i>. The file does not exist. This may be the first time the user has logged in. Default preferences will be used.</b></p> <p><b>Explanation:</b> A preferences file was not found for this user. This is an expected message when a user logs in for the first time.</p> <p><b>User Response:</b> None.</p>  |
| AMW4421E | <p><b>The user preferences could not be saved in <i>preferences_file</i>. An error occurred while the file was being read.</b></p> <p><b>Explanation:</b> An IOException occurred while an attempt was made to save the user's preferences.</p> <p><b>User Response:</b> check the permissions on the directory in which the file should be saved.</p>  |



# Troubleshooting

---

This appendix describes the logs and traces generated by the product, identifies the tools that are available to help you resolve problems on Windows endpoints, see “Endpoint Tools (Windows)” on page 193, and provides a section on problem determination, see “Problem Determination” on page 193.

In Tivoli Monitoring both log and trace files are provided to help in troubleshooting.

## Logs and Traces

When trying to resolve problems you should start by looking at the log file, if there is one. Log files are written in a readable format and stored in the **\$DBDIR/AMW/logs** directory for managed node logs or in the **\$LCF\_DATDIR/LCFNEW/AMW/logs** directory for endpoint logs. A utility is provided to convert them into XML. Currently log files are available only for non-windows endpoints.

If the problem cannot be resolved from the log file, then the trace file needs to be provided to Customer Support.

Log files are called **msg\_xxxx.log**. Trace files are called **trace\_xxxx.log**.

## The Log Record Format

The Log Record format is as follows:

```
<F>Date1<F>Date2<F>ProductID<F>Component<F>Server<F>ProcessID<F>MessageID<F>LogText
```

where:

**Date1** Time the log was produced specified in milliseconds, since January 1st 1970, for example, **1015343592000**

**Date2** Time the log was produced specified in GMT. It includes the date, time, and time zone, for example, **Tue Mar 05 15:53:12 2002 GMT**

**ProductID**

The three letter code assigned to the product for identifying its messages uniquely; **AMW**, for Tivoli Monitoring.

**Component**

Represents a run-time grouping of a product’s parts. If a product has multiple applications, the Component name will reflect the name of the application.

**Server**

Host name (managed node label or endpoint label).

**ProcessID**

The Process ID of the process that produced the log message.

**MessageId**

The unique numeric message identifier.

**LogText**

The message text.

**Trace Record Format**

The trace record format is as follows:

<F>Date1<F>Date2<F>ProductID<F>Component<F>Server<F>ProcessID<F>TraceLev<F>FileName  
<F>Method<F>Thread<F>LogText<F>Exception

where:

**Date1** Time the log was produced specified in milliseconds, since January 1st 1970, for example, **1015343592000**.

**Date2** Time the log was produced specified in GMT. It includes the date, time, and time zone, for example, **Tue Mar 05 15:53:12 2002 GMT**.

**ProductID**

The three letter code assigned to the product for identifying its messages uniquely, **AMW**, for Tivoli Monitoring.

**Component**

Represents a run-time grouping of a product's parts. If a product has multiple applications, the Component name will reflect the name of the application.

**Server**

Host name (managed node label or endpoint label).

**ProcessID**

The Process ID of the process that produced the log message. It is only present for Java processes.

**TraceLev**

The level of detail the trace represents:

- MAX**            Deep level of tracing
- MIN**            Minimum level of tracing
- MID**            A level between MIN and MAX
- OTHER**        A level other than MID, MID, MAX

**FileName**

Name of the file or class to which the trace refers

**Method**

The method in the class to which the trace refers

**Thread**

Thread to which the trace refers. Thread element refers to the platform-specific notion of a thread.

**LogText**

A description of the error.

**Exception**

Exception element will depend upon the specific language/platform. For Java, it will be a stack trace.

## Tool to Create XML File

The formatter program creates an XML-based file from the log or trace generated by Tivoli Monitoring. It is located on the Tivoli Monitoring Tools CD in the directory LogToXML. It accepts three parameters:

- The first parameter defines whether the product is dealing with a Log file or a Message file.
- The second parameter is the name of the source file (either a LOG or a MESSAGE file).
- The third parameter is the name of the file to be created in XML.

Here is an example:

```
prepareLog LOG ./trace_x.log ./trace_x.xml
```

**Note:** Before running the prepare Log program, the Java Virtual Machine 1.3.0 path must be set.

The logs and traces provided by Tivoli Monitoring are as follows:

- “Server Traces” on page 181
- “Gateway Traces” on page 182
- Endpoint logs
  - “Windows Endpoint Logs” on page 186
  - “Profile Distribution Endpoint Logs” on page 187
  - “Non-Windows Endpoints Logs” on page 187

## Server Traces

At the server, the product maintains a Distribution log, in addition to the Tivoli Management Framework logs.

### Distribution Logs

A log is maintained at the server containing the distribution results when a profile is pushed (distributed) to subscribers, as follows:

#### Name

**msg\_profile-name.log**

#### Location

\$DBDIR/AMW/logs

#### Configuration

To configure the log when distributing by means of the command **wdmistrib**, use the options **-e**, **-i**, and **-w**. There are no configuration options when the GUI is used to perform a distribution. The default is that all options are specified.

### Tivoli Management Framework Logs

See *Tivoli Management Framework: User's Guide* and *Tivoli Management Framework: Reference Manual* for full details of the logs available in Tivoli Management Framework.

### Profile Core Trace

A log is maintained at the gateway of the activities carried out when the profile core engine is running. It records all messages output by the process **tmw2k\_profile\_core**. The details are as follows:

**Process Name**

tmw2k\_profile\_core

**Trace Name**

**trace\_tmnt\_profile\_core*n*.log** (where *n* is a number in the range 1 - 9; as each file becomes full the number is incremented, cycling back to 1 when file 9 is full.)

**Location**

\$DBDIR/AMW/logs

**Configuration**

To configure the log, use the command **wdmconfig** to change the following variables:

- core.trace\_level: min=0; max=2; default=1
- core.trace\_size: default=500000 (0.5 MB)

## Gateway Traces

At the gateway, the product maintains five traces:

- Heartbeat Engine Traces
- Task Engine Trace
- Tivoli Business Systems Manager Engine Traces
- Tivoli Business Systems Manager Adapter Trace
- Tivoli Business Systems Manager Transport Trace
- Profile Core Trace
- Request Manager Trace

All managed node components produce traces. Traces can be configured using the command:

**wdmconfig -D component\_name.trace\_level=value**

**wdmconfig -D component\_name.trace\_size=value**

where *component\_name* is the name of the component that produces the trace:

- **heartbeat**, for the heartbeat engine
- **task**, for the task engine
- **tbsma**, for the Tivoli Business Systems Manager engine
- **core**, for the profile core engine
- **request-manager**, for the request manager engine
- **gw**, for the endpoint upcall

You can activate (and deactivate) traces for all components by using the component name **dmml**, for example: **wdmconfig -D dmml.trace\_level = 4**

## Heartbeat Engine Traces

A trace is maintained at the gateway of the activities carried out when the heartbeat engine is running. It records all messages output by the process **tmnt\_hb\_eng**. The details are as follows:

**Process Name**tmnt\_hb\_eng

---

**Trace Name**

**trace\_tmnt\_hb\_engn.log** (where *n* is a number in the range 1 - 9; as each file becomes full the number is incremented, cycling back to 1 when file 9 is full.)

**Location**

\$DBDIR/AMW/logs

**Configuration**

To configure the log, use the command:

**wdmconfig -D heartbeat.trace\_level = <value1>**

**wdmconfig -D heartbeat.trace\_size = <value2>** where:

- <value1> : min=0; max=2; default=1
- <value2> : default=500000 (0.5 MB)

The heartbeat engine must be stopped using the **wdmmn** command (see “wdmmn” on page 139), and started using the **wdmheartbeat** command (see “wdmheartbeat” on page 131) for any changes in the configuration variables to take effect.

**Task Engine Trace**

A trace is maintained at the gateway of the activities carried out when the task engine is running to perform tasks on attached endpoints as determined by resource model definitions. It records all messages output by the process **tmnt\_task\_eng**. The details are as follows:

**Process Name**

tmnt\_task\_eng

**Trace Name**

**trace\_tmnt\_task\_engn.log** (where *n* is a number in the range 1 - 9; as each file becomes full the number is incremented, cycling back to 1 when file 9 is full.)

**Location**

\$DBDIR/AMW/logs

**Configuration**

To configure the log, use the command:

**wdmconfig -D task.trace\_level = <value1>**

**wdmconfig -D task.trace\_size = <value2>** where:

- <value1> : min=0; max=2; default=1
- <value2> : default=500000 (0.5 MB)

The task engine must be started and stopped using the **wdmmn** command (see “wdmmn” on page 139), for any changes in the configuration variables to take effect.

**Tivoli Business Systems Manager Engine Traces**

Two traces are maintained at the gateway of the activities carried out when the Tivoli Business Systems Manager engine is running. They record all messages output by the process **tmnt\_tbsm\_eng**, which implements the CORBA methods. The details are as follows:

**Process Names**

tmnt\_tbsm\_eng, tmnt\_tbsm\_wrapper

**Trace Names**

**trace\_tmnt\_tbsm\_engn.log** (where *n* is a number in the range 1 - 9; as each file becomes full the number is incremented, cycling back to 1 when file 9 is full.)

**trace\_tmnt\_tbsm\_wrappern.log** (where *n* is a number in the range 1 - 9; as each file becomes full the number is incremented, cycling back to 1 when file 9 is full.)

**Location**

\$DBDIR/AMW/logs

**Configuration**

To configure the log, use the command:

**wdmconfig -D tbsma.trace\_level = <value1>**

**wdmconfig -D tbsma.trace\_size = <value2>** where:

- <value1> : min=0; max=2; default=1
- <value2> : default=500000 (0.5 MB)

The Tivoli Business Systems Manager engine must be stopped using the **wdmmn** command (see “wdmmn” on page 139), for any changes in the configuration variables to take effect (it will be restarted automatically by the next discovery request).

**Endpoint Upcall Traces**

A trace is maintained at the gateway of the upcall messages sent to the gateway from the endpoints. It contains details of the following:

- endpoint registration upcalls
- events and indications sent from the endpoint component
- task upcalls

It records all messages output by the process **tmnt\_gtw\_eng**, which receives the upcalls. The details are as follows:

**Process Name**

tmnt\_gtw\_eng

**Trace Name**

**trace\_tmnt\_gtw\_engn.log** (where *n* is a number in the range 1 - 9; as each file becomes full the number is incremented, cycling back to 1 when file 9 is full.)

**Location****UNIX or Linux**

/tmp/traces

**Windows**

\$DBDIR/TMP/traces

**Configuration**

To configure the log, use the command:

**wdmconfig -D gw.trace\_level = <value1>**

**wdmconfig -D gw.trace\_size = <value2>** where:

- <value1> : min=0; max=2; default=1
- <value2> : default=500000 (0.5 MB)



## Tivoli Business Systems Manager Adapter Trace

A log is maintained at the gateway of the activities carried out by the Tivoli Business Systems Manager adapter, as follows:

### Log Name

User-defined; default is `dm.trc` (when the log is full it is renamed to *filename.old*, deleting any existing file with that name, and a new log file is created.)

### Location

As defined in the **wdmconfig** configuration variable *adapter.working.dir* (default is `$DBDIR/dmm1`).

### Configuration

To configure the log, use the **wdmconfig** command (see page 111) to modify the variables *trace.filename*, *adapter.trace.enable* and *adapter.trace.level* (see “Configuring Tivoli Business Systems Manager Adapter” on page 74 for more details). You should note that traces generated by the Tivoli Business Systems Manager Transport trace are also stored in this log.

## Tivoli Business Systems Manager Transport Trace

A trace is maintained at the gateway of the activities carried out by the Tivoli Business Systems Manager adapter when sending events or messages to the Tivoli Business Systems Manager CommonListener, as follows:

### Trace Name

User-defined; default is `dm.trc` (when the trace is full it is renamed to *filename.old*, deleting any existing file with that name, and a new trace file is created.)

### Location

As defined in the **wdmconfig** configuration variable *adapter.working.dir* (default is `$DBDIR/dmm1`).

### Configuration

To configure the trace, use the **wdmconfig** command (see page 111) to modify the variables *trace.filename*, *transport.trace.enable* and *adapter.transport.level* (see “Configuring Tivoli Business Systems Manager Adapter” on page 74 for more details). You should note that traces generated by the Tivoli Business Systems Manager Adapter trace are also stored in this trace.

## Request Manager Trace

A trace is maintained at the gateway of the activities carried out when the request manager processor is running. It records all messages output by the process `tmnt_rm_eng`. The details are as follows:

### Process Name

`tmnt_trm_eng`

### Trace Name

`trace_tmnt_rm_engn.log` (where *n* is a number in the range 1 - 9; as each file becomes full the number is incremented, cycling back to 1 when file 9 is full.)

### Location

`$DBDIR/AMW/logs`

### Configuration

To configure the log, use the command:

```
wdmconfig -D request-manager.trace_level = <value1>
```

**wdmconfig -D request-manager.trace\_size = <value2>** where:

- <value1> : min=0; max=2; default=1
- <value2> : default=500000 (0.5 MB)

The request manager processor must be started and stopped using the **wdmmn** command (see “wdmmn” on page 139), for any changes in the configuration variables to take effect.

## Endpoint Logs and Traces

Logs and traces maintained at Windows endpoints are different from those at non-Windows endpoints.

### Windows Endpoint Logs

The product maintains an Endpoint Engine Log at Windows endpoints, and there are also logs maintained by WMI.

### Endpoint Engine Log

The main trace log generated by the Tivoli Monitoring engine at Windows endpoints records the activities of the endpoint engine, as follows:

#### Process Name

Tmw2k and Tmw2k\_ep

#### Trace Name

**Tmw2k.log** (when the log is full the oldest 20% of messages are deleted)

#### Location

\$LCF\_DATDIR/LCFNEW/Tmw2k

#### Configuration

To configure the trace issue the command **wdmtrceng** from the Server or managed node, identifying the endpoint at which you want to configure the log. You can set any of the following parameters:

- Trace filename
- Trace level, from 0 to 2
- Maximum file size

Each line in the log contains the following columns:

- Date
- Trace Level
- Component
- Thread Id
- Message

To implement any changes made with the **wdmtrceng** command you must stop and start the endpoint engine.

### WMI Log

The WMI log files records the activities of WMI in collecting the data required by the resource models. The WMI log files are located in the directory:  
%SystemRoot%/system32/wbem/logs For details see the WMI documentation.

## Profile Distribution Endpoint Logs

The product maintains a Profile Distribution Endpoint log.

### Endpoint Engine Log

The trace log generated by the Tivoli Monitoring distribution process records the MDist2 activities of the engine update, as follows:

#### Process Name

Tmw2k\_ep

#### Log Name

**lcf.d.log**

#### Location

\$LCF\_DATDIR

#### Configuration

To configure the log issue the command **lcf.d** to set the `-d` level of trace (3 is suggested for tracing information).

## Non-Windows Endpoints Logs

The product maintains four logs at the endpoint:

- Endpoint Engine Update Log
- Endpoint Engine Log and Trace
- Endpoint Native Trace
- Endpoint JMX Log

### Endpoint Engine Update Log

This log maintains details of the activities of the engine update process, which is the process that launches and controls the endpoint engine, as follows:

#### Process Name

dmxeu

#### Log Name

**trace\_dmxeu.log** (when the log is full it is renamed to `dmxeu.old`, deleting any existing file with that name, and a new log file is created.)

#### Location

\$LCFDATDIR/LCFNEW/AMW/logs

#### Configuration

To configure the log issue the command **wdmtrceng** from the Server or managed node, identifying the endpoint at which you want to configure the log. You should note that this command maintains a common configuration for all logs at a non-Windows endpoint. You can set any of the following parameters:

- Trace level, from 0 to 2
- Maximum file size: default=2500000 (2.5MB)

### Endpoint Engine Log and Trace

This log maintains details of the activities of the engine, which is the process that runs the resource models and sends events and indications to the gateway, as follows:

**Process Name**

dmxengine

**Log Name**

**msg\_dmxengine.log** (when the log is full it is renamed to `dmxengine.old`, deleting any existing file with that name, and a new log file is created.)

**Trace Name**

**trace\_dmxengine.log** (when the log is full it is renamed to `dmxengine.old`, deleting any existing file with that name, and a new log file is created.)

**Location**

`$LCF_DATDIR/LCFNEW/AMW/logs`

**Configuration**

To configure the log issue the command **wdmtrceng** from the Server or managed node, identifying the endpoint at which you want to configure the log. You should note that this command maintains a common configuration for all logs at a non-Windows endpoint. You can set any of the following parameters:

- Trace level, from 0 to 2
- Maximum file size: default=2500000 (2.5MB)

### Endpoint Native Trace

This log maintains details of the activities of the native processes which obtain the resource information required by the resource models, as follows:

**Process Name**

dmxntv

**Log Name**

**trace\_dmxntv.log** (when the log is full it is renamed to `dmxntv.old`, deleting any existing file with that name, and a new log file is created.)

**Location**

`$LCFDATDIR/LCFNEW/AMW/logs`

**Configuration**

To configure the log issue the command **wdmtrceng** from the Server or managed node, identifying the endpoint at which you want to configure the log. You should note that this command maintains a common configuration for all logs at a non-Windows endpoint. You can set any of the following parameters:

- Trace level, from 0 to 2
- Maximum file size: default=2500000 (2.5MB)

### Endpoint JMX Log

This log maintains details of the activities of the JMX process, which is a Tivoli implementation of Java Management Extension. It is only written when the trace level is set to 3. The details are as follows:

**Process Name**

Tmx4j

**Log Name**

**Tmx4j\_1.log** (when the log is full it is renamed to `Tmx4j_2.log`, deleting any existing file with that name, and a new log file is created.)

**Location**

```
$LCF_DATDIR/LCFNEW/Tmw2k/UNIX
```

**Configuration**

To configure the log issue the command **wdmtrceng** from the Server or managed node, identifying the endpoint at which you want to configure the log. You should note that this command maintains a common configuration for all logs at a non-Windows endpoint. You can set any of the following parameters:

- Trace level, 3 (verbose)
- Maximum file size: default=1000000 (1.0MB)

**Web Health Console Logs and Traces**

The Web Health Console has a facility for both standard message logging, as well as advanced debug tracing. Message logging and minimum level debug tracing are always on and writing to their own files. These files can be found under the standard Tivoli Monitoring log location, /Tivoli/AMW/logs.

**Modifying Web Health Console Tracing Parameters**

Tracing can be adjusted by modifying the tracing parameters for the Web Health Console application.

- Edit the file:

```
WHC_INSTALL_DIR\installedApps\dm.ear\dm.war\  
WEB-INF\classes\com\ibm\dm\web\util\PDLog.properties
```

- Change the line:

```
tmeLogger.trc.level=DEBUG_MIN
```

to:

```
tmeLogger.trc.level=DEBUG_MID
```

or:

```
tmeLogger.trc.level=DEBUG_MAX
```

depending on how much tracing you want, MIN, MID or MAX. MID provides a good amount of Web Health Console operation, MAX provides a great deal of detailed internal operation.

You can also adjust the lines:

```
file.maxFiles=3  
file.maxFileSize=1024
```

to change the number of trace files written and the max size of the files before it rolls over to a new file.

Once these changes are made, stop and start the WebSphere Application Server to enable the changes.

**Note:** The Web Health Console will run slower while in MID or MAX tracing. This should be turned back to MIN as soon as possible.

## Autotrace

Autotrace is a process tracing software from The Kernel Group Inc. (TKG), and is available on Solaris, HP-UX, Windows, and AIX platforms. It is used to collect information at an endpoint which is stored in a configurable memory buffer. You choose when to write a snapshot of the buffer to a file, and you then send the file to Tivoli Customer Support for analysis. The information written to the trace file consists of the input and output parameters for each process call.

Autotrace consists of two elements:

### **At the endpoint and Tivoli management region server**

A trace collector, enabled and controlled by you.

### **At Tivoli Customer Support**

A trace analyzer operated by the Tivoli Customer Support staff.

## Installing the Autotrace trace collector

Autotrace, Version 3.1.5 is supplied on the product CD. It can be installed on an endpoint, server, or managed node. It is installed on an endpoint using the following procedure:

1. Copy the file `Tools/TKG/tkg315.tar` to the directory on the endpoint where you want it to be installed.
2. Run the following command to unpack Autotrace:  
`tar -xvf tkg315.tar`
3. Set up the following environment variables, where `$HOME` is the directory where you have unpacked Autotrace.

### **AIX** Issue the following commands:

```
PATH=$HOME/aix4-r1:$PATH
export PATH
LIBPATH=$HOME/aix4-r1:$LIBPATH
export LIBPATH
```

### **HPUX**

Issue the following commands:

```
PATH=$HOME/hpux10:$PATH
export PATH
SHLIB_PATH=$HOME/hpux10:$SHLIB_PATH
export SHLIB_PATH
```

### **Solaris**

Issue the following commands:

```
PATH=$HOME/solaris2:$PATH
export PATH
LD_LIBRARY_PATH=$HOME/solaris2:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH
```

### **Windows**

Refer to Autotrace documentation.

**Note:** If you are already using Autotrace on the endpoint for another product, there is no need to install Autotrace again.

## Configuring Autotrace

Autotrace is configured by editing the following files: product and config files, which can be found in the directory `$HOME/etc/`.

## Product File

The product file lets you configure the following aspects:

### Assign a name to a product id

Autotrace-enabled products are compiled with a product id that is recognized by Autotrace. Tivoli Monitoring is supplied with a product id = “49420003”. The product id must not be changed in this file.

You can associate a name with the product id so that the product can be referred to using the name instead of the product id, when using commands. The default name for Tivoli Monitoring is “DM”, but it can be changed to a name of your choice, by editing this file.

An example of the file as shipped, is as follows:

```
# AutoTrace product install database
#
# This file contains a partial list of products using AutoTrace.
# If the product is instrumented, but is not installed into the control
# channel via `atctl init', it is NOT traced.
#
# The file format is flat ascii, one product record per line.
# There are three fields, separated by whitespace (space and tabs);
# the first is the product id expressed as a hex value, the second
# is a short version of the product's name, and the last field is
# the product's home directory. All fields but the first are optional.
# A field value of dash '-' is interpreted as an empty field.
#
# Product Name Home
49420001 TME -
49420003 DM -
```

## Config File

The config file lets you configure the following aspects for each channel:

### Choose the trace channel (chan)

Autotrace is capable of monitoring more than one product simultaneously, by assigning the product id of each product being monitored to one of up to 255 trace channels. Tivoli Monitoring is supplied with the default of channel 21. However, you can choose to change this, by editing this file, and setting the value of the **chan** attribute of the **product DM** stanza to a different value.

### Set the channel buffer size (size)

Autotrace saves data into a trace channel buffer in memory, the size of which can be configured, from 100K to 256M (the supplied default is 5M). When the channel is full, the oldest data is overwritten first. To change the value edit the **size** attribute of the **channel 21** stanza.

**Note:** Autotrace utilizes shared memory segments for trace channels. By default, Solaris restricts the size of shared memory segments to 1MB. The default settings in Autotrace use shared memory segments or trace channels of up to 10MB. To allow the Solaris system to create these larger shared memory segments you must add the following line to `/etc/system`:

```
set shmsys:shminfo_shmmax=0xffffffff
```

and reboot the machine. This configuration option sets the maximum size of a shared memory segment to unlimited. Other values are also possible; however a trace channel cannot be larger than the maximum allowed by this statement.

### Set process name length (nlen)

By default, Autotrace uses the first 8 characters of the process name when writing to the trace channel. You can change this to any other multiple of 4 up to 64, by editing the **nlen** attribute of the **channel 21** stanza.

An example of the file as shipped, is as follows:

```
# AutoTrace configuration file
#
# This file is used to customize trace configurations.
# It is installed (encoded into AutoTrace control channel)
# by `atctl init' (sync) or `atctl config replace'.
# Processes only examine the installed configuration when
# they start; existing processes ignore any changes.
# The installed configuration may be displayed using `atctl config'.
#
# The file format is a relaxed stanza. Stanzas may be defined
# in any order desired. There are four keywords that introduce
# a new stanza; most take an argument to restrict their effect:
#   default
#   product $id      # 32-bit value or a product name
#   process $name    # process name as invoked
#   channel $number # 1..255, inclusive
#
# Each stanza accepts a set of attributes, used to define the
# configuration:
#   chan $number    # 1..255, inclusive
#   size $number    # 100k..256M, inclusive
#   nlen $number    # 0..32, inclusive, multiple of 4 (0,4,8,12,...)
#   auth $number    # 0 or 1
#
default:
  chan 1
  size 1m
  nlen 30
product DM:
  chan 21
channel 21:
  size 5M
  nlen 8
```

## Controlling Autotrace

The trace collector is controlled by the following command:

**atctl** Controls the Autotrace trace collector.

### Syntax

**atctl init etc**

**atctl on DM all**

**atctl off DM all**

**atctl info**

**atctl snap 21 *snap\_file***

### Description

This command lets you initialize Autotrace, activate the trace for Tivoli Monitoring, view the trace and copy a snapshot into a file. The options shown here are a subset of the commands supplied with Autotrace; for full details see the Autotrace documentation.



## Options

### **init etc**

Initializes Autotrace, using the configuration file (`config`) found in the directory defined as the second argument, in this case `etc`.

### **on DM all**

Activates the trace for the product defined in the second argument (in the case of Tivoli Monitoring this, by default, is “DM”) and for the processes defined in the third argument; you will normally define **all**, but you may be requested by Tivoli Customer Support to input one or more specific process ids here, which will be supplied to you.

### **off DM all**

Deactivates the trace on all processes for the specified product.

### **info**

Displays the status of all active channels of the trace. The information comprises:

- Number, size and version of the library that created the channel
- Number of processes that have ever attached to the channel
- Number of records that have been written to the channel (but not the number of records currently in the channel)
- The time that the channel was either created or reset (snapped)

### **snap 21 snap\_file**

Copies the contents of the indicated trace channel (in this case the DM default of channel 21) to a snap file called `snapxy.at`, where:

**x** = Channel number  
**y** = A unique number

The snap resets the channel to zero to allow for the accumulation of more information.

## Endpoint Tools (Windows)

The diagnostic tools available in a Windows environment are provided with the WMI software. To use any of these tools please refer to the proprietary documentation. Those that are useful for troubleshooting at the endpoint are as follows;

### **wbencntl**

This is a configuration tool that allows you to change the WMI trace settings and some backup options. It is installed automatically with WMI

### **wbemtest**

Allows you to make synchronous and asynchronous queries on WMI objects. It can be used to simulate the queries carried out by the Tivoli Monitoring engine. It can be used on remote Windows endpoints. Installed automatically with WMI.

### **cimstudio**

This allows you to navigate the CIM repository and query the CIM instances. It also provides WMI SDK help. Installed automatically with WMI

## Problem Determination

The following are some common problems with their resolutions:

Table 15. Problems and Their Workarounds

Problem	Solution
Missing prerequisites	<ul style="list-style-type: none"> <li>■ A prerequisite is missing, for example SNMP for the TCP/IP resource model, Network Monitoring Agent for the Network InterfaceCard resource model, or an operating system other than Windows 2000 for the Printer resource model, etc.</li> <li>■ Dependency (MOF or Provider or other) not set in the package.</li> </ul>
Commands <b>wdmcmd</b> and <b>wdmlseng</b> issued from a server with Tivoli Monitoring 5.1 through a managed node with Tivoli Monitoring (Advanced Edition) 4.1 fail on endpoints with Tivoli Monitoring (Advanced Edition) 4.1	<ul style="list-style-type: none"> <li>■ Run the <b>DMFixCliCmdsOn41MNS</b> task that is provided with Tivoli Monitoring 5.1 on the managed node. <b>Note:</b> Executing this task enables the server to issue only the <b>wdmcmd</b> and <b>wdmlseng</b> commands through the managed node with Tivoli Monitoring (Advanced Edition) 4.1 on endpoints with Tivoli Monitoring (Advanced Edition) 4.1.</li> </ul>
Renaming an endpoint does not update the endpoint cache on the corresponding gateway	<p>When renaming an endpoint, use the <b>wep set_label</b> command with the <b>-s</b> option, as in the following syntax:</p> <pre>wep old_endpoint_label set_label -s new_endpoint_label</pre> <p>This enables the information in the endpoint cache to be updated on the gateway, as soon as the endpoint sends an upcall to the gateway.</p>
A server with Tivoli Monitoring 5.1 distributes profiles on endpoints that are running Tivoli Monitoring (Advanced Edition) 4.1, with the following results: On UNIX endpoints the profile distribution is performed, but the resource models created with Tivoli Monitoring 5.1 detects an error ■ On Windows endpoints, the profile distribution fails	<p>Run the <code>dm_endpoint_uninstall.sh</code> script at each endpoint component. For more information about running the script, see “Uninstalling the Endpoint Components” on page 34.</p>
Events not sent to Tivoli Enterprise Console	<ul style="list-style-type: none"> <li>■ Tivoli Enterprise Console server not specified in the profile.</li> <li>■ ACF not installed on the gateway when sending secure events.</li> <li>■ Baroc not installed on the server.</li> </ul>
Uninstall at Windows endpoint fails	<ul style="list-style-type: none"> <li>■ Uninstall at a Windows endpoint can fail because a provider is locked by WMI. To resolve the problem, stop the Windows Management Instrumentation from the Windows Control Panel, and retry after about 15 minutes.</li> </ul>
Commands fail on endpoints with error in profile name	<ul style="list-style-type: none"> <li>■ Check that the profile has been fully qualified with the region name. The correct syntax is <i>profile#region</i>.</li> </ul>
Working with UNIX or Linux endpoints, you have uninstalled and reinstalled the server or gateway, and the endpoint does not respond correctly	<ul style="list-style-type: none"> <li>■ Stop the product engine and the Tivoli Management Agent at all endpoints, and wait until you are certain that both processes have stopped.</li> <li>■ Delete the following files in the data directory within the engine installation directory at all endpoints: <ul style="list-style-type: none"> <li>*.dmprf</li> <li>engine.pid</li> </ul> </li> <li>■ Restart the agent and the endpoint engine at all endpoints.</li> </ul>

Table 15. Problems and Their Workarounds (continued)

Problem	Solution
You have uninstalled and reinstalled the product at the server or managed node, and the managed node functions (heartbeat engine, Tivoli Business Systems Manager engine, task engine) are not working correctly	<ul style="list-style-type: none"> <li>■ Restart the gateway</li> </ul>
Messages and events from the product are appearing in English on a non-English language Tivoli Enterprise Console server	<ul style="list-style-type: none"> <li>■ Use the <b>wdmrm</b> command (see page 144) with the <b>-addcat</b> option to install the appropriate NLS catalog for the language in question.</li> </ul>
A <b>wmdiscovery</b> command fails with the error: REGISTER TIMEOUT	<ul style="list-style-type: none"> <li>■ The problem is probably caused by a misalignment of the time settings of the systems where the Tivoli Business Systems Manager's Adapter and CommonListener are installed. The Adapter expects that the CommonListener will respond to a discovery command within a configurable time-out (the default is 300 seconds). If the systems are out of synchronization by more than this value, the command will fail with the indicated error. Two possible solutions are as follows: <ul style="list-style-type: none"> <li>• Adjust the time settings on the two systems to be the same.</li> <li>• If the time settings cannot be changed, change the time-out value by using the <b>wdmconfig -D adapter.register.timeout=time</b> command, where <i>time</i> is a number of seconds greater than the time difference between the two systems.</li> </ul> </li> <li>■ Network problem between the Adapter and the CommonListener, causing a genuine time-out to occur. Resolve the problem and retry the command.</li> <li>■ Incorrect configuration of the Tivoli Business Systems Manager Adapter, which could have, for example, an incorrect IP address for the CommonListener. Check the settings are correct and retry the command. If the problem persists reset the time-out to a large value, and try again.</li> <li>■ The port used by the Tivoli Business Systems Manager Adapter to communicate with the CommonListener is busy. Wait for a brief period, then retry the command. If the problem persists, modify the port number by issuing the <b>wdmconfig</b> command to change the values for the keys <b>transport.request.port</b> and <b>transport.response.port</b>. These values are set by default to 6969, and should be changed to the same common value that you know is available for use.</li> </ul>
The installation of JRE with SIS fails when the chosen install path is on a system running Windows NT Workstation, and the path contains a directory with spaces in its name.	<ul style="list-style-type: none"> <li>■ The directory name must be specified between single quotes, as in this example: <code>D:\'Program Files'\jre</code>.</li> </ul>
A distribution fails with the error that JRE was not found.	<ul style="list-style-type: none"> <li>■ Check that the correct version (1.3.0) of JRE is installed on the target system, and run the <b>DMLinkJre</b> task to ensure that the product is linked to the directory where that version resides on the target system. It is advisable to use the IBM version of JRE, to facilitate any support activity that may be necessary.</li> </ul>

Table 15. Problems and Their Workarounds (continued)

Problem	Solution
A distribution fails for any reason not connected with the product during the installation of the resource model on the endpoint, such as the disk is full.	<ul style="list-style-type: none"> <li>■ One option is to run the <b>DMEndpointUninstall</b> task. This cleans up all product files on the endpoint. When you retry the distribution, the product components will automatically be reinstalled.</li> </ul>
A distribution fails with a message containing the following sentence: “The specified segment ID (<RM name>@<interp>) with version (x.x) cannot be removed because it is currently in use. Try again later.”	<ol style="list-style-type: none"> <li>1. List all repeaters in your Tivoli management region using wrpt with no option</li> <li>2. For every repeater, check if the Mdist2 segment indicated by the message is in use by using the command (wdepot &lt;repeater&gt; list -l, see “Reference count” counter).</li> <li>3. If greater than zero, check if any pending Mdist2 distribution is being processed by the repeater (wmdist -l &lt;repeater&gt;). If yes, wait until it ends normally or, if the distribution is not in a useful state, cancel it and then run the following command: “wdepot &lt;repeater&gt; delete id^version”.</li> <li>4. Repeat step 2 and 3 until the mdist2 segment is no more referenced.(See TMF Reference Manual for more details on wdepot and wmdist commands.)</li> </ol>

## Web Health Console Troubleshooting

The Web Health Console is an application that works and interacts with other IBM Tivoli software. Understanding the very basics of these interactions will enable basic troubleshooting of the Web Health Console software.

### Tivoli Monitoring 5.1 Internal Processes

The Web Health Console communicates with Tivoli Monitoring in two different ways. The first is directly to Tivoli Monitoring Endpoint engines, and the second is to the Tivoli Monitoring Request Manager processes running on Managed Nodes.

#### Tivoli Monitoring Endpoint engine

This is where the Resource Models run on individual endpoints, the means of collecting and storing data. The Web Health Console connects to the endpoints to retrieve certain types of information, as well as run commands against the endpoint (start engine, and so on)

#### Request Manager Process

This process is the aggregating process that collects endpoint information to make frequent endpoint calls more efficient, putting less stress on the Tivoli Monitoring system.

### WebSphere Application Server and HTTPServer

Two separate but closely linked piece of software in this solution is the WebSphere Application Server, and IBM Apache HTTPServer.

WebSphere Application Server is the application server which provides an environment for the J2EE enterprise Web Health Console application. It provides services which allow the XSLT, JSP, Servlets and Java components of the Web Health Console to run. The HTTPServer provides a scalable and robust implementation of the HTTP protocol, which

allows the Web Health Console to intercept Web requests. The HTTPServer primarily forwards the requests it receives back to the WebSphere Application Server to be handled by the Web Health Console.

## Unable to Connect to the Main Web Health Console Web Page

### Incorrect main Web Health Console page

Make sure that you are connecting to the proper main Web Health Console page. The correct URL is: `http://machineName/dmwhc`

### Incorrect login information

Make sure the host name is correct and that you can connect to it using the Tivoli desktop. It is possible that you need to fully qualify the Tivoli management region server host name in order to connect. Verify that the user and password is correct and that you can connect to it using the Tivoli desktop.

### HTTPServer is not running

Restart the HTTPServer and try logging in again.

### WebSphere Application Server is not running

Restart the WebSphere Application Server HTTPServer and try logging in again.

## Slow or Unresponsive Web Health Console Screens or Panels

Because the Tivoli Monitoring data is retrieved from the Tivoli Monitoring Request Manager process, it is necessary for the Request Manager to retrieve the data before it can be sent to the Web Health Console. Many things can effect the load of the Request Manager, including the number of Web Health Console users, the number of WebSphere Application servers connecting to the Tivoli management region processes, and the number of endpoints in the Tivoli management region.

Other operations from the Web Health Console connect directly to the endpoints, and the response time for retrieving data depends on the response time of the Tivoli Monitoring engine in servicing the data request. This time depends on the machine speed and performance, as well as the Tivoli Monitoring engine tasks.

Configure the Request Manager by changing the number of threads. See section on configuration.

## Unable to Perform any Operations at Some Random Point during a Web Health Console Session

### The main Tivoli process (oserv) may be offline or had a "reexec" run against it.

Try logging out of the current Web Health Console sessions and log back in. Logging back in will reset the Tivoli connection and clear the problem.

### WebSphere Application Server or the HTTPServer is not responding or is offline

This is unusual once it has successfully started, but is possible. Refer to the "Unable to connect to the main Web Health Console web page" above, with the WebSphere Application Server and HTTPServer restart steps.

## Error Messages Displaying Graphs

### Invalid environment set while starting WebSphere Application Server.

If, when generating Historical or Online Metric graphs, you see the following error message:

```
AMW4805E An error occurred while creating the graph
```

Make sure of the following in the UNIX environment where WebSphere was started:

1. The DISPLAY variable was set correctly. The DISPLAY variable must be set correctly in the shell environment where WebSphere is started. This must be done before WebSphere is started. If your Unix machine has an X Server and you intend to stay logged on to the machine after WebSphere is started, then you can set DISPLAY to point at the X Server on the local machine. For example,

```
export DISPLAY=:0.0
<WEBSPPHERE_DIR>/bin/startServer.sh
```

If your Unix machine does not have an X Server or you intend to log off the machine after WebSphere is started, then you must set DISPLAY to another machine where you can obtain an X client session - a Windows machine running Exceed, or another Unix machine where which has an X Server running and a logged on user. For example,

```
export DISPLAY=<ip_address>:0.0
<WEBSPPHERE_DIR>/bin/startServer.sh
```

2. The user who started the WebSphere process and the host on which WebSphere was started each have authority to access the X Server that is being pointed to by the DISPLAY variable. Authority to access an X Server is dictated by the xhost command. For example,

```
xhost +<hostname>
xhost +<username>
```

## Other Possible Problems

### WebSphere Application Server not starting up properly

This is an unlikely situation assuming the installation succeeded to begin with. Refer to the trouble shooting section on the installation of the WebSphere Application Server.

### HTTPServer not starting up properly

This is an unlikely situation assuming the installation succeeded to begin with. How to check: Refer to the trouble shooting section on the installation of the WebSphere Application Server.

### Tivoli Monitoring Endpoint Engine not up or able to be contacted

If an endpoint cannot be contacted (either by the Web Health Console or the Tivoli Monitoring Request Manager directly), there are several possible reasons. In order to check if the endpoint has "logged in" properly to the Tivoli Monitoring middle layer processes, you can run the command `wdmmncache -l`. If an endpoint you expect to be there is not in that list, then it has not connected properly and more general Tivoli Monitoring troubleshooting needs to be followed.

### Tivoli Monitoring Endpoint Engine running but data is not being received or Resource Models are not running properly

It is possible that an endpoint can be contacted but data cannot be retrieved, or there is no data to be retrieved. Check if the endpoint has valid data to be retrieved, you can run the command

```
wdmlseng -e endpointName -verbose
```

This command will print out the Resource Models running on this endpoint, the status of those resource models, and the health of those Resource Models. If data is not there that you expect to be there, more general Tivoli Monitoring troubleshooting needs to be followed.

## Starting and Stopping WebSphere Application

There are several advanced troubleshooting steps possible for investigating problems. Some of these only really mean something to Tivoli Monitoring support, but they are listed here for referenced and for help when working with Tivoli Monitoring support personnel. Although you should normally never have to change the status of the WebSphere Application Server, it can be started and stopped manually.

### In UNIX:

To start: Export a display for the logged in users screen, such as  
`> export DISPLAY=machineName:0.0`

Run the script file `WHC_INSTALL_DIR/bin/startServer.sh`

To stop: Run the script file `WHC_INSTALL_DIR/bin/stopServer.sh`

### In Windows:

To start: Start the Windows Service WebSphere Application Server.

To stop: Stop the Windows Service WebSphere Application Server

or

To start: Run the batch file `WHC_INSTALL_DIR/bin/startServer.bat`

To stop: Run the batch file `WHC_INSTALL_DIR/bin/stopServer.bat`

## Starting and Stopping HTTPServer

Although you should normally never have to change the status of the Apache HTTPServer, it can be started and stopped manually.

### In UNIX:

To start: Run the file `APACHE_INSTALL_DIR/bin/apachectl start`

To stop: Run the file `APACHE_INSTALL_DIR/bin/apachectl stop`

### In Windows:

To start: Start the Windows Service "IBM HTTP Server"

To stop: Stop the Windows Service "IBM HTTP Server"

## Serviceability Tasks

Tivoli Monitoring provides three serviceability tasks:

- `DMCollectEpLog`
- `DMCollectMnLog`
- `DMCollectEpEnv`

To run the serviceability tasks:

1. From the Tivoli Desktop select **Desktop** → **TMR Connections** → **Top Level Policy Regions**. The Top Level Policy Regions dialog opens.
2. Double-click the **TivoliDefaultMw2kRegion** icon. The Policy Region dialog for that policy region opens.
3. Double-click the Tivoli Monitoring Tasks icon. The Task Library dialog for that library opens.
4. Run one of the following three serviceability tasks.

**DMCollectEpLog**

This task collects in a tar file created at the endpoint, all the endpoint logs and information about the size and dates of the binaries as well as the current and universal time the logs were created.

For UNIX/Linux platforms, the following files are collected:

```
$LCF_DATDIR/lcfd.log
$LCF_DATDIR/lcfd.bk
$LCF_DATDIR/last.cfg
$LCF_DATDIR/LCFNEW/Tmw2k/Unix/Tmx4j1.log
$LCF_DATDIR/LCFNEW/Tmw2k/Unix/Tmx4j2.log
$LCF_DATDIR/LCFNEW/AMW/logs/trace_xxxxx.log
$LCF_DATDIR/LCFNEW/AMW/logs/msg_xxxxx.log
$LCF_DATDIR/LCFNEW/Tmw2k/Unix/data/dmxout.log (this is a file
that traces errors at Java engine startup)
```

For Windows platforms, the following files are collected:

```
$LCF_DATDIR/lcfd.log
$LCF_DATDIR/lcfd.bk
$LCF_DATDIR/last.cfg
$LCF_DATDIR/LCFNEW/Tmw2k/Unix/Tmw2k.log
```

The task accepts the name of the tar file as argument.

Any core dumps from the engine are not included in the tar file to avoid impacting the task performance. Core dumps can be found in the directory:

```
$LCF_DATDIR\LCFNEW\Tmw2k\Unix
```

**DMCollectMnLog**

This task collects in a tar file created at the managed node in the \$DBDIR directory, all the managed node logs and traces, including event logs for Windows platforms.

For UNIX/Linux platforms, the following files are collected:

```
$DBDIR/oservlog
$DBDIR/gatelog
/tmp/traces/trace_tnmt_gtw_engn.log
$DBDIR/AMW/logs/trace_xxxx.log
$DBDIR/<odstat output>
$DBDIR/<wtrace -jHk $DBDIR output>
```

For Windows platforms, the following files are collected:

```
$DBDIR/oservlog
$DBDIR/gatelog
$DBDIR/tmp/traces/trace_tnmt_gtw_engn.log
$DBDIR/AMW/logs/trace_xxxx.log
$DBDIR/<odstat output>
$DBDIR/<wtrace -jHk $DBDIR output>
```

The task accepts the name of the tar file as argument.

**DMCollectEpEnv**

This task collects information about the environment at the endpoint. The data collected is written to a file using the Execute Task dialog (Save to File option).

For UNIX/Linux platforms, the following information is collected:



- platform (AIX, HP-UX, ...)
- operating system version
- disk space statistics and file system installation at the endpoint
- memory statistics (available and used)
- environment variable settings
- list of system patches installed

For Windows platforms, the following information is collected:

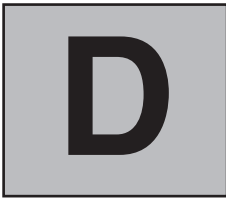
Output from the **winmsd** command.

For Windows 2000 the report is created in  
\$LCF\_DATDIR/winmsdreport.txt.

For Windows NT the report is created in  
\$LCF\_DATDIR/<hostname>.txt

This task does not accept arguments.





# Accessibility

---

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in Tivoli Monitoring enable users to:

- Use assistive technologies such as screen-reader software and a digital speech synthesizer to hear what is displayed on the screen
- Operate specific or equivalent features using only the keyboard
- Magnify what is displayed on the screen

In addition, the product documentation has been modified to include features to aid accessibility:

- All documentation available in both HTML and convertible PDF formats to give the maximum opportunity for users to apply screen-reader software.
- All images provided with alternative text so that users of the documentation with vision impairments can understand the contents of the images.

## Using Assistive Technologies

Assistive technology products such as screen-readers, function with both the text-based and graphical user interfaces found in Tivoli Monitoring. Consult the assistive technology product documentation for specific information about using it to access command line or graphical interfaces.

Additional accessibility features might be included as part of the user interface of a particular Tivoli Monitoring component. Check with the individual component's documentation for any additional information about accessibility.

## Magnifying What is Displayed on the Screen

In all components of Tivoli Monitoring other than the Web Health Console, users can magnify the screens used by the product's user interfaces using facilities provided by the operating systems on which the product is run. For example, in a Windows environment you can change the screen settings to a lower resolution to enlarge the font sizes of the text on the screen. Information about these facilities is provided in the relevant operating system documentation.

## Documentation in Accessible Formats

All user documentation is provided in HTML format, which can be read directly by assistive tools such as screen readers, or in *convertible* PDF format. Convertible PDF files are those that can be converted from PDF to HTML by the Adobe PDF to HTML converter. For information about converting PDF documents to HTML, refer to the Adobe book *Optimizing Adobe PDF Files for Accessibility*

## Using Alternative Text

All documentation images are provided with an alternative text that can be read by assistive tools such as screen readers.



# Use of Rhino: JavaScript for Java

---

Rhino is an open-source implementation of JavaScript written entirely in Java. It is typically embedded into Java applications to provide scripting to end users. Its use in Tivoli Monitoring is governed by *The Mozilla Public License Version 1.1*, given below, as amended by *The Netscape Public License Version 1.1*, given on page 211.

## The Mozilla Public License Version 1.1

### 1. Definitions.

- 1.0.1. “Commercial Use” means distribution or otherwise making the Covered Code available to a third party.
- 1.1. “Contributor” means each entity that creates or contributes to the creation of Modifications.
- 1.2. “Contributor Version” means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.
- 1.3. “Covered Code” means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.
- 1.4. “Electronic Distribution Mechanism” means a mechanism generally accepted in the software development community for the electronic transfer of data.
- 1.5. “Executable” means Covered Code in any form other than Source Code.
- 1.6. “Initial Developer” means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.
- 1.7. “Larger Work” means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.
- 1.8. “License” means this document.
- 1.8.1. “Licensable” means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
- 1.9. “Modifications” means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:
  - A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.
  - B. Any new file that contains any part of the Original Code or previous Modifications.
- 1.10. “Original Code” means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.
- 1.10.1. “Patent Claims” means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.
- 1.11. “Source Code” means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor’s choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. “You” (or “Your”) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, “You” includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

## **2. Source Code License.**

### *2.1. The Initial Developer Grant.*

The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

- (a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and
- (b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).
- (c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.
- (d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

### *2.2. Contributor Grant.*

Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

- (a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and
- (b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
- (c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.
- (d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

## **3. Distribution Obligations.**

### *3.1. Application of License.*

The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients’ rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

### 3.2. Availability of Source Code.

Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

### 3.3. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

### 3.4. Intellectual Property Matters

#### (a) Third Party Claims.

If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

#### (b) Contributor APIs.

If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

#### (c) Representations.

Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

### 3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

### *3.6. Distribution of Executable Versions.*

You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

### *3.7. Larger Works.*

You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

## **4. Inability to Comply Due to Statute or Regulation.**

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

## **5. Application of this License.**

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

## **6. Versions of the License.**

### *6.1. New Versions.*

Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

### *6.2. Effect of New Versions.*

Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

### *6.3. Derivative Works.*

If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)



## 7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN “AS IS” BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

## 8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as “Participant”) alleging that:

(a) such Participant’s Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant’s Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant’s Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and releasers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

## 9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

**10. U.S. GOVERNMENT END USERS.**

The Covered Code is a “commercial item”, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of “commercial computer software” and “commercial computer software documentation”, as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

**11. MISCELLANEOUS.**

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

**12. RESPONSIBILITY FOR CLAIMS.**

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

**13. MULTIPLE-LICENSED CODE.**

Initial Developer may designate portions of the Covered Code as “Multiple-Licensed”. “Multiple-Licensed” means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

**EXHIBIT A -Mozilla Public License.**

“The contents of this file are subject to the Mozilla Public License Version 1.1 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/MPL/>

Software distributed under the License is distributed on an “AS IS” basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is \_\_\_\_\_.

The Initial Developer of the Original Code is \_\_\_\_\_. Portions created by \_\_\_\_\_ are Copyright (C) \_\_\_\_\_. All Rights Reserved.

Contributor(s): \_\_\_\_\_.

Alternatively, the contents of this file may be used under the terms of the \_\_\_\_\_ license (the “[ ] License”), in which case the provisions of [ ] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [ ] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [ ] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [ ] License.”

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

# The Netscape Public License Version 1.1

## AMENDMENTS

The Netscape Public License Version 1.1 (“NPL”) consists of the Mozilla Public License Version 1.1 with the following Amendments, including Exhibit A-Netscape Public License. Files identified with “Exhibit A-Netscape Public License” are governed by the Netscape Public License Version 1.1.

Additional Terms applicable to the Netscape Public License.

### I. Effect.

These additional terms described in this Netscape Public License -- Amendments shall apply to the Mozilla Communicator client code and to all Covered Code under this License.

### II. “Netscape’s Branded Code”

Means Covered Code that Netscape distributes and/or permits others to distribute under one or more trademark(s) which are controlled by Netscape but which are not licensed for use under this License.

### III. Netscape and logo.

This License does not grant any rights to use the trademarks “Netscape”, the “Netscape N and horizon” logo or the “Netscape lighthouse” logo, “Netcenter”, “Gecko”, “Java” or “JavaScript”, “Smart Browsing” even if such marks are included in the Original Code or Modifications.

### IV. Inability to Comply Due to Contractual Obligation.

Prior to licensing the Original Code under this License, Netscape has licensed third party code for use in Netscape’s Branded Code. To the extent that Netscape is limited contractually from making such third party code available under this License, Netscape may choose to reintegrate such code into Covered Code without being required to distribute such code in Source Code form, even if such code would otherwise be considered “Modifications” under this License.

### V. Use of Modifications and Covered Code by Initial Developer.

#### V.1. In General.

The obligations of Section 3 apply to Netscape, except to the extent specified in this Amendment, Section V.2 and V.3.

#### V.2. Other Products.

Netscape may include Covered Code in products other than the Netscape’s Branded Code which are released by Netscape during the two (2) years following the release date of the Original Code, without such additional products becoming subject to the terms of this License, and may license such additional products on different terms from those contained in this License.

#### V.3. Alternative Licensing.

Netscape may license the Source Code of Netscape’s Branded Code, including Modifications incorporated therein, without such Netscape Branded Code becoming subject to the terms of this License, and may license such Netscape Branded Code on different terms from those contained in this License.

### VI. Litigation.

Notwithstanding the limitations of Section 11 above, the provisions regarding litigation in Section 11(a), (b) and (c) of the License shall apply to all disputes relating to this License.

**EXHIBIT A-Netscape Public License.**

“The contents of this file are subject to the Netscape Public License Version 1.1 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/NPL/>

Software distributed under the License is distributed on an “AS IS” basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is Mozilla Communicator client code, released March 31, 1998.

The Initial Developer of the Original Code is Netscape Communications Corporation. Portions created by Netscape are Copyright (C) 1998-1999 Netscape Communications Corporation. All Rights Reserved.

Contributor(s): \_\_\_\_\_.

Alternatively, the contents of this file may be used under the terms of the \_\_\_\_\_ license (the “[\_\_\_\_\_] License”), in which case the provisions of [\_\_\_\_\_] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [\_\_\_\_\_] License and not to allow others to use your version of this file under the NPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [\_\_\_\_\_] License. If you do not delete the provisions above, a recipient may use your version of this file under either the NPL or the [\_\_\_\_\_] License.”



## Environment Variables

This chapter documents the environment variables used by Tivoli Monitoring.

The following table shows the list of environment variables set by triggering events (implemented through tasks, program responses, and Common Information Model (CIM) methods).

*Table 16. Environment variables set by triggering events.*

Variable	Definition
<b>EVENT_NAME</b>	The Aggregated Event Name
<b>EVENT_MESSAGE</b>	The event message.
<b>EVENT_SEVERITY</b>	The event severity.
<b>TMW_EVENT</b>	Same as <b>EVENT_NAME</b> . It is deprecated and remains for backward compatibility only.
<b>TMW_PROFILE</b>	Same as <b>PROFILE</b> . It is deprecated and remains for backward compatibility only.
<b>\$(EVENT_NAME)_Propertyxxx</b>	Local event attributes.

The following table shows the list of environment variables that can be set through tasks, program responses, CIM methods, custom script or monitoring collections started by resource model scripts.

*Table 17. More environment variables.*

Variable	Definition
<b>ADMIN</b>	The name of the administrator responsible for the most recent distribution of the profile containing the monitor.
<b>ENDPOINT</b>	The name of the endpoint object through which the profile containing the monitor was distributed.
<b>ENDPOINT CLASS</b>	The Tivoli class (or "type") name of the endpoint object.
<b>ENDPOINT OID</b>	The Tivoli object ID of the endpoint object.
<b>HOST</b>	This is the Hostname taken with <code>GetHostByName()</code> .
<b>OPERATOR</b>	The identity through which the administrator gained access to the Tivoli environment.
<b>PROFILE</b>	Full name of the profile.
<b>RESOURCE_MODEL</b>	Resource Model internal name.

---

*Table 17. More environment variables. (continued)*

Variable	Definition
<b>RESOURCE_MODEL_DESC</b>	Resource Model descriptive name.
<b>PROFILEOID</b>	The Tivoli object ID of the profile that contains the model.



# Migration Considerations

---

This appendix provides guidelines to:

- Tivoli Distributed Monitoring (Classic Edition) users who are migrating to Tivoli Monitoring 5.1
- Tivoli Web Component Manager users who are migrating to Tivoli Monitoring 5.1

It describes some considerations you should make when planning your migration to make your migration as easy and efficient as possible.

## Migrating from Tivoli Distributed Monitoring (Classic Edition) to Tivoli Monitoring

Tivoli Monitoring 5.1 allows you to monitor availability and performance status of resources on your systems to identify bottlenecks and potential resource problems.

Users of Tivoli Distributed Monitoring (Classic Edition) can easily migrate to using Tivoli Monitoring 5.1.

### Overview

Tivoli Monitoring 5.1 applies monitoring in a different way from Tivoli Distributed Monitoring (Classic Edition). Whereas the Tivoli Distributed Monitoring (Classic Edition) is based on the concept of monitoring capability collections and monitors, Tivoli Monitoring 5.1 applies preconfigured, automated best practices to the automated monitoring of essential system and application resources. Basically, Tivoli Distributed Monitoring (Classic Edition) provides the means for retrieving data through scripts or commands, verifies that the retrieved values do not exceed given thresholds, and finally, based on the user's customization, triggers some response (Tivoli Enterprise Console<sup>®</sup> events, e-mail, notices, and so on). Tivoli Distributed Monitoring (Classic Edition) leaves the correlation and the problem's root cause analysis to the user.

Tivoli Monitoring 5.1 is based on the concept of resource models, implemented through best practice scripts, or reference models, and the definition in Common Information Model (CIM) in terms of monitored resources. See the Web site:

[http://www.dmtf.org/standards/cim\\_spec\\_v22/](http://www.dmtf.org/standards/cim_spec_v22/)

for more details. Moreover, Tivoli Monitoring 5.1 provides a set of resource models designed to detect run time bottlenecks and other potential problems and to automatically recover from critical situations, eliminating the need for system administrators to manually scan through extensive performance data.

Tivoli Monitoring 5.1 may use processes that form part of the endpoints' operating systems to obtain resource data. On Windows systems it uses the Windows Management Instrumentation (WMI), which is Microsoft®'s implementation of CIM. See the Web site:

[http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/aboutwmi\\_11pl.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/aboutwmi_11pl.asp)

WMI allows applications to retrieve information about the current status of a system. On UNIX® and Linux platforms the information collection agent is incorporated in the product based on CIM specifications.

The IBM Tivoli Monitoring Version 5.1 product bundle provides both Tivoli Monitoring 5.1 and Tivoli Distributed Monitoring (Classic Edition) 3.7, so that users can use both and plan their migration to Tivoli Monitoring 5.1 in the smoothest possible way. Users should take the following two aspects into consideration:

### **Coexistence**

The two products can coexist as they have two different implementations at the server, gateways, and endpoints. The installation of Tivoli Monitoring 5.1 does not change the Tivoli Distributed Monitoring (Classic Edition) environment or configuration. Even when working in Compatibility Mode, see "Compatibility Mode" on page 224, the two products can coexist and run entirely independently.

### **Migration**

To make the migration from Tivoli Distributed Monitoring (Classic Edition) to Tivoli Monitoring 5.1 as easy as possible and to save the user's investment on custom scripts and home-developed monitoring collections, Tivoli Monitoring 5.1 runs in "Compatibility Mode", the new working mode that allows Tivoli Monitoring 5.1 users to use Tivoli Distributed Monitoring (Classic Edition) monitor collections and custom scripts within resource models.

Due to the different nature of the two versions of Tivoli Distributed Monitoring, the migration can only be semi-automated. In addition to the Compatibility Mode which gives users the possibility of using their scripts or monitors by leveraging Tivoli Monitoring 5.1 features, Tivoli Monitoring 5.1 also provides a migration helper script, `dmae_sentryprofile_analyser.sh`, which provides information on how and where the metric monitored by the current Tivoli Distributed Monitoring (Classic Edition) profiles can be replaced using Tivoli Monitoring 5.1. The script will be able to automatically create Tivoli Monitoring 5.1 profiles containing resource models that try to address the same problems as the Tivoli Distributed Monitoring (Classic Edition) monitors. More details about this script can be found later in the chapter.

To take full advantage of the Tivoli Monitoring 5.1 key features such as instance autodiscovery, metric correlation, aggregation, data logging, and so on, may require some Visual Basic or JavaScript coding skills that might not be immediately available, so to help to introduce Tivoli Distributed Monitoring (Classic Edition) users to the Tivoli Monitoring 5.1 monitoring approach Tivoli Monitoring 5.1 provides a wizard-driven process within the workbench to create resource models and automatically generate all the required code. The wizard allows the user to follow the simple monitoring paradigm in Tivoli Distributed Monitoring (Classic Edition) by leveraging the new functionality offered by Tivoli Monitoring 5.1. The wizard approach relieves the user from writing additional code by generating code that applies simple Tivoli Distributed Monitoring (Classic Edition) logic:

```
if (metric value - comparison - threshold) then response action
```



Using the wizard together with the compatibility mode allows users to replicate the same monitoring they already have in place with Tivoli Distributed Monitoring (Classic Edition). More details of the wizard can be found in *IBM Tivoli Monitoring: Workbench User's Guide*.

## Guidelines for Migrating to Tivoli Monitoring 5.1

To move monitoring solutions from Tivoli Distributed Monitoring (Classic Edition) to Tivoli Monitoring 5.1 implementation, it is important to better understand the key features that Tivoli Monitoring 5.1 offers at the different phases of its use.

### Design Time

Tivoli Monitoring 5.1 provides the workbench, an integrated development environment (IDE) to design, create, test, and debug resource models. See the *IBM Tivoli Monitoring: Workbench User's Guide* for more information. Basically, new Tivoli Monitoring 5.1 users should consider the workbench as the place where they can build their own monitoring best practice, where they can modify the current out-of-the-box resource model best practices, and where they can customize and configure the resource models in a more detailed way. In Tivoli Monitoring 5.1 users should see the workbench as an additional and more complete GUI where they can customize resource models in addition to the TME<sup>®</sup>-based GUI. The importance of the workbench is mainly related to the possibility of creating and debugging new resource models.

### Data sources

Tivoli Monitoring 5.1 is able to gather data not only from CIM data sources but also from Tivoli Distributed Monitoring (Classic Edition) monitors and custom scripts (Compatibility Mode). In this way Tivoli Monitoring 5.1 enhances the set of data sources with those already supported by Tivoli Distributed Monitoring (Classic Edition). However, it is still preferable to work in native mode, where the monitored data is collected from the underlying CIM implementation. While on UNIX and Linux platforms the CIMOM implementation is embedded on the engine, on Windows platforms the engine is based on WMI implementation. See the Web site:

[http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/aboutwmi\\_11pl.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/aboutwmi_11pl.asp)

Microsoft operating systems and Backoffice applications are delivered with their own WMI providers so that the monitoring of such resources can be implemented very quickly.

### Aggregation

Differently from Tivoli Distributed Monitoring (Classic Edition), in Tivoli Monitoring 5.1 all events can be consolidated locally at the endpoint before flowing to the upper layers such as Tivoli Enterprise Console or Tivoli Business Systems Manager. This feature is very important from a scalability point of view. Also, the best practice can leverage this functionality to make the troubleshooting analysis smarter and more realistic (for example, whether a process exceeds its CPU usage just once or many times). New Tivoli Monitoring 5.1 users should always take this feature into account when generating an event.

### Clearing

As the starting point for Tivoli Monitoring 5.1 is providing information for problems that are critical to operating system or application resources, Tivoli Monitoring 5.1 notifies users only when the problem arises through Tivoli Enterprise Console or Tivoli Business Systems Manager, and only when the problem has been solved, does it send a warning with a Clearing Event.

### **Correlation**

The most important feature is definitely having the possibility to write best practices to troubleshoot problems before they happen. That is, the root cause analysis can be performed by the resource model script that, using the Tivoli Monitoring 5.1 engine API, defines the data to be collected and, once it has been collected by the engine, retrieves all the metric values and correlates them to establish the final cause of arising problems. The correlation is then achieved by following a programming model that, in simple cases can be implemented by using the wizard-driven process. Metric values collected through the compatibility mode can of course be correlated too.

### **Multimetric**

When starting to write new resource models, new Tivoli Monitoring 5.1 users should consider that in each resource model it is possible to collect an undefined number of resources and for each resource an undefined number of metrics. Moreover it gives the possibility to have autodiscovery for all the instances of a resource.

**Note:** Implementing a resource model that looks just at one metric of a resource may be expensive as Tivoli Monitoring 5.1 creates a new thread and a set of objects for each script. Better results can be obtained by designing the resource model to address a specific set of problems (for example, memory and CPU bottlenecks).

### **Automation**

When building new resource models, or while modifying existing ones, Tivoli Monitoring 5.1 users can set recovery actions called built-in actions to be triggered whenever an indication is consolidated. The built-in actions are the running of CIM methods (see *IBM Tivoli Monitoring: Workbench User's Guide*) against a CIM class or CIM class instance.

### **Logging**

Tivoli Monitoring 5.1 allows users to log data to a local database. Data stored in such a database can be retrieved and viewed through the Web Health Console. Data retrieved through the compatibility mode can also be logged.

## **Deployment Time**

Tivoli Monitoring 5.1 resource models are configured and deployed through Tivoli Management Framework just as they are in Tivoli Distributed Monitoring (Classic Edition). Instead of using a Monitoring Collection and monitors, Tivoli Monitoring 5.1 users use resource models, and instead of creating a SentryProfile create a Tivoli Monitoring 5.1profile, *Tmw2kProfile*. As the configuration and deployment procedures of Tivoli Monitoring 5.1 follow the paradigm followed by all profile-based Tivoli monitoring applications, we will describe only the things that differ from Tivoli Distributed Monitoring (Classic Edition) usage.

### **Resource model installation**

Once Tivoli Monitoring 5.1 users have built their own resource model they can build the package with the workbench (see *IBM Tivoli Monitoring: Workbench User's Guide*) and then install it on the Tivoli management region, using the **wdmrm** command.

### **Tivoli Enterprise Console and Tivoli Business Systems Manager Customization**

Tivoli Monitoring 5.1 allows users to set a unique destination Tivoli Enterprise Console server per profile. This means that all resource models belonging to the

same profile will send events to the Tivoli Enterprise Console server. The target Tivoli Enterprise Console can be specified from the properties dialog.

### **Task configuration**

Tivoli tasks can be run as recovery actions. Differently from Tivoli Distributed Monitoring (Classic Edition) they can be triggered only when an indication has been consolidated.

**Note:** In the Task environment there are the Event properties in the form of environment variables.

### **Profile distribution**

Tivoli Monitoring 5.1 supports endpoints only. This means that there is no managed node Tivoli Monitoring 5.1 monitoring engine, and implies some rework to be done when replacing Tivoli Distributed Monitoring (Classic Edition) profiles with Tivoli Monitoring 5.1 profiles because Tivoli Distributed Monitoring (Advanced Edition) cannot be distributed on managed nodes.

## **Run Time**

Once a Tivoli Monitoring 5.1 profile and its resource models have been distributed to endpoints, the following considerations must be made for Tivoli Monitoring 5.1:

### **Multithread**

Tivoli Monitoring 5.1 operates in a multithread environment. This means that each script runs in a separate thread and no processes are generated except those needed to run in compatibility mode. Tivoli Monitoring 5.1 users should take this into account when creating a resource model that looks at only one metric.

### **Web Health Console**

Tivoli Monitoring 5.1 provides a web-based GUI called Web Health Console to browse the Tivoli Monitoring 5.1 monitored endpoints. Details of the Web Health Console features can be found in Chapter 7, “Introduction to the Web Health Console” on page 83 and Chapter 8, “Using the Web Health Console” on page 93.

### **Tivoli Enterprise Console rule for clearing event**

Tivoli Monitoring 5.1 provides a Tivoli Enterprise Console rule to automatically close events for which a Clearing event is generated.

## **Migration Process**

As described above, Tivoli Monitoring 5.1 provides a set of facilities to help and facilitate Tivoli Distributed Monitoring (Classic Edition) users to migrate their monitoring solutions into a Tivoli Monitoring 5.1 environment. There is no tool to completely automate the migration process because, as described above, the two products are really different.

So what should new Tivoli Monitoring 5.1 users do? How can they migrate to the new monitoring infrastructure? Here is a description of the facilities that, used together, will help users in their migration task. Tivoli Monitoring 5.1 users who do not plan to use the Sentry Profile analyzer script should still read the following section because it describes the thought process to be followed when planning to migrate.

### **Sentry Profile Analyzer**

The first facility is the Sentry Profile analyzer script that analyses the contents of all the Sentry Profiles and, based on a mapping table provided with Tivoli Monitoring 5.1, produces a report suggesting how the monitors can be replaced with resource models or how new resource models can be created to collect the same data.

### Installation Path

The installation of Tivoli Monitoring 5.1 on TME stores the script named `dmae_sentryanalyser.sh` in the `$BINDIR/TME/Tmw2k/migration_helper` directory. The mapping table file named `monitors_rm_table` is also stored in the same directory.

### Mapping Table Content

The mapping table defines:

- If the metric values returned by each monitor provided by Tivoli Distributed Monitoring (Classic Edition) are collected by a Tivoli Monitoring 5.1 resource model, for example, the monitor `AvailBytes` of the `NT_Memory` monitoring collection is collected by the `Memory` resource model.

- If a metric is defined in a Tivoli Distributed Monitoring (Classic Edition) CIM class: that is, the corresponding CIM class.

**Note:** many metrics usually belong to the same class, for example, the monitor `AvailBytes` of `NT_Memory` can be found as a property of the CIM class `TMW_Memory`, and the monitor `Committed Bytes` can be found as a property of the same `TMW_Memory` class.

- For the metrics defined in a CIM class, could there be a Tivoli Monitoring 5.1 MOF file where they are defined. See the Web site:

<http://www.dmtf.org/education/cimtutorial/extend/spec.php#MOFLanguage> for more details. For example, the class `TMW_Memory` is defined in the file `TMW_Resources10.mof`.

**Note:** the `TMW_Resources10.mof` file is automatically installed on WMI with the endpoint engine when the first push is performed.

All the other mof files are only installed in the related CIM implementation when the resource model containing these files as dependencies gets downloaded. This means that creating a resource model that uses a resource defined in `TMW_Resource10.mof` there is no need to add that file to the dependency, while for all other cases the user must add the mof file to the dependency of the resource model.

- If a metric is made available by any CIM provider, independently of whether or not it is a property of a CIM class, for example, the `TransitionFaults` metric is made available by the `PerfProv` WMI provider, but there is not any CIM class already available on Window NT that indirectly uses that provider to collect that counter.

At the Web site:

[http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/us\\_perfcount](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/us_perfcount)

users can find more information on how to use performance data providers. At the Web site:

[http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/us\\_prov\\_33mx.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/us_prov_33mx.asp)

users can find more information on all other WMI providers. All providers used on Windows platforms are provided by default with WMI core, so there is no need to install or download these on the endpoints.

- Could there be a case such as `TMW_NetworkPortInfo` whose provider has been developed by Tivoli and then in order to be used on an endpoint must be added to the dependencies of a resource model. All providers used for UNIX and Linux platforms, as there is no native CIM implementation, have been developed by Tivoli and their entire implementation must be added to the resource model dependencies. The implementations of all UNIX and Linux providers consist of:
  - A tar file containing a set of Java class files
  - For each supported platform, a native shared library responsible for gathering data.

Based on the above considerations, in order to reuse a UNIX or Linux CIM class inside a resource model, users must add to the dependencies the related mof file and the tar file (they are common to all platforms) and the related shared library for each platform to be supported.

The Sentry Profile analyzer script must be run from the Tivoli environment on any Server or Managed Node. Based on the content of the mapping table described above, the Sentry Profile analyzer analyzes all Sentry Profiles present in the Tivoli management region and produces a report that suggests the way to proceed in the migration process.

Optionally, using `-p` option can also create Tivoli Monitoring 5.1 profiles with the resource models covering as much as possible the resources monitored by the Tivoli Distributed Monitoring (Classic Edition) monitors.

### The Analyzer Script Process

The script proceeds with the following paradigm.

- It verifies whether the metrics of the monitors set in a `SentryProfile` are collected by any Tivoli Monitoring 5.1. If they are, the script generates a section in the report describing which monitor can be replaced with which resource model. Optionally a `Tmw2kProfile` with the same name as that containing the monitor will be created. Other resource models can also be added to that profile. Note that the configuration of a monitor in terms of arguments, responses, schedules, and so on, cannot be replicated because of the different product object models (see the section above). Usually the arguments in Tivoli Distributed Monitoring (Classic Edition) monitors are used to identify a specific resource instance. This is in conflict with the format of the out-of-the-box Tivoli Monitoring 5.1 resource models which try to autodiscover the failing instances at run time. Current resource models in fact do not accept resource instances as input. Nevertheless they can easily be modified with the workbench by using the parameters that only look and perform their analyses on specific instances. Also, other customization related to responses and schedules cannot be automatically recreated within a resource model but instead existing resource models can be modified in order to simulate the Tivoli Distributed Monitoring (Classic Edition) configuration.
- If no resource models that collect a specific metric are found in the mapping table then the script tries to identify whether a CIM class exists which provides that value. If one does then a section is added to the report. This only happens for Windows Platforms because there are several providers and CIM classes already made available by Microsoft. This information is really important

because users, using the workbench (with or without the wizard), can easily take advantage of these classes to create a more sophisticated resource model running in native mode.

- Sometimes there might be the case where the script finds a metric that is not implemented by any CIM class, but that is provided by a provider such as PerfProv. Microsoft provides with WMI core the WMI performance counter provider that can be used to define CIM classes representing the objects and their counters as they appear in the Windows performance monitor. Instructions on how to use the provider can be found at the Web site:

[http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/aboutwmi\\_11pl.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/aboutwmi_11pl.asp)

Good examples can be seen in the `TMW_Resources10.mof` that is stored in `$BINDIR/../../lcf_bundle40/Tmw2k/Mof`. Most Tivoli Monitoring 5.1 Windows resource models are based on CIM classes using the PerfProv WMI provider. Here is an example:

On Windows platforms the provider for instances or properties of a given class is specified through the "Provider" CIM qualifier. For example the provider for the following class `NTProcesses` is the "PerfProv".

```
[dynamic, provider("PerfProv"), ClassContext("local|Process")]
class NTProcesses
{
    [key]
    String Process;
    [PropertyContext("ID Process")]
    uint32 ID;
    [PropertyContext("Working Set")]
    uint32 WorkingSet;
};
```

The script will add a section to the report suggesting that the user considers creating a new resource model based on new CIM classes that can be created using the indicated provider.

- If the monitor examined cannot be remapped in any CIM class without the creation of a new provider (see the Web site [http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/aboutwmi\\_11pl.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/aboutwmi_11pl.asp) for how to create WMI providers), the script suggests using the Compatibility Mode in conjunction with the Wizard-driven process and choosing the Tivoli Distributed Monitoring Classic Monitoring Collection as the data source. Note that in order to optimize the use of monitors users should consider using more than one monitor of the same Monitoring collection inside the same resource model wherever possible.
- If the monitor is a custom script (string or numeric) and cannot be remapped in any CIM class without the creation of a new provider (see the Web site [http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/aboutwmi\\_11pl.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/aboutwmi_11pl.asp) for how to create WMI providers), the script suggests using the Compatibility Mode in conjunction with the Wizard-driven process and choosing **Custom Script** as the data source.
- If the monitor is an async monitor then users should ask Tivoli services to replace that functionality as it has not yet been implemented.

**Examples**

Here is an example of report generated by the Sentry Profile analyzer script:

```
# Tivoli Distributed Monitoring (Advanced Edition),
# Version 4.1 (C) Copyright IBM Corporation 2001.
# All rights reserved.
```

The metrics collected by the following monitors are collected by the by the following RESOURCES MODELS:

	COLLECTION	MONITOR	RESOURCE MODEL	CIM CLASSES(PROVIDER)
UnixProfile#MyRegion	Unix_Sentry	File size	DMXFile	DMXFile(DMXFileIlt)
UnixProfile#MyRegion	Unix_Sentry	File size	DMXSecurity	DMXFileSec(DMXFileSecIlt)
UnixProfile#MyRegion	Unix_Sentry	Space free	DMXFileSystem	DMXFileSystem(DMXFileSystemIlt)
WinNTProfile#MyRegion	NT_NetworkMonitor	Network utilizat	TMW_NetworkIntCard	TMW_NetworkSegment(PerfProv)
WinNTProfile#MyRegion	NT_LogicalDisk	Avg Disk sec/Trans	TMW_LogicalDisk	TMW_LogicalDisk(PerfProv)

The metrics collected by the following monitors can be found in the following CIM CLASSES:

COLLECTION	MONITOR	CIM CLASSES(PROVIDER)
------------	---------	-----------------------

Please consider to create Cim class to use the performance provider.  
 For more information visit:  
[www.http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmsdk/r\\_prov\\_2joy.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmsdk/r_prov_2joy.asp).

	COLLECTION	MONITOR	PROVIDER
DummyProfile#NewPolicy	NT_Cache	Async Pin Reads/sec	PerfProv
DummyProfile#NewPolicy	NT_Cache	Pin Reads/sec	PerfProv
DummyProfile#NewPolicy	NT_Cache	Sync Pin Reads/sec	PerfProv
DummyProfile#NewPolicy	NT_System	File Control Bytes/sec	PerfProv
WinNTProfile#MyRegion	NT_Cache	Async Pin Reads/sec	PerfProv
WinNTProfile#MyRegion	NT_Cache	Pin Reads/sec	PerfProv
WinNTProfile#MyRegion	NT_Cache	Sync Pin Reads/sec	PerfProv

The metrics collected by the following monitors are not collected by any resource model and can not be found in any CIM class.  
 Please consider to use WORKBENCH WIZARD choosing 'DM CLASSIC MONITORING COLLECTION' to import the monitor in a resource model

	COLLECTION	MONITOR
DummyProfile#NewPolicy	Unix_Sentry	daemonct
DummyProfile#NewPolicy	Unix_Sentry	diskusedpct
DummyProfile#NewPolicy	Unix_Sentry	diskused
OS400Profile#EMEA-region	OS/400 Job	SubType
OS400Profile#EMEA-region	OS/400 Object	ObjectOwner
UnixProfile#MyRegion	Unix_Sentry	daemonct

The metrics collected by the following monitors are not collected by any resource model and can not be found in any CIM class.  
 Please consider to use WORKBENCH WIZARD choosing 'CUSTOM SCRIPTS' to import

```
| the monitor in a resource model
|
|           COLLECTION           MONITOR
|
| =====
| Please consider to ask Tivoli services for replacing the following monitors:
|
|           COLLECTION           MONITOR
|
| UnixProfile#MyRegion           Unix_Sentry           sasync
```

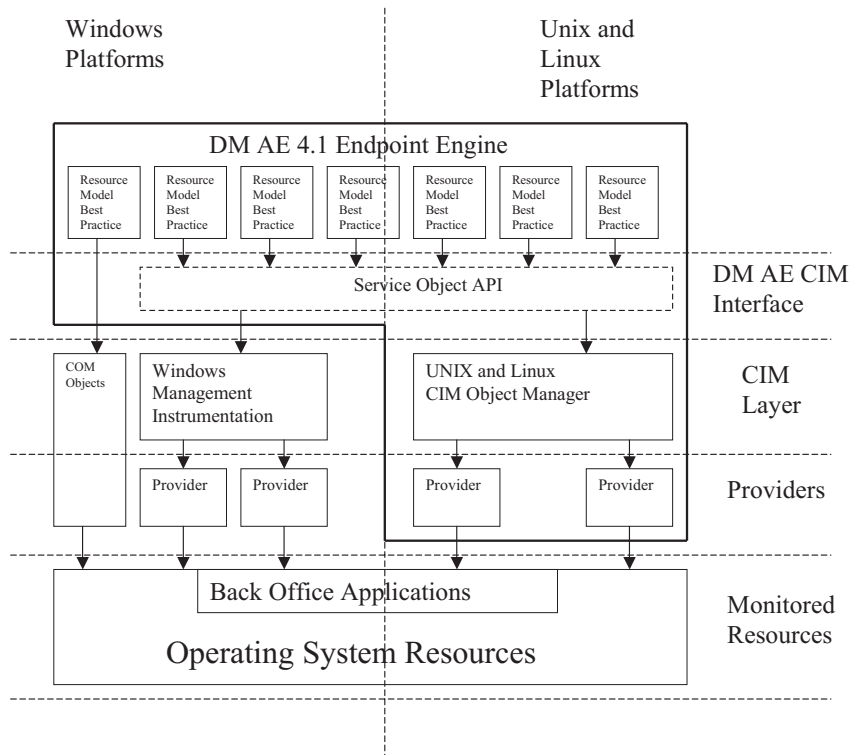
**Compatibility Mode**

The compatibility mode is a new working mode allowing Tivoli Monitoring 5.1 users to use Tivoli Distributed Monitoring (Classic Edition) monitors inside a Tivoli Monitoring 5.1 resource model. In this way, Tivoli Monitoring 5.1 can collect data to analyze, not only from the CIM data source, but also from Tivoli Distributed Monitoring (Classic Edition) monitoring sources (also known as probes). This means that Tivoli Distributed Monitoring (Classic Edition) users can recycle their existing customized Tivoli Distributed Monitoring (Classic Edition) monitoring collections into the new Tivoli Monitoring 5.1 resource models.

Tivoli Distributed Monitoring (Advanced Edition) 4.1 uses native mode to collect data, relying on CIM Object Manager implementation (WMI on Windows or an endpoint Tivoli Distributed Monitoring (Advanced Edition) 4.1 engine itself on UNIX and Linux), to collect data about system resources. The following picture shows the endpoint Tivoli Distributed Monitoring (Advanced Edition) 4.1 engine architecture and how the product uses CIM Object Manger implementations in the Tivoli Distributed Monitoring 4.1 release, that is, before applying the 4.1-DMA0001 patch. The Tivoli Distributed Monitoring (Advanced Edition) 4.1 engine works only in "native mode" and resource model best practice scripts interact with the underlying CIMOM implementation through a set of API. The CIMOM is responsible for loading the providers that, in turn, get performance and availability data from system and application resources. Windows resource models may interact with COM objects directly from the scripts. In this case the CIM layer is not used.

The following picture illustrates the Tivoli Distributed Monitoring (Advanced Edition) 4.1 native mode:

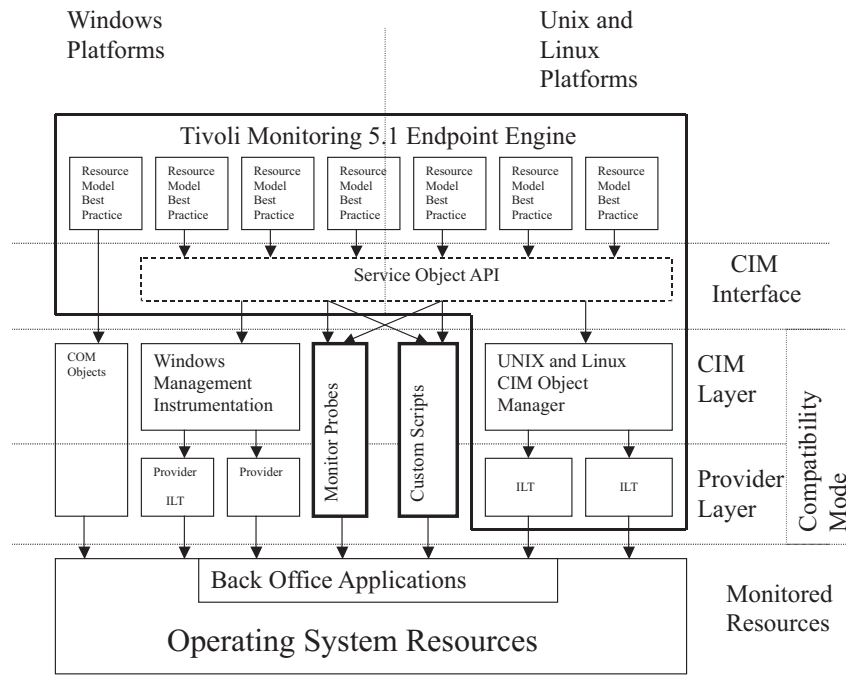




The compatibility mode can be used in addition to the native mode, so it is possible to have resource models that use both these technologies in their implementation. The compatibility mode provides an easy way to import Tivoli Distributed Monitoring (Classic Edition) probe implementation into a resource model without the user having to write any additional code, by supplying the workbench with a new wizard that imports Tivoli Distributed Monitoring (Classic Edition) monitoring sources. The Wizard process is described in *IBM Tivoli Monitoring: Workbench User's Guide*.

The following picture shows the endpoint Tivoli Monitoring 5.1 engine architecture. It can be noted the Service Object API now enables the resource model scripts to use monitoring collection probes and custom scripts and CIM in any combination.

**Note:** Asynchronous monitors are not supported.



### Wizard Process

The workbench provides a wizard, that is a GUI- driven process, to create new resource models. Basically, the wizard, starting from a selected CIM class taken from the WMI repository, from a monitoring collection, or from a custom script, displays a sequence of dialogs with default values already filled in to drive the user to create simple resource models. At the end of the wizard process all the needed Visual Basic or JavaScript code will be automatically generated. This code will implement a monitoring logic very similar to that used by Tivoli Distributed Monitoring (Classic Edition), that is:

```
if (metric value - comparison - threshold) then indication
```

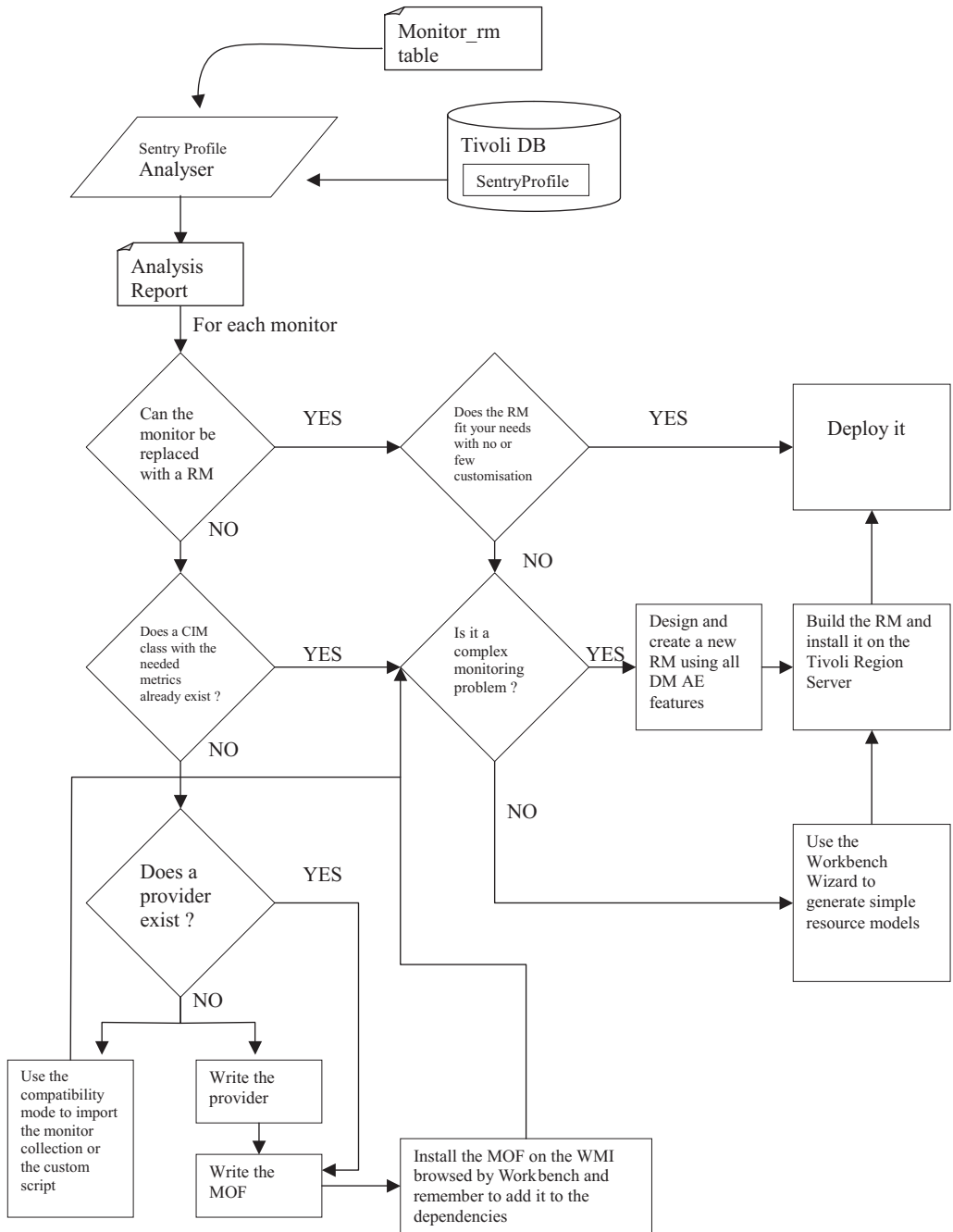
Obviously the generated code will take into account the Tivoli Monitoring 5.1 features described in earlier sections so it will be optimized to fit into the new monitoring paradigm. The wizard process is described in detail in *IBM Tivoli Monitoring: Workbench User's Guide*

### How Everything Fits Together

The following picture summarizes the migration approach that new Tivoli Monitoring 5.1 users should follow after taking into consideration what has been described above:

1. The Sentry Analyzer script analyzes the Sentry Profile based on a monitor-resource model mapping table
2. The Sentry Analyzer script generates a report
3. For each monitor in the report the user can:
  - a. If a resource model collecting the related metric exists:
    - 1) If the resource model addresses the monitoring problems controlled by the monitor, customize the resource model and deploy it.
    - 2) If the resource model does not address the monitoring problems controlled by the monitor, go to step 4.

- b. If a resource model collecting the related metric does not exist, but a CIM class and its provider do, then go to step 4.
- c. If a resource model collecting the related metric does not exist and neither does a CIM class, but a provider does (usually this happens for WMI performance data providers):
  - 1) Write the MOF.
  - 2) Then go to step 4.
- d. If a resource model collecting the related metric does not exist, and nor do a CIM class nor a provider:
  - 1) Write the provider.
  - 2) Write the MOF as indicated.  
—OR—  
Use the compatibility mode.
  - 3) Then go to step 4.
- 4. If the monitoring problem is complex and needs correlation and aggregation:
  - a. Create a new resource model or modify an existing one using all features provided by Tivoli Monitoring 5.1.
  - b. Go to step 6.
- 5. If the monitoring problem is a simple one that follows the Tivoli Distributed Monitoring (Classic Edition) monitoring paradigm, then use the wizard process in the workbench to create a new resource model.
- 6. Build the resource model package and install it on the Tivoli management region using the **wdmrm** command.
- 7. Deploy the resource model.



## Migrating from Tivoli Web Component Manager

Tivoli Monitoring 5.1 provides some tools to facilitate the migration of Tivoli Web Component Manager users to Tivoli Monitoring:

- A Java tool (called Tims2XML) which parses the database on the Tivoli Web Component Manager server (also referred to as TIMS server) and retrieves information about all TIMS tasks.

- A set of Tivoli Monitoring resource models that have a behaviour very similar to Tivoli Web Component manager tasks

## Tims2XML

TIMs2XML is a Java application that collects data related to Tivoli Web Component Manager tasks which are running on a TIMS server. Once collected, the data is automatically formatted into XML.

TIMs2XML is available on the Tivoli Monitoring CD.

### Installation and Customization

To be used, TIMs2XML must be installed on the TIMS server whose tasks needs to be monitored.

Before using the tool, the *launch.bat* file (or *launch.sh* on UNIX) need to be customized. The CLASSPATH must be set to point to specific .jar files. These are:

- *fw-common.jar*
- *fw-svr.jar*
- *properties.jar*

In addition, the *lib* and *lib/properties* directories must be included in the CLASSPATH.

Another required component is the JDBC library that is provided with your Oracle/DB2 installation, for example *classes111.zip* for Oracle.

The following is an example of CLASSPATH definition:

```
set TIMS_HOME=D:\Tivoli\Internet\ManagementServer\TIMS set
CLASSPATH=%TIMS_HOME%\lib\fw-common.jar set
CLASSPATH=%CLASSPATH%;%TIMS_HOME%\lib\fw-svr.jar set
CLASSPATH=%CLASSPATH%;%TIMS_HOME%\lib\properties.jar set
CLASSPATH=%CLASSPATH%;%TIMS_HOME%\lib set
CLASSPATH=%CLASSPATH%;%TIMS_HOME%\lib\properties set
CLASSPATH=%CLASSPATH%;D:\Oracle\Ora81\jdbc\lib\classes111.zip set
CLASSPATH=%CLASSPATH%;.
```

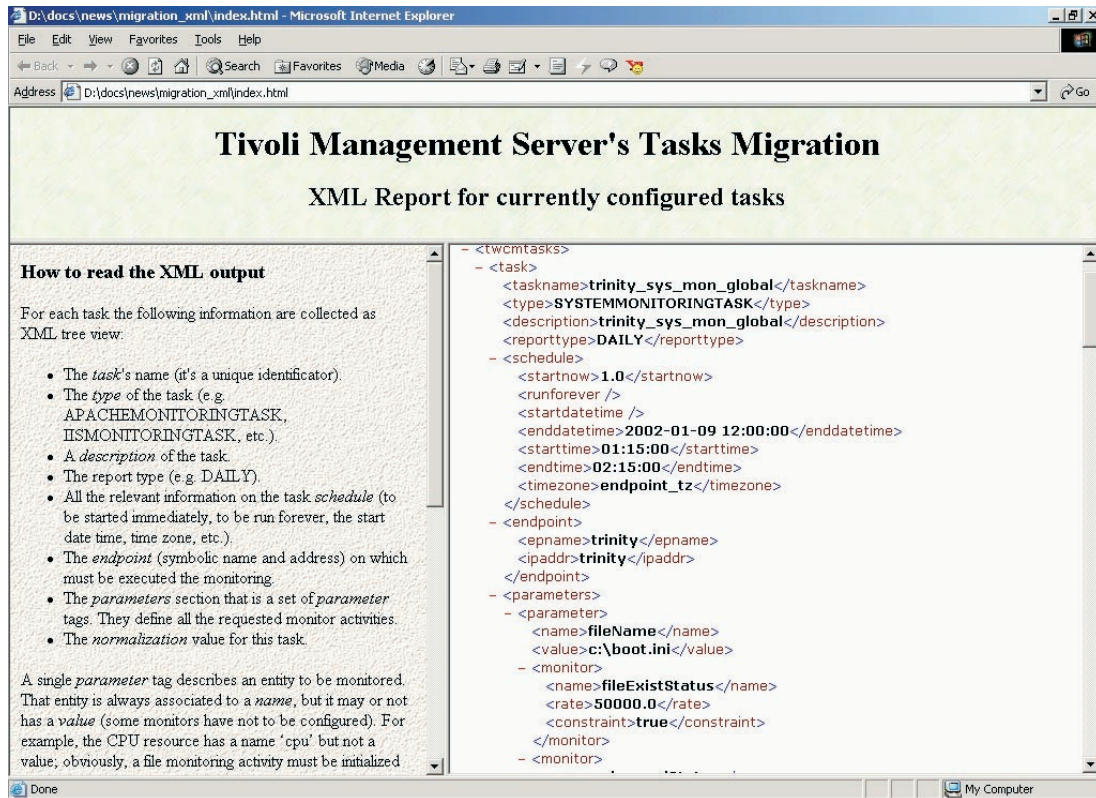
Once you have set the CLASSPATH, the Tims2XML can be launched by double-clicking the *launch.* file or by issuing the following command:

```
java Tims2XML > TIMS_tasks.xml
```

where *TIMS\_tasks.xml* is the name of the XML file that is generated.

## Structure of the XML file

A batch file and some HTML files are provided on the Tivoli Monitoring CD to show the contents of the XML file on Internet Explorer. The following is a sample:



The XML contains the following information for each task:

- *taskname*: the name of the task, as a unique identifier
- *type*: the type of task, which can be either of the following values:
  - APACHEMONITORINGTASK
  - IISMONITORINGTASK
  - SYSTEMMONITORINGTASK
- *description*: a description of the task
- *reporttype*: the type of report, which can be either of the following values:
  - DAILY
  - WEEKLY
  - MONTHLY
  - None

```

<?xml version="1.0" ?>
<!DOCTYPE twcmtasks (View Source for full doctype...)>
- <twcmtasks>
  - <task>
    <taskname>trinity_sys_mon_global</taskname>
    <type>SYSTEMMONITORINGTASK</type>
    <description>trinity_sys_mon_global</description>
    <reporttype>DAILY</reporttype>
    + <schedule>
    + <endpoint>
    + <parameters>
      <normalization>5.0</normalization>
    </task>
  </twcmtasks>

```

- *schedule*: information related to the task schedule
  - *startnow*: whether the job starts immediately or not
  - *runforever*: whether the job runs forever or not
  - *startdatetime*: starting date and time
  - *endtime*: ending date and time
  - *timezone*: information about the time zone
- *endpoint*: the endpoint on which the monitoring takes place

```

- <schedule>
  <startnow>1.0</startnow>
  <runforever />
  <startdatetime />
  <enddate>2002-01-09 12:00:00</enddate>
  <starttime>01:15:00</starttime>
  <endtime>02:15:00</endtime>
  <timezone>endpoint_tz</timezone>
</schedule>
- <endpoint>
  <epname>trinity</epname>
  <ipaddr>trinity</ipaddr>
</endpoint>

```

- *parameters*: a set of parameter tags that define the requested monitor activities
- *normalization*: the normalization value for the task. The normalization process occurs after a threshold has exceeded (see the *constraint* tag), and when the observed value is above or below a specified normalization value (for example, 10%)

Each single *parameter* tag describes an entity to be monitored. The entity is always associated to a *name*, but it may have (or not) a *value* (some monitors do not need to be configured). For example, the CPU resource has a name `cpu` but not a value; however, a file monitoring activity must be initialized with a path/file name value.

A parameter can be associated with several monitors, or no monitor at all (for example, a configuration parameter may not have a monitor associated). However, if a monitor is activated, then the XML output contains a monitor section.

The generic structure of a *monitor* tag is the following:

- *name* (mandatory)
- *rate* (not mandatory)
- *constraint* (not mandatory or empty)

The *rate* tag may not be applicable to a monitor (for example, it is not applicable to the Windows Event Log monitor).

The *constraint* tag is not applicable when the monitored attribute has a report only type (for example, the changedStatus monitor for a FILE resource), otherwise the *constraint* tag applies.

### System Monitoring Tasks

This section describes the XML output for the following System Monitoring tasks:

- Process Monitoring
- CPU Monitoring
- UNIX File System Monitoring

### Process Monitor Task

For the Process Monitor task, the *parameter* section shows the following tags:

- The *name* tag always contains the same identifier (processName)
- The name of the process is inside the *value* tag
- If the **N\_Threads Monitor** is active, then its monitor section contains:
  - The numThreads identifier into the *name* tag
  - The sampling interval (in seconds) into the *rate* tag
  - The threads threshold into the *constraint* tag
- If the **% CPU Monitor** is active, then its monitor section contains:
  - The percCPU identifier into the *name* tag
  - The sampling interval (in seconds) into the *rate* tag
  - The CPU occupation threshold (%) into the *constraint* tag
- If the **% Memory Monitor** is active, then its monitor section contains:
  - The percMem identifier into the *name* tag
  - The sampling interval (in seconds) into the *rate* tag
  - The memory occupation threshold (%) into the *constraint* tag



As an example, consider the Tivoli Web Component Manager configuration panel for the Process Monitor task:

Process			
Process Name		Sampling Interval (in seconds)	
java		50	
Enable	Monitor Name	Event Violation	Event Recovery
<input checked="" type="checkbox"/>	N Threads 30	Warning	Warning
<input checked="" type="checkbox"/>	CPU 50 %	Warning	Warning
<input checked="" type="checkbox"/>	Memory %	Warning	Warning

The following picture shows the corresponding section in the XML file:

```

- <parameter>
  <name>processName</name>
  <value>java</value>
- <monitor>
  <name>numThreads</name>
  <rate>50000.0</rate>
  <constraint>30.0</constraint>
</monitor>
- <monitor>
  <name>percCPU</name>
  <rate>50000.0</rate>
  <constraint>50.0</constraint>
</monitor>
- <monitor>
  <name>percMem</name>
  <rate>50000.0</rate>
  <constraint />
</monitor>
</parameter>
    
```

### CPU Monitor Task

For the CPU Monitor task, the *parameter* section shows the following tags:

- The *name* tag always contains the same identifier (cpu)
- The name of the process is inside the *value* tag
- If the **System CPU Monitor** is active, then its monitor section contains:
  - The sysCPU identifier into the *name* tag
  - The sampling interval (in seconds) into the *rate* tag
  - The system CPU occupation threshold into the *constraint* tag
- If the **% User CPU Monitor** is active, then its monitor section contains:

- The userCPU identifier into the *name* tag
- The sampling interval (in seconds) into the *rate* tag
- The user CPU occupation threshold (%) into the *constraint* tag
- If the % **Total CPU Monitor** is active, then its monitor section contains:
  - The totCPU identifier into the *name* tag
  - The sampling interval (in seconds) into the *rate* tag
  - The total CPU occupation threshold (%) into the *constraint* tag

As an example, consider the Tivoli Web Component Manager configuration panel for the CPU Monitor task:

CPU				
Enable	Monitor Name	Sampling Interval (in seconds)	Event Violation	Event Recovery
<input checked="" type="checkbox"/>	System CPU 40 %	50	Warning	Warning
<input checked="" type="checkbox"/>	User CPU %	50	Warning	Warning
<input checked="" type="checkbox"/>	Total CPU 80 %	50	Warning	Warning

The following picture shows the corresponding section in the XML file:

```

- <parameter>
  <name>cpu</name>
  - <monitor>
    <name>sysCPU</name>
    <rate>50000.0</rate>
    <constraint>40.0</constraint>
  </monitor>
  - <monitor>
    <name>userCPU</name>
    <rate>50000.0</rate>
    <constraint />
  </monitor>
  - <monitor>
    <name>totCPU</name>
    <rate>50000.0</rate>
    <constraint>80.0</constraint>
  </monitor>
</parameter>

```

### UNIX File System Monitor Task

Within a single task it is possible to monitor up to 10 UNIX File Systems simultaneously. Therefore we univocally associate each file system to a *parameter* section:

- The *name* tag always contains the same identifier (fileName)
- The file system mount point is inside the *value* tag
- The name of the process is inside the *value* tag

- If the **Percentage Used Monitor** is active, then its monitor section contains:
  - The fsUsed identifier into the *name* tag
  - The sampling interval (in seconds) into the *rate* tag
  - The FS used threshold into the *constraint* tag
- If the **% MB Used Monitor** is active, then its monitor section contains:
  - The mbFSUsed identifier into the *name* tag
  - The sampling interval (in seconds) into the *rate* tag
  - The FS used threshold (in MBytes) into the *constraint* tag
- If the **INodes Free Monitor** is active, then its monitor section contains:
  - The iNodesFSFree identifier into the *name* tag
  - The sampling interval (in seconds) into the *rate* tag
  - The INodes Free threshold (%) into the *constraint* tag
- If the **Percentage INodes Free Monitor** is active, then its monitor section contains:
  - The percINodesFSFree identifier into the *name* tag
  - The sampling interval (in seconds) into the *rate* tag
  - The INodes Free threshold (%) into the *constraint* tag

As an example, consider the Tivoli Web Component Manager configuration panel for the File System Monitor task:

Monitor Name	Sampling Interval (in seconds)	Event Violation	Event Recovery
Percentage Used Monitor	50	Unknown	Unknown
MB Used Monitor	40	Unknown	Unknown
INodes Free Monitor	20	Unknown	Unknown
Percentage INodes Free Monitor	10	Unknown	Unknown

File System Name	Percentage Used Monitor		MB Used Monitor		INodes Free Monitor		Percentage INodes Free Monitor	
	Enable	Percentage Used	Enable	MB Used	Enable	INodes Free	Enable	Percentage INodes Free
1 /data	<input checked="" type="checkbox"/>	50 %	<input checked="" type="checkbox"/>	1024	<input checked="" type="checkbox"/>	5000	<input checked="" type="checkbox"/>	70 %

The following picture shows the corresponding section in the XML file:

```
- <parameter>
  <name>fileSystemName</name>
  <value>/data</value>
  - <monitor>
    <name>fsUsed</name>
    <rate>50000.0</rate>
    <constraint>50.0</constraint>
  </monitor>
  - <monitor>
    <name>mbFSUsed</name>
    <rate>40000.0</rate>
    <constraint>1024.0</constraint>
  </monitor>
  - <monitor>
    <name>iNodesFSFree</name>
    <rate>20000.0</rate>
    <constraint>5000.0</constraint>
  </monitor>
  - <monitor>
    <name>percINodesFSFree</name>
    <rate>10000.0</rate>
    <constraint>70.0</constraint>
  </monitor>
</parameter>
```

## Resource Models

To assist with the migration, Tivoli Monitoring provides a set of resource models that have a behaviour very similar to Tivoli Web Component Manager tasks.

The Tivoli Monitoring CD contains six resource models (three for Windows, and 3 for UNIX):

- Process Monitor resource model (Windows and UNIX)
- Memory Monitor resource model (Windows and UNIX)
- Logical Disk Monitor resource model (Windows)
- File System Monitor (UNIX)

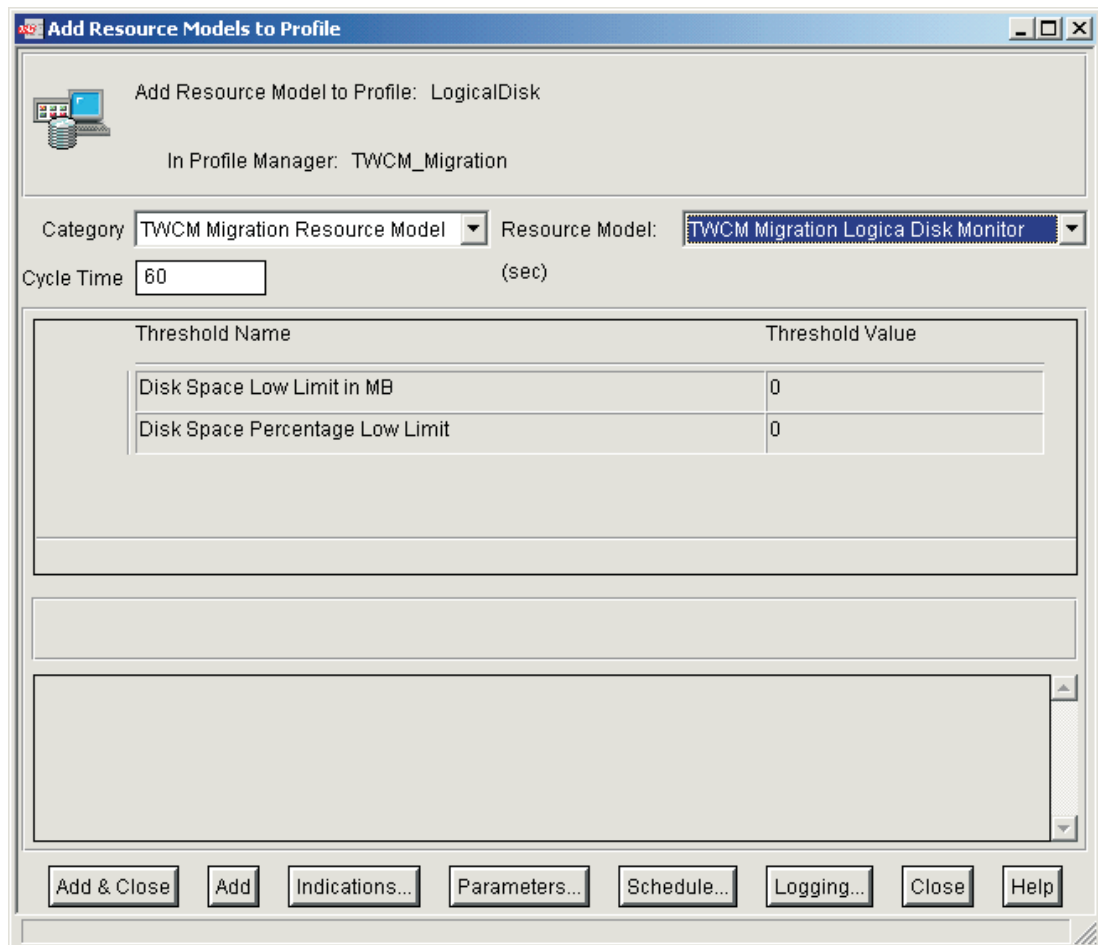
In order to use these resource models, you need to add them to the Tivoli Monitoring server by means of the **wdmrm -add** command.

### Logical Disk Resource Model

The Logical Disk resource model monitors logical disks on a Windows system.

To use this resource model, create a new Tivoli Monitoring profile, then select the **TWCM Migration Resource Model** category, and the **TWCM Migration Logical Disk Monitor**

resource model, as shown in the following dialog:



This resource model can monitor disk space and disk space percentage.

The thresholds value is by default set to zero, so that no event is generated by the resource model, but you can modify this value by clicking the **Threshold Name** and entering a new value.

You can select the logical disks that needs to be monitored by clicking the **Parameter** button and by modifying the default values.

By default, data logging is not enabled. To have a log, click **Logging**.

### Processor Resource Model

The Memory resource model is available for both Windows and UNIX systems.

To use this resource model, create a new Tivoli Monitoring profile, then select the **TWCM Migration Resource Model** category, and the **TWCM Migration Processor Model** (or **TWCM Migration Unix Processor Model**) resource model, as shown in the following

screen:

Category	TWCM Migration Resource Model	Resource Model:	TWCM Migration Unix Processor Model
Cycle Time	60	(sec)	
Threshold Name			
User Percentage Upper Threshold			
System Percentage Upper Threshold		100	
Total Percentage Upper Threshold		100	

### Memory Resource Model

The Memory resource model is available for both Windows and UNIX systems.

To use this resource model, create a new Tivoli Monitoring profile, then select the **TWCM Migration Resource Model** category, and the **TWCM Migration Memory Model** (or **TWCM Migration Unix Memory Model**) resource model.

### File System Resource Model

The File System resource model is available only for UNIX systems.

To use this resource model, create a new Tivoli Monitoring profile, then select the **TWCM Migration Resource Model** category, and the **TWCM Migration Memory Model** (or **TWCM Migration Unix File System Model**) resource model.

This resource model can monitor Free INode Percentage and Free Space Percentage. By default, all thresholds are set to zero so that the resource model does not generate any event.

To select the file systems to monitor, click **Parameter** and set the appropriate mount point values (if no values are set, the resource model will give an error).

### Migration Sample: Memory Monitor Task

This section describes how a Memory Monitor Task on a UNIX system can be migrated to Tivoli Monitoring.

The following screen shows the Tivoli Web Component Manager configuration panel for the Memory Monitor task:

Memory				
Enable	Memory Used	Sampling Interval (in seconds)	Event Violation	Event Recovery
<input checked="" type="checkbox"/>	49 %	50	Warning	Warning

Once the Tims2XML tool has retrieved the information about the tasks running on the TIMS server, we get the following XML output for the Memory Monitor task:

```

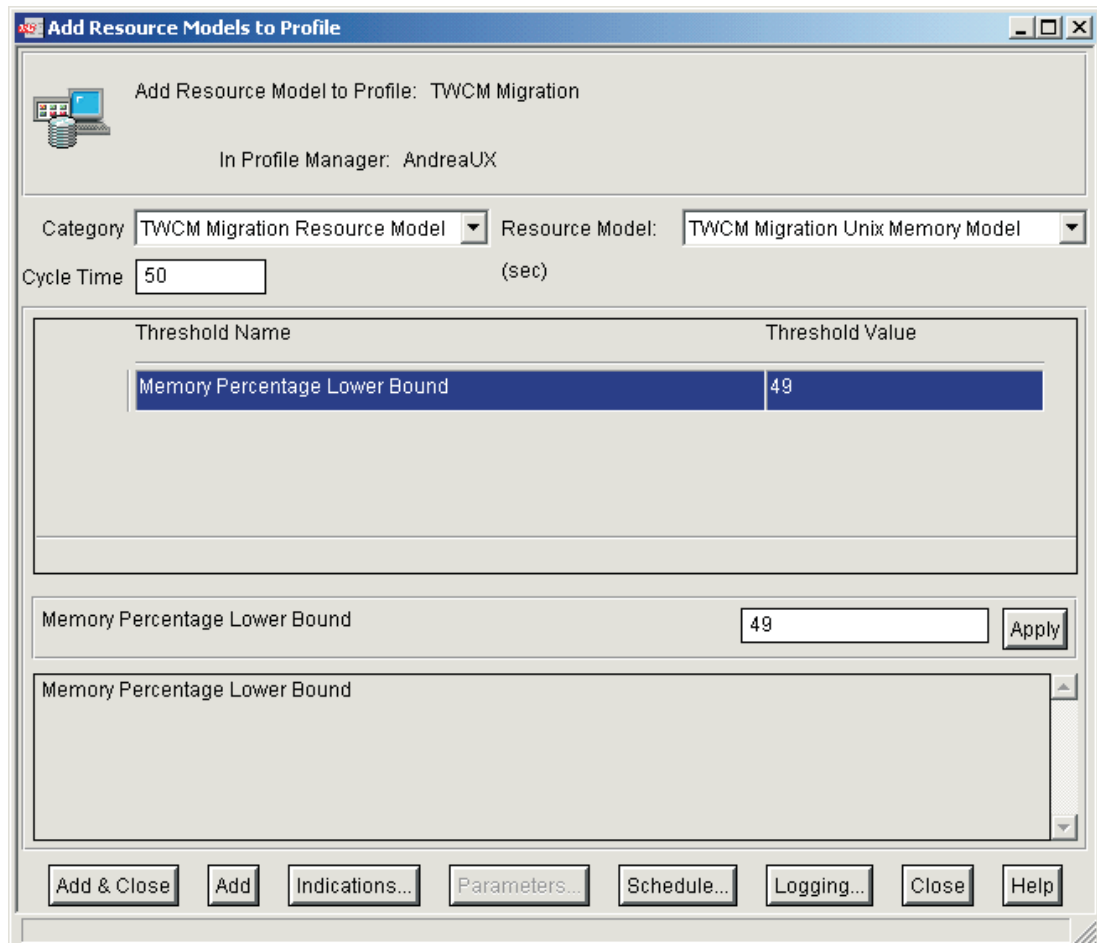
- <parameter>
  <name>memory</name>
  - <monitor>
    <name>memUsed</name>
    <rate>50000.0</rate>
    <constraint>49.0</constraint>
  </monitor>
</parameter>

```

To replicate this behaviour in Tivoli Monitoring, a new profile needs to be created with the following data:

- Category: TWCM Migration Model
- Resource Model: TWCM Migration Unix Memory Model
- Cycle Time: 50
- Memory Percentage Lower Bound: 49

as shown in the following dialog:







---

# Glossary

## A

**Adapter Configuration Facility.** In the Tivoli Enterprise Console, a graphical user interface that enables a Tivoli administrator to easily configure and customize event adapters.

## C

**cache.** A buffer storage that contains frequently accessed instructions and data; it is used to reduce access time.

**configuration file.** A file that specifies the characteristics of a system device or network.

## D

**data view.** In IBM Tivoli Monitoring, a way of displaying data in the Web Health Console.

**database browser.** In IBM Tivoli Monitoring, the Web Health Console component that provides hierarchical access to information about current resource problems. This information can come from any endpoint to which the Web Health Console has access, and it might pertain to any resource model installed on that endpoint. See also Web Health Console and resource model.

**default policy.** In a Tivoli environment, a set of resource property values that are assigned to a resource when the resource is created.

**defragmentation.** The process of running a software utility to rewrite fragmented data to contiguous sectors of a computer storage medium to improve access and retrieval time.

**Desktop Management Task Force (DMTF).** An alliance of computer vendors that was convened to define streamlined management of the diverse operating systems commonly found in an enterprise.

## E

**endpoint.** In a Tivoli environment, a Tivoli client that is the ultimate recipient for any type of Tivoli operation.

**event.** In a Tivoli environment, any significant change in the state of a system resource, network resource, or network application. An event can be generated for a problem, for the resolution of a problem, or for the successful completion of a task. Examples of events are the normal starting and stopping of a process, the abnormal termination of a process, and the malfunctioning of a server. See also *indication*.

**event class.** In the Tivoli Enterprise Console, a classification for an event that indicates the type of information that the event adapter will send to the event server.

## F

**fragmentation.** An operating system's process of writing different parts of a file to discontinuous sectors on a computer storage medium when contiguous space that is large enough to contain the entire file is not available. When data is thus fragmented, the time that it takes to access the data may increase because the operating system must search different tracks for information that should be in one location.

## G

**gateway.** In a Tivoli environment, software running on a managed node that provides all communication services between a group of endpoints and the rest of the Tivoli environment. This gateway includes the multiplexed distribution (MDist) function, enabling it to act as the fanout point for distributions to many endpoints.

## H

**Web Health Console.** In IBM Tivoli Monitoring, a component that displays real-time and historical data for any resource model at any endpoint. Using the graphical user interface, users can locate individual problems associated with one or more resources. The status is displayed as a value between 0 (representing an identified problem, that is, an event) and 100 (representing no recent indications). Users can select views of resource problems as tabular data, different types of charts, and so on.

## I

**IBM Tivoli Monitoring.** A Tivoli application that applies preconfigured, automated best practices to the automated monitoring of essential system resources. The application detects bottlenecks and other potential problems and provides for the automatic recovery from critical situations, which eliminates the need for system administrators to manually scan through extensive performance data. The application also integrates seamlessly with other Tivoli Availability solutions, including the Tivoli Business Systems Manager and the Tivoli Enterprise Console. Previously called Tivoli Distributed Monitoring for Windows.

**indication.** An entity triggered by the occurrence of a problem in an endpoint relating to one or more resources. Indications are consolidated into events within the endpoint being monitored. See also *event*.

**installation repository (IR).** In Tivoli Software Installation Service (SIS), the directory that contains reusable installation images and other data that is used by SIS.

---

## M

**managed node.** In a Tivoli environment, any managed resource on which the Tivoli Management Framework is installed.

**monitoring collection.** In Tivoli Distributed Monitoring (Classic Edition), a collection of predefined monitors. Several monitoring collections are packaged with Tivoli Distributed Monitoring, but Tivoli administrators can also use custom-developed and third-party monitoring collections. See also *custom monitor*.

## O

**object.** In object-oriented design or programming, a concrete realization of a class that consists of data and the operations associated with that data.

**oserv.** The Tivoli service that is used as the object request broker (ORB). This service runs on the Tivoli management region server and each managed node.

## P

**policy region.** In a Tivoli environment, a group of managed resources that share one or more common policies. Tivoli administrators use policy regions to model the management and organizational structure of a network computing environment. The administrators can group similar resources, define access to and control the resources, and associate rules for governing the resources.

**policy subregion.** In a Tivoli environment, a policy region created or residing in another policy region. When a policy subregion is created, it initially uses the resource and policy properties of the parent policy region. The Tivoli administrator can later change or customize these properties to reflect the specific needs and differences of the subregion.

**profile.** In a Tivoli environment, a container for application-specific information about a particular type of resource. A Tivoli application specifies the template for its profiles; the template includes information about the resources that can be managed by that Tivoli application. A profile is created in the context of a profile manager; the profile manager links a profile to the Tivoli resource (for example, a managed node) that uses the information contained in the profile. A profile does not have any direct subscribers.

**profile manager.** In a Tivoli environment, a container for profiles that links the profiles to a set of resources, called subscribers. Tivoli administrators use profile managers to organize and distribute profiles. A profile manager is created in the context of a policy region and is a managed resource in a policy region.

## R

**reference model.** In the context of Tivoli software, the model configuration for a system or set of systems that is used to

maintain consistent configurations in a distributed environment. In Tivoli Inventory, reference models are created in the configuration repository.

**resource.** In Tivoli Monitoring context, a resource is anything that affects the operation of the system and includes physical and logical disks, CPUs, memory, printers, as well as the processes running, services, such as LanMan and the Windows event log, and TCP/IP.

**resource model.** In IBM Tivoli Monitoring, the logical modeling of one or more resources, along with the logic on which cyclical data collection, data analysis, and monitoring are based. Related events and actions are triggered, if required. For any resource model, users can specify individual thresholds and event aggregation rules. See also *event*.

**rule base.** In the Tivoli Enterprise Console, one or more rule sets and the event class definitions for which the rules are written. The Tivoli Enterprise Console uses the rule base in managing events. An organization can create many rule bases, with each rule base fulfilling a different set of needs for network computing management.

## S

**Software Installation Service (SIS).** A Tivoli product that provides an easy-to-use, efficient interface for installing Tivoli Enterprise software. SIS uses Tivoli's MDist technology and provides automated checking for prerequisite software, a reusable repository of installation images, and both graphical and command line interfaces for deploying Tivoli products to a large number of computers.

**subscriber.** In a Tivoli environment, a managed node, a profile manager, an endpoint, or another Tivoli client that is subscribed to a profile manager. Although profiles are distributed to a subscriber, the subscriber may or may not be the final destination of the profile distribution.

## T

**task library.** In a Tivoli environment, a container in which a Tivoli administrator can create and store tasks and jobs.

**threshold.** (1) In software products, a value that defines a limit for a monitored condition. (2) In IBM Tivoli Monitoring, a threshold is a named property with a user-defined value. Typically, the value specified for a threshold represents a significant level of a performance-related entity, which, if exceeded, a system administrator might want to know about.

**Tivoli Distributed Monitoring.** Previous name of Tivoli Distributed Monitoring (Classic Edition).

**Tivoli Distributed Monitoring (Advanced Edition).** Previous name of IBM Tivoli Monitoring.

**Tivoli Distributed Monitoring (Classic Edition).** A Tivoli application that provides distributed monitors for monitoring system resources. The application initiates necessary corrective actions and informs system administrators of potential

---

problems. These monitors can be centrally configured and deployed to monitor individual machines. The application also integrates seamlessly with other Tivoli Availability solutions, including the Tivoli Enterprise Console. Previously called Tivoli Distributed Monitoring.

**Tivoli Distributed Monitoring for Windows.** Previous name of Tivoli Distributed Monitoring (Advanced Edition).

**Tivoli Enterprise Console.** A Tivoli product that collects, processes, and automatically initiates corrective actions for system, application, network, and database events; it is the central control point for events from all sources. The Tivoli Enterprise Console provides a centralized, global view of the network computing environment; it uses distributed event monitors to collect information, a central event server to process information, and distributed event consoles to present information to system administrators.

**Tivoli environment.** The Tivoli applications, based upon the Tivoli Management Framework, that are installed at a specific customer location and that address network computing management issues across many platforms. In a Tivoli environment, a system administrator can distribute software, manage user configurations, change access privileges, automate operations, monitor resources, and schedule jobs.

**Tivoli management agent.** In the Tivoli environment, an agent that securely performs administrative operations.

**Tivoli Management Framework.** The base software that is required to run the applications in the Tivoli product suite. This software infrastructure enables the integration of systems management applications from Tivoli Systems Inc. and the Tivoli Partners. In a Tivoli environment, the Tivoli Management Framework is installed on every client and server; however, the Tivoli management region server is the only server that holds the full object database.

**Tivoli management region.** In a Tivoli environment, a Tivoli server and the set of clients that it serves. An organization can have more than one Tivoli management region. A Tivoli management region addresses the physical connectivity of resources whereas a policy region addresses the logical organization of resources.

---

---

# Index

## A

- Access ODBC driver, as prerequisite 21
- accessibility 203
- actions (recovery), concept 11
- actions, built-in
  - adding 48
  - editing 47
  - removing 48
- Adapter Configuration Facility, as prerequisite for Tivoli Enterprise Console server 22
- AIX, supported versions 19
- assured delivery, MDist2 57
- asynchronous delivery, MDist2 56
- Autotrace 190

## B

- books
  - feedback xviii
  - online xviii
  - ordering xviii
- built-in actions
  - adding 48
  - concept 11
  - editing 47
  - removing 48
- bulk data transfer 17

## C

- cache, endpoint 64
- check-point and restart, MDist2 57
- clearing events, concept 10
- coexistence 22
- collecting the data 15
- commands
  - for adding customized resource models to a profile 44
  - for adding default resource models to a profile 43
  - for customizing resource models 55
  - for managing default resource models at the server 41, 108
  - for managing profiles at the server 40, 108
  - for managing resource models and profiles at endpoints 58
  - for managing the heartbeat function at gateways 60
  - for managing the product at endpoints 108
  - for managing the product at gateways 58, 109
  - for managing the Tivoli Business Systems Manager Adapter 108
  - installing product with 28
  - upgraded names 107
  - wdmcmd 110

- commands (*continued*)
  - wdmconfig 111
  - wdmDiscovery 114
  - wdmdumpprf 118
  - wdmeditprf 122
  - wdmeng 129
  - wdmheartbeat 131
  - wdmloadprf 133
  - wdmlseng 135
  - wdmmn 139
  - wdmmngcache 141
  - wdmrm 116, 144
  - wdmtrceng 146
  - winstall 28
  - wtmntaddrm 148
  - wtmntdefrm 149
  - wtmntrmrm 150
- components
  - diagram 13
  - endpoint 15
  - Health Console 15
  - IBM Tivoli Monitoring 14
  - installation 25
  - introduction 13
  - Tivoli Business Systems Manager Adapter 15
  - upgrading 29
- concepts
  - actions (recovery) 11
  - built-in actions 11
  - clearing events 10
  - cycles 7
  - data logging 12
  - event monitors 10
  - events 8
  - events (clearing) 10
  - Gathering Historical Data Warehouse 12
  - heartbeat function 12
  - holes 8
  - indications 7
  - logging (data) 12
  - monitors 10
  - occurrences 8
  - parameters 7
  - profiles 12
  - recovery actions 11
  - resource models 6
  - resources 6
  - scheduling 12
  - tasks (Tivoli Management Framework) 12
  - thresholds 7
  - Tivoli Management Framework tasks 12
- configuration
  - heartbeat function 59
  - Tivoli Business Systems Manager Adapter 74
- Customer Support xx

---

customizing profiles and resource models 37  
cycles, concept 7

## D

data collection 15  
data depoting, MDist2 57  
data flow  
    Health Console, events 16  
    resource models 15  
    Tivoli Business Systems Manager CommonListener,  
        events 16  
    Tivoli Enterprise Console server, events 16  
data logging  
    concept 12  
    customizing 53  
    endpoint prerequisites for Windows NT 4.0 21  
desktop, installing product with 25  
directory names, notation xxi  
distribution  
    logs/traces 181  
    options 55

## E

e-mail contact xx  
endpoint 190  
    Autotrace log/traces 190  
    cache viewing 64  
    collecting the data at 15  
    component 15  
    data collection 15  
    engine  
        logs/traces 186, 187  
    engine update  
        logs/traces 187  
    JMX  
        logs/traces 188  
    logs/traces 186  
    native  
        logs/traces 188  
    obtaining resource data 16  
    prerequisites 22  
    profile management 58  
    resource model management 58  
    uninstalling product 34  
endpoint upcall  
    logs/traces 184  
environment variables, notation xxi  
error messages 151  
event monitors  
    concept 10  
    Health Console 11  
    Tivoli Business Systems Manager 11  
    Tivoli Enterprise Console server 10  
events  
    concept 8

events (*continued*)  
    customizing 44  
    data flow  
        Health Console 16  
        Tivoli Business Systems Manager CommonListener 16  
        Tivoli Enterprise Console server 16  
    events (clearing), concept 10

## F

feedback about publications xx  
firewalls 17

## G

gateway  
    logs/traces 182  
    managing 58  
    uninstalling product from 34  
Gathering Historical Data Warehouse  
    concept 12

## H

hardware requirements  
    IBM Tivoli Monitoring 20  
Health Console  
    as monitor, concept 11  
    component 15  
    data flow, events 16  
heartbeat function  
    concept 12  
    control 60  
    data flows 61  
    description 59  
    endpoint registration 61  
    engine, logs/traces 181, 182  
    installation 59  
    managing 60  
    monitoring 63  
    statuses 59  
    viewing the endpoint cache 64  
holes  
    concept 8  
    customizing 44, 46  
HPUX, supported versions 20

## I

IBM Tivoli Monitoring  
    coexistence 22  
    component 14

---

## IBM Tivoli Monitoring (*continued*)

- hardware requirements 20
- installation 25
- notice group addition 31
- overview 5
- software requirements 20
- uninstalling 34

### indications

- concept 7
- customizing 44

### informational messages 151

### installation

- before beginning 24
- default resource models 31
- heartbeat function 59
- Java Runtime Environment 32
- of product using commands 28
- of product with Software Installation Service 24
- of product with Tivoli Desktop 25
- Tivoli Business Systems Manager Adapter 73

## J

### Java Runtime Environment

- as prerequisite 21, 22
- installation 32

### Jet 4.0 Service Pack 3, as prerequisite 21

## L

### library, product xviii

### Linux

- endpoint engine logs and traces 187
- endpoint engine update logs and traces 187
- endpoint JMX logs and traces 188
- endpoint native logs and traces 188
- Red Hat for Intel, supported versions 19
- SuSE for Intel, supported versions 19
- SuSE for S/390, supported versions 19
- Turbo for Intel, supported versions 19

### logging (data)

- concept 12
- customizing 53
- endpoint prerequisites for Windows NT 4.0 21

### logs

- distribution 181
- endpoint 186
- endpoint Autotrace
  - UNIX 190
- endpoint engine
  - Linux 187
  - UNIX 187
  - Windows 186, 187
- endpoint engine update
  - Linux 187
  - UNIX 187

### logs (*continued*)

- endpoint JMX
  - Linux 188
  - UNIX 188
- endpoint native
  - Linux 188
  - UNIX 188
- endpoint upcall 184
- gateway 182
- heartbeat function engine 181, 182
- overview 179
- server 181
- task engine 183, 185
- Tivoli Business Systems Manager Adapter 185
- Tivoli Business Systems Manager engine 183
- Tivoli Business Systems Manager transport 185
- Tivoli Management Framework 181
- Windows Management Instrumentation 186

## M

### managed resources, setting up 38

### manuals

- feedback xviii
- online xviii
- ordering xviii

### mdac\_typ.exe, as prerequisite 21

### MDist2

- assured delivery 57
- asynchronous delivery 56
- check-point and restart 57
- data depoting 57
- using 56

### messages

- error 151

### Microsoft Access ODBC driver, as prerequisite 21

### monitors

- concepts 10
- diagram 13
- Health Console 11
- Tivoli Business Systems Manager 11
- Tivoli Enterprise Console server 10

## N

### notation

- environment variables xxi
- path names xxi
- typeface xxi

### notice group

- IBM Tivoli Monitoring
  - adding 31

---

## O

- occurrences
  - concept 8
  - customizing 46
- ODBC driver for Microsoft Access, as prerequisite 21
- online publications xix
- operating systems supported
  - for IBM Tivoli Monitoring 19
- ordering publications xix
- overview
  - IBM Tivoli Monitoring 5

## P

- parameters
  - concepts 7
  - customizing 51
- path names, notation xxi
- platforms supported 19
- preinstallation notes 24
- prerequisites
  - hardware 20
  - software 20
  - Software Installation Service 22
  - Tivoli Business Systems Manager Adapter 22
  - upgrading 22
- problem determination 193
- profile manager 38
  - creating 39
  - for rerunning failed distributions 57
- profiles
  - adding customized resource models 43
  - adding default resource models 41
  - concept 12
  - creating 39
  - customizing 37
  - description 38
  - distributing using MDist2 56
  - key points 38
  - managing at endpoint 58
  - using commands to manage at the server 40
- publications
  - feedback xviii
  - online xviii
  - ordering xviii

## R

- recovery actions, concept 11
- Red Hat Linux for Intel, supported versions 19
- removing
  - default resource models 31
- resource models
  - concept 6
  - creating with workbench 17
  - customized, adding to profiles 43

- resource models (*continued*)
  - customizing 37, 44
  - data flow 15
  - default
    - adding to profiles 41
    - installing 31
    - removing 31
    - using commands to manage at the server 41
  - installing default 31
  - managing at endpoint 58
  - modifying with workbench 17
  - removing default 31
- resources
  - concept 6
  - obtaining data at endpoint 16
- rules, scheduling 53

## S

- scheduling
  - concept 12
  - customizing 52
  - rules 53
- Secure Socket Layer 3 17
- security considerations 17
- server
  - logs/traces 181
  - uninstalling product from 34
- Software Installation Service
  - installing product with 24
  - prerequisites 22
- software requirements
  - IBM Tivoli Monitoring 20
- Solaris, supported versions 19
- SSL3 17
- statuses, heartbeat function 59
- supported platforms 19
- SuSE Linux for Intel, supported versions 19
- SuSE Linux for S/390, supported versions 19

## T

- task engine
  - logs/traces 183, 185
- tasks (Tivoli Management Framework)
  - concept 12
  - specifying 48
- thresholds
  - concept 7
  - customizing 44
- timing rules 53
- Tivoli Business Systems Manager
  - as monitor, concept 11
  - description 73
  - engine
    - logs/traces 183



---

Tivoli Business Systems Manager (*continued*)

- integration
  - disabling 78
  - enabling 78
  - overview 73
- transport
  - logs/traces 185
  - viewing data 77
- Tivoli Business Systems Manager Adapter
  - component 15
  - configuration 74
  - hardware requirements 20
  - installation 25, 73
  - logs/traces 185
  - managing 79
  - prerequisites 22
  - using 76
- Tivoli Business Systems Manager CommonListener, data flow for events 16
- Tivoli Customer Support xx
- Tivoli Decision Support for Server Performance Prediction
  - installation 25
- Tivoli Desktop, installing product with 25
- Tivoli Distributed Monitoring (Advanced Edition)
  - upgrading 29
- Tivoli Distributed Monitoring (Classic Edition)
  - coexistence 22
- Tivoli Distributed Monitoring for Windows
  - coexistence 22
  - version 3.7.1 as prerequisite for upgrade 22
- Tivoli Distributed Monitoring Workbench 17
- Tivoli Enterprise Console server
  - as monitor, concept 10
  - data flow for events 16
  - enabling of events 68
  - enabling of heartbeat messages 68
  - endpoint prerequisites 22
  - identifying to the product 67
  - integration 67
- Tivoli Enterprise Data Warehouse
  - enabling Tivoli Monitoring to work with 34
- Tivoli Management Framework
  - firewalls 17
  - logs/traces 181
  - software requirements 21
  - tasks, concept 12
- Tivoli Monitoring, coexistence 22
- tools, endpoint, Windows 193
- traces
  - distribution 181
  - endpoint 186
  - endpoint Autotrace
    - UNIX 190
  - endpoint engine
    - Linux 187
    - UNIX 187
    - Windows 186, 187
  - endpoint engine update
    - Linux 187
    - UNIX 187

traces (*continued*)

- endpoint JMX
  - Linux 188
  - UNIX 188
- endpoint native
  - Linux 188
  - UNIX 188
- endpoint upcall 184
- gateway 182
- heartbeat function engine 181, 182
- overview 179
- server 181
- task engine 183, 185
- Tivoli Business Systems Manager Adapter 185
- Tivoli Business Systems Manager engine 183
- Tivoli Business Systems Manager transport 185
- Tivoli Management Framework 181
- Windows Management Instrumentation 186
- troubleshooting 179
- Turbo Linux for Intel, supported versions 19

## U

- uninstalling
  - from endpoint 34
  - from gateway 34
  - from server 34
  - product, overview 34
- UNIX
  - endpoint Autotrace logs and traces 190
  - endpoint engine logs and traces 187
  - endpoint engine update logs and traces 187
  - endpoint JMX logs and traces 188
  - endpoint native logs and traces 188
- UNIX, supported versions 19
- upgrading
  - instructions 29
  - prerequisites 22

## V

- variables, notation for xxi

## W

- warning messages 151
- wdmcmd 110
- wdmconfig 111
- wdmdiscovery 114
- wdmdumpprf 118
- wdmeditprf 122
- wdmeng 129
- wdmheartbeat 131
- wdmloadprf 133
- wdmlseng 135

---

wdmmn 139  
wdmmngcache 141  
wdmrm 116, 144  
wdmtrceng 146  
what's new in this release xv  
who should read this guide xv  
Windows  
    2000, supported versions 19  
    NT 4.0  
        endpoint prerequisites 21  
        supported versions 19  
    XP, supported versions 19  
Windows Management Instrumentation  
    as prerequisite 21  
    logs/traces 186  
winstall 28  
workbench 17  
wtmntaddrm 148  
wtmntdefrm 149  
wtmnttrrm 150





SH19-4569-00

