# BEA WebLogic Server™

## and BEA WebLogic Express™

## SNMP Management Guide

BEA WebLogic Server Version 6.1
Document Date: June 24, 2002

## Copyright



## Restricted Rights Legend



## Trademarks or Service Marks



**BEA WebLogic Server SNMP Management Guide**

| Document Date | Software Version |
|---|---|
| June 24, 2002 | BEA WebLogic Server Version 6.1 |

# Contents

## 3. Using Multiple SNMP Agents



## A. Sources of SNMP Information

# About This Document

This document explains the management subsystem provided for configuring and monitoring your WebLogic Server implementation. It covers the following topics:

- Chapter 1, “Using SNMP to Manage WebLogic Server,” describes basic concepts of Simple Network Management Protocol as they apply to managing WebLogic Servers. Setting up the WebLogic SNMP agent is also described.
- Chapter 2, “Trap Notifications,” describes the characteristics of WebLogic enterprise-specific SNMP trap notifications and how to configure the the WebLogic SNMP agent to generate SNMP traps.
- Chapter 3, “Using Multiple SNMP Agents,” describes how to use the WebLogic SNMP agent as a master agent that proxies for other SNMP agents.
- Appendix A, “Sources of SNMP Information,” provides sources of additional information about Simple Network Management Protocol.

## Audience

This document is intended mainly for system administrators who will be managing the WebLogic Server application platform and its various subsystems.

# e-docs Web Site

BEA product documentation is available on the BEA corporate Web site. From the BEA Home page, click on Product Documentation.

# How to Print the Document

You can print a copy of this document from a Web browser, one main topic at a time, by using the File→Print option on your Web browser.

A PDF version of this document is available on the WebLogic Server documentation Home page on the e-docs Web site (and also on the documentation CD). You can open the PDF in Adobe Acrobat Reader and print the entire document (or a portion of it) in book format. To access the PDFs, open the WebLogic Server documentation Home page, click Download Documentation, and select the document you want to print.

Adobe Acrobat Reader is available at no charge from the Adobe Web site at http://www.adobe.com.

# Contact Us!

Your feedback on BEA documentation is important to us. Send us e-mail at docsupport@bea.com if you have questions or comments. Your comments will be reviewed directly by the BEA professionals who create and update the documentation.

In your e-mail message, please indicate the software name and version you are using, as well as the title and document date of your documentation. If you have any questions about this version of BEA WebLogic Server, or if you have problems installing and running BEA WebLogic Server, contact BEA Customer Support through BEA WebSupport at http://www.bea.com. You can also contact Customer Support by using the contact information provided on the Customer Support Card, which is included in the product package.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address
- Your machine type and authorization codes
- The name and version of the product you are using
- A description of the problem and the content of pertinent error messages

# Documentation Conventions

The following documentation conventions are used throughout this document.

| Convention | Usage |
|---|---|
| Ctrl+Tab | Keys you press simultaneously. |
| *italics* | Emphasis and book titles. |
| `monospace text` | Code samples, commands and their options, Java classes, data types, directories, and file names and their extensions. Monospace text also indicates text that you enter from the keyboard.<br>*Examples*:<br>`import java.util.Enumeration;`<br>`chmod u+w *`<br>`config/examples/applications`<br>`.java`<br>`config.xml`<br>`float` |
| *`monospace italic text`* | Variables in code.<br>*Example*:<br>`String` *`CustomerName`*`;` |

| Convention | Usage |
|---|---|
| UPPERCASE TEXT | Device names, environment variables, and logical operators.<br>*Example*s:<br>LPT1<br>BEA_HOME<br>OR |
| `{ }` | A set of choices in a syntax line. |
| `[ ]` | Optional items in a syntax line. *Example*:<br>`java utils.MulticastTest -n name -a address [-p portnumber] [-t timeout] [-s send]` |
| `\|` | Separates mutually exclusive choices in a syntax line. *Example*:<br>`java weblogic.deploy [list\|deploy\|undeploy\|update] password {application} {source}` |
| `...` | Indicates one of the following in a command line:<br>■ An argument can be repeated several times in the command line.<br>■ The statement omits additional optional arguments.<br>■ You can enter additional parameters, values, or other information |
| .<br>.<br>. | Indicates the omission of items from a code example or from a syntax line. |

# 1 Using SNMP to Manage WebLogic Server

WebLogic Server software includes the ability to communicate with enterprise-wide management systems using Simple Network Management Protocol (SNMP). The WebLogic Server SNMP capability enables you to integrate management of WebLogic Servers into an SNMP-compliant management system that gives you a single view of the various software and hardware resources of a complex, distributed system.

This section discusses the following topics:

- The SNMP Agent/Manager Model
- The SNMP Agent Role in a WebLogic Domain
- SNMP MIB for WebLogic
- SNMP Community Names
- Setting Up the WebLogic SNMP Agent

## The SNMP Agent/Manager Model

SNMP management is based on the agent/manager model described in the network management standards defined by the International Organization for Standardization (ISO). In this model, a network/systems manager exchanges monitoring and control information about system and network resources with distributed software processes called *agents*.

Any system or network resource that is manageable through the exchange of information is a *managed resource*. This could be a software resource such as a Java Database Connectivity (JDBC) connection pool or a hardware resource such as a router.

Agents function as "collection devices" that typically gather and send data about the managed resource in response to a request from a manager. In addition, agents often have the ability to issue unsolicited reports to managers when they detect certain predefined thresholds or conditions on a managed resource. In SNMP terminology, these unsolicited event reports are called *trap notifications.*

A manager relies upon a database of definitions and information about the properties of managed resources and the services the agents support — this makes up the Management Information Base (MIB). When new agents are added to extend the management reach of a manager, the manager must be provided with a new MIB component that defines the manageable features of the resources managed through that agent. The manageable attributes of resources, as defined in an SNMP-compliant MIB, are called *managed objects*. Defining the heterogeneous components of an enterprise's distributed systems within a common MIB on the management station provides a unified perspective and single access point for managing system and network resources.

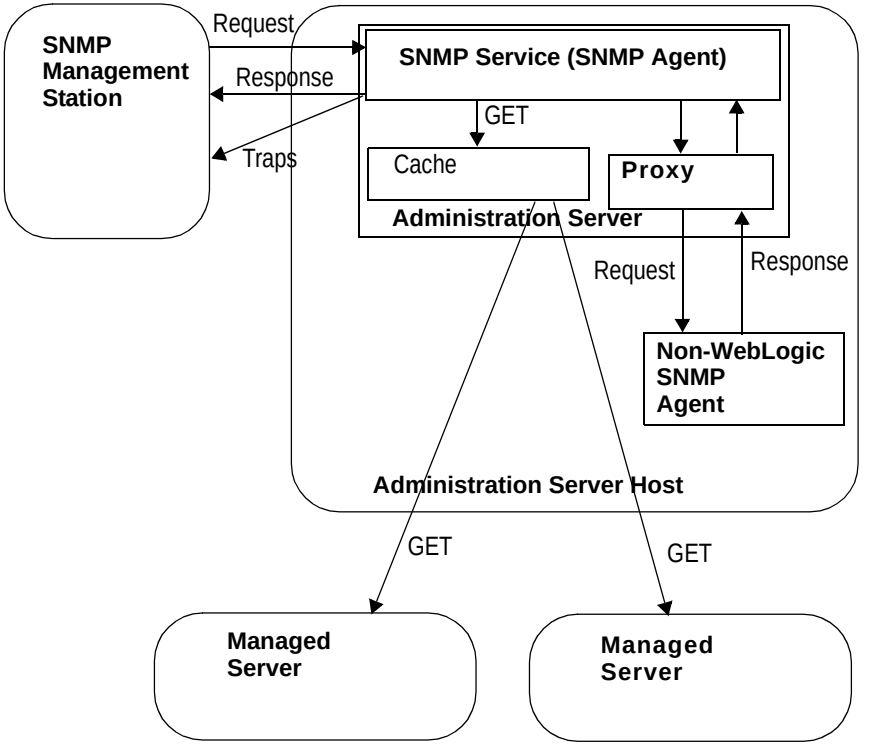
# The SNMP Agent Role in a WebLogic Domain

An inter-related set of WebLogic Server resources managed as a unit is called a *domain*. A domain includes one or more WebLogic Servers, and may include WebLogic Server clusters.

Within each WebLogic domain one server is the Administration Server; other servers in the domain are Managed Servers. A typical J2EE application may include components distributed across multiple Managed Servers. The Administration Server provides the central point of control for configuring and monitoring the entire WebLogic domain. For more on WebLogic domains, see the WebLogic Server Administration Guide.

The WebLogic Administration Server also has the ability to run the SNMP Service. When the SNMP Service is enabled for a WebLogic domain, the Administration Server is functioning as the SNMP agent for that WebLogic domain.

**Figure 1-1 SNMP Management of a WebLogic Domain**




You can use the WebLogic SNMP agent to do the following:

- Respond to simple GET requests from an SNMP manager for the current value of WebLogic attributes.
- Send trap notifications to SNMP managers when the Administration Server comes up and when any Managed Server goes up or down.
- Send trap notifications to SNMP managers when messages are logged in a Managed Server that satisfy criteria that you specify.

- Send trap notifications to SNMP managers when a WebLogic configuration attribute that you specify has changed value.
- Offload polling of WebLogic attributes to the WebLogic Administration Server using standard Java Management Extension (JMX) monitors, based on thresholds and polling intervals that you define. A trap notification is sent to the SNMP manager when the criteria you specify are satisfied.
- Act as a proxy agent that passes requests from an SNMP manager to other SNMP agents (such as an Oracle database agent) on the same machine.

# SNMP MIB for WebLogic

Extending the management reach of an SNMP manager to encompass a WebLogic domain requires a MIB that defines the manageable WebLogic attributes ("managed objects" in SNMP terminology) for the SNMP management system. The WebLogic MIB is defined in an SNMP-compliant file written in Abstract Syntax Notation One (ASN.1). This file is `BEA-WEBLOGIC-MIB.asn1`. This file is located under the following directory: *WL_HOME*`\wlserver6.1\lib`.

A MIB is based on a hierarchical relationship between managed objects. This hierarchical relationship is a tree structure, called the MIB tree or registration tree. Each managed object in the MIB is assigned a unique number called an *object identifier* (OID). An OID consists of a left-to-right sequence of integers. This sequence defines the location of the object in the MIB tree. By specifying a unique path through the tree to the object, the OID allows the object to be identified uniquely. Each node in the path defined in an OID has both a number and a name associated with it. The path `.1.3.6.1.4.1` defines the `private.enterprises` OID and each number beneath that node on the tree represents the branches in the tree reserved for a particular vendor.

The BEA MIBS are registered at the location `.1.3.6.1.4.1.140` in the tree. And the WebLogic Server MIB consists of all OIDs below `.1.3.6.1.4.140.625`. For each manageable WebLogic attribute the WebLogic MIB defines a corresponding OID. For example, `.1.3.6.1.4.1.140.625.360.1.60` is the OID for `serverRuntimeState`. When an SNMP manager requests the current value of a

WebLogic attribute, it indicates the attribute by specifying the corresponding OID. For more information about the contents of the WebLogic MIB, see the WebLogic Server SNMP MIB Reference.

**Note:** The enterprise OID for the SNMP agent for WebLogic 6.1 differs from the enterprise OID for WebLogic used with the WebLogic 5.1 SNMP agent. The enterprise OID for WebLogic 6.1 is `.1.3.6.1.4.140.625`. The textual name of the WebLogic Server node in the registration tree is now `wls`.

# SNMP Community Names

To ensure that the entity requesting data from the WebLogic SNMP agent has permission to obtain the data, and to verify that the agent has permission to send trap notifications to a target manager, SNMP uses textual passwords called *community names*.

When you set up the SNMP agent capability of the WebLogic Administration Server (described in Setting Up the WebLogic SNMP Agent), one of the things you must specify is the community name that the agent expects from the SNMP manager. If the agent receives an SNMP request with an incorrect community name, it automatically generates an `authenticationFailure` trap that is sent to the source of the request.

## Specifying the Target Server in Management Requests

In order to retrieve any attribute of any WebLogic Server in a domain, the SNMP manager must send the request to the Administration Server, acting as the SNMP agent for the domain. Because the same attribute, such as `serverUptime`, may exist on each of the WebLogic Servers, there needs to be a way to determine which WebLogic Server the manager is requesting data for.

In order to accomplish this, the target server name is appended to the SNMP password (community) as part of the community string sent with that request.

To specify a particular Managed Server, the SNMP manager sends to the WebLogic SNMP agent a community string that has the following form:

*community_prefix@server_name*

where *community_prefix* is the actual SNMP community name and *server_name* is the name of the target Managed Server. The *community_prefix* value sent by the manager must match the value that you set in the Community Prefix field when you configure the SNMP agent (as described in Setting Up the WebLogic SNMP Agent).

To send a management request for the attributes of the Administration Server, the SNMP manager must send a community string to the WebLogic SNMP agent that has the following form:

*community_prefix*

where *community_prefix* is the actual SNMP community name. The *community_prefix* value sent by the manager must match the value that you set in the Community Prefix field when you configure the SNMP agent (as described in Setting Up the WebLogic SNMP Agent).

If the SNMP manager sends a community string with the form

*community_prefix@domain_name*

the Administration Server returns the values of the specified attribute for every server in the domain.

## How to Access Runtime Information

**Note:** Consult the vendor information for specific information on how to configure your SNMP Management system to access runtime information.

This section provides an example of how to obtain runtime information on the `serverRuntimeListenAddress` for an administration server (myserver) and a managed server (clusterServer1). The MIB definition for `serverRuntimeListenAddress` is:

```
iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).bea(140).wls(625).se
rverRuntimeTable(360).serverRuntimeEntry(1).serverRuntimeListenAddress(30)
```

The following examples use a generic tool, `snmpgetnext`, to provide runtime information. `snmpgetnext` has the following arguments:

| Argument | Description |
|---|---|
| `-c communitystring` | Specifies the community string sent to the WebLogic Server SNMP agent. |
| `host` | Name of the host. |
| `OID` | Object Identifier for the runtime information you wish to access. |

To get the `serverRuntimeListenAddress` for a managed server:

```
snmpgetnext  -c public@clusterServer1 localhost .1.3.6.1.4.1.140.625.360.1.30
```

To get the `serverRuntimeListenAddress` for an administration server, use one of the following methods:

```
snmpgetnext  -c public@myserver      localhost  .1.3.6.1.4.1.140.625.360.1.30
snmpgetnext  -c public               localhost  .1.3.6.1.4.1.140.625.360.1.30
snmpgetnext                          localhost  .1.3.6.1.4.1.140.625.360.1.30
```

# Setting Up the WebLogic SNMP Agent

To make your WebLogic domain manageable via SNMP, do the following:

1. Load the WebLogic MIB (the file `BEA-WEBLOGIC-MIB.asn1`) into the SNMP management system.

   To figure out how to do this you may need to consult the vendor documentation for your SNMP management system.

2. Define the destinations where the WebLogic SNMP agent will send trap notifications:

   - With the Administration Server running, invoke the Administration Console, if it isn't running already.

- Select SNMP→SNMP Trap Destinations on the left panel. This invokes the SNMP Trap Destinations table.
- Select the link Create a new SNMP Trap Destination which invokes the Trap Destination configuration page.
- Fill in the Name field. This is a user-defined name of this particular destination.
- Fill in the Community field with the Community string that the target SNMP management station expects for incoming trap notifications.
- Enter the hostname or IP address of the SNMP management station in the Host field.
- Enter the Port number on which the SNMP management station expects to receive trap notifications. The default is 162.
- Click Apply to create your new SNMP trap destination table entry.

You can create additional entries in the SNMP trap destination table if you want traps to be sent to multiple targets.

3. Set up the SNMP agent for the WebLogic domain.

   To do this, do the following:

   - With the Administration Server running, invoke the Administration Console, if it isn't running already.
   - Select *domain_name*→Configuration→SNMP on the left panel (where *domain_name* is the name of the domain the SNMP service will manage).
   - Check the Enabled box so that the agent is activated. (By default the agent is off.)

     The SNMP agent functionality is activated only if this value is set to `true`.
   - In the SNMP Port field type the number of the port on which the SNMP manager will be sending requests. The default is 161.
   - The number in the MIB Data Refresh Interval field is the interval (in seconds) at which the SNMP Service will update its cache of current WebLogic attribute values. The minimum valid value is 30 seconds.

     The SNMP Service responds to SNMP manager requests for WebLogic attribute values with the value in its cache. It does not check the managed resource itself each time it receives a management request. The MIB Data

Refresh Interval is therefore the most frequent interval at which the manager can get a real update of the attribute value. If this interval is set to a very small value, this may have an adverse performance impact as the Administration Server will be doing a GET on every WebLogic attribute in the MIB at this interval.

- Set the Server Status Check Interval Factor.

  The SNMP Service automatically checks to determine if WebLogic Managed Servers are up or down. If the SNMP Service finds that a Managed Server which was up is now down, it sends a `serverShutDown` enterprise-specific trap to the targets defined in the SNMP Trap Destination table. A `serverStart` trap is sent if a server that was down is now up.

  The most frequent interval at which server status can be checked is the interval defined for MIB Data Refresh Interval. If you want that interval to be the same as the MIB Data Refresh Interval, enter 1 in the Server Status Check Interval Factor field. Only integer values can be used in this field. The value is multiplied times the MIB Data Refresh Interval to determine the interval at which server status is checked.

- Enter the SNMP community (password) that the manager will be using for requests sent to the WebLogic SNMP agent in the Community Prefix field. The default value is `public`. However, to secure access to the values of the WebLogic attributes, it is recommended that you set Community Prefix to some value other than `public`.

- The integer value in the Debug Level field determines whether internal messages are generated indicating what the agent code is doing. The maximum value is 3. The default is 0, which generates no debug messages.

Click Apply to implement your changes to the SNMP agent configuration. For these changes to actually take effect, you will need to reboot the Administration Server.

CHAPTER

# 2 Trap Notifications

This section discusses the following topics:

- SNMP Trap Format
- Attribute Change Traps
- Log Message Traps
- Monitor Traps

## Overview of WebLogic SNMP Trap Types

A report of the occurrence of an event or crossing of a threshold, sent to an SNMP manager by an SNMP agent, is called a *trap notification*. There are several types of trap notification that the WebLogic SNMP agent software can generate:

- Predefined WebLogic SNMP Traps

  These traps are automatically generated by the SNMP agent when certain predefined conditions occur (if the agent is enabled).

- Attribute Change Traps

  These traps are generated when WebLogic attributes you select change in value.

- Log Message Traps

  These traps are emitted when log messages satisfying criteria you specify are generated on a local WebLogic Server.

- Monitor Traps

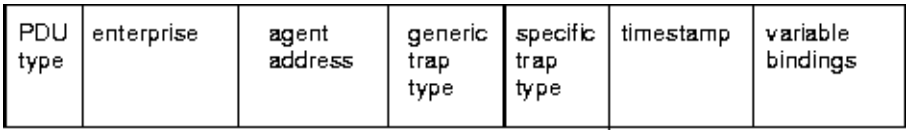
These traps are generated when a Java Management Extension (JMX) monitor that you have created detects the crossing of a threshold, or the occurrence of a specified condition, as defined by you.

This section describes how to set up the WebLogic SNMP agent to generate these various types of trap notification. For information on how to set up the destinations where trap notifications are to be sent, refer to Using SNMP to Manage WebLogic Server.

# SNMP Trap Format

The SNMP standard defines a trap notification sent to a manager as a protocol data unit (PDU) with the fields indicated in Figure 2-1.

**Figure 2-1 SNMP Trap Packet**




The fields have the following meaning:

- `PDU type` identifies the packet as a trap notification.
- `enterprise` is the vendor identification (OID) for the systems/network management subsystem that generated the trap. All traps generated by the WebLogic SNMP agent have the WebLogic OID `.1.3.6.1.4.140.625` in the `enterprise` field.
- `agent address` is the IP address of the node where the trap was generated.
- `generic trap type` is an integer in the range of 0 to 6. Type 6 is an `enterpriseSpecific` trap type, which has no standard interpretation in SNMP. The interpretation of the trap depends upon the value in the specific trap type field, which is defined by a vendor's custom MIB.
- `specific trap type` is a number that further specifies the nature of the event that generated the trap in the case of traps of generic type 6 (`enterpriseSpecific`). For `enterpriseSpecific` traps generated by the

WebLogic SNMP agent, the values in the specific trap type field are those indicated in Table 2-1. These values are defined by the BEA WebLogic MIB.

- `timestamp` is the length of time between the last re-initialization of the agent that issued the trap and the time at which the trap was issued.
- `variable bindings` provide additional information pertaining to the trap.This field consists of name/value pairs. The significance of this field is vendor-specific. The content of the variable bindings in `enterpriseSpecific` traps generated by the WebLogic SNMP agent is determined by the WebLogic MIB definitions (see the WebLogic SNMP MIB Reference). The variable bindings are discussed below.

**Note:** The enterprise OID used by the WebLogic 6.1 SNMP agent differs from the enterprise OID used with the WebLogic 5.1 SNMP agent. The enterprise OID for WebLogic 6.1 is `.1.3.6.1.4.140.625`.

# WebLogic Specific Trap Types

The following table describes the specific trap types for the `enterpriseSpecific` traps generated by the WebLogic SNMP agent.

**Table 2-1 WebLogic Specific Trap Types**

| WebLogic Specific Trap Number | Type of Trap | Meaning |
|---|---|---|
| 60 | Log Message Trap | Generated when a message is logged at a server that matches the user-defined criteria for a sending a log notification trap. |
| 65 | serverStart Trap | Generated when the agent detects that a Managed Server is up that was formerly down. |
| 70 | serverShutDown Trap | Generated when the agent detects that a Managed Server that was up is now down. |
| 75 | Monitor Trap | Generated when a user-defined JMX monitor detects the crossing of a threshold or occurrence of an event, as defined by the user. |

| WebLogic Specific Trap Number | Type of Trap | Meaning |
|---|---|---|
| 80 | Attribute Change Trap | Generated when the agent detects that an attribute selected by the user has changed in value. |

# Predefined WebLogic SNMP Traps

The WebLogic SNMP agent generates the following generic traps automatically:

- `coldStart` trap — This trap has a `generic` trap number of 0 and is generated whenever the Administration Server comes up, if the SNMP Service is enabled.
- `authenticationFailure` trap — This trap has a generic trap number of 4 and is sent to any management station that sends an incorrect SNMP community string. The community string prefix is the actual password and must match the value that you set in the Community Prefix field when configuring the WebLogic SNMP agent in the Administration Console. (See Using SNMP to Manage WebLogic Server for the required format for community strings as sent by SNMP managers.)

All other trap notifications generated by the WebLogic SNMP agent are enterprise-specific traps (generic type 6).

The following enterpriseSpecific trap notifications are also generated automatically by the agent:

- `serverStart` — This trap is generated whenever the SNMP agent detects that a WebLogic Managed Server that was down is now up.

  This trap has a specific type value of 65. The first two name/value pairs in the variable bindings will be the start time and the server name.

- `serverShutDown` — This trap is generated whenever the SNMP agent detects that a WebLogic Managed Server that was up is now down.

  This trap has a specific type value of 70. The first two name/value pairs in the variable bindings will be the down time and the server name.

# Attribute Change Traps

To set up the WebLogic SNMP agent to notify your SNMP agent when a selected WebLogic configuration attribute has changed, do the following:

1. Invoke the Administration Console (if it is not already running).
2. Select SNMP→Traps→SNMP Attribute Changes in the left pane. This invokes the attribute change table. This table lists the filters that you have created to send SNMP traps when the agent detects a change on a selected configuration attribute. There is one filter in the table for each configuration attribute that the agent is monitoring.
3. To create a new attribute change filter, select the Create a new Attribute Change link to invoke the Attribute Change screen. Fill out the fields on this screen as follows:
   - Name — Enter a name for this filter. You might name the filters to suggest the attribute that you are monitoring.
   - Attribute MBean Type — This is the type of the configuration MBean that includes the attribute you wish to monitor.
   - Attribute MBean Name — This is the name of the configuration MBean that has the attribute you wish to monitor.
   - Attribute Name — This is the name of the attribute you wish to monitor.
   - Select the servers on which you want to check for attribute changes.
4. Click Apply to create the new attribute change filter.
5. To activate the new attribute change filter, restart the Administration Server.

An attribute change trap notification includes the following name/value pairs in the variable bindings:

- trapTime — The time at which the trap was generated.
- trapServerName — The name of the Administration Server.
- trapMBeanName — Name of the MBean that includes the attribute.
- trapMBeanType — Type of the MBean that includes the attribute.

- trapAttributeName — Name of the configuration attribute that has changed.
- trapAttributeChangeType — The value can be either `ADD`, `REMOVE`, or `UPDATE`.
- trapAttriruteOldVal — Value of the attribute before the change.
- trapAttributeNewVal — Value of the attribute after the change.

**Note:** Creation of monitors for changes in run-time attributes is not supported. Only attributes in the configuration MIB can be monitored for change of attribute value.

# Log Message Traps

The WebLogic logging subsystem logs messages into a local log at each WebLogic Server. The SNMP agent can register a log message filter on a local server which selects log messages that the agent wants to be notified about. When a log message is generated on the local WebLogic Server that satisfies the filter, a JMX log notification is sent to the agent and the agent generates an SNMP log notification trap.

You can define the log notification filter to select log messages based on the following attributes of the log message:

- Severity level
- Subsystem name
- User ID
- Message ID
- Message substring (a string to search for in the message text)

## Creating a Log Notification Filter

To create a log notification filter, do the following:

1. Invoke the Administration Console (if it isn't running already)

2. Select SNMP→Traps→SNMP Log Filters in the left pane. This invokes the SNMP Log Filter table, which lists all the filters that you have registered with local servers.
3. To create a new log filter, select the Create a new Log Filter link to invoke the SNMP Log Filter screen. You will need to fill out the fields on this screen as follows:
   - Give the new filter a name in the Name field.
   - Select the servers that you wish to have this filter registered with.
   - Select the attributes and attribute values that you wish the local server to use to select log messages that will be used to generate log message traps.
4. Click Apply to create the new log message filter.
5. Restart the Administration Server to activate the new log message filter.

# Variable Bindings in Log Message Traps

The attributes of the log message are passed to the SNMP manager in the variable bindings of the trap. Log notification traps have the following name/value pairs on the variable bindings:

- `trapTime` — Time when the trap is generated.
- `trapServerName` — Name of the local server that generated the log message.
- `trapMachineName` — Name of the machine that the server is running on that generated the log message.
- `trapLogThreadId` — Thread ID from the log message.
- `trapLogTransactionId` — Transaction Id, if any, from the log message. There will only be a transaction Id if the log message occurred in the context of a transaction.
- `trapLogUserId` — The User Id from the log message.
- `trapLogSubsystem` — The subsystem name from the log message.
- `trapLogMsgId` — The log message Id from the log message.

- `trapLogSeverity` — The message severity level from the log message.
- `trapLogMessage` — The text of the log message.

For more information on log messages and the WebLogic Server logging subsystem, see the WebLogic Server Administration Guide.

# Monitor Traps

The WebLogic SNMP agent allows you to configure Java Management Extension (JMX) monitors to poll WebLogic resources at a specified interval to check for the occurrence of conditions or the crossing of thresholds, as defined by you, the user. When a user-defined monitor detects the specified condition, a trap notification is sent to the SNMP manager. This feature allows you to offload polling of WebLogic resources from the SNMP management station to the WebLogic Administration Server.

You can configure three types of JMX monitor:

- **Counter Monitor**
  A counter monitor defines a threshold that is an integer value. A trap is generated if the agent detects that attribute equals or exceeds the threshold value. You can also specify values to add to or subtract from the threshold for subsequent checks of the attribute value.
- **Gauge Monitor**
  A gauge monitor defines a high and a low threshold and generates a trap when the value is equal to or exceeds the high threshold or is equal to or less than the low threshold.
- **String Monitor**
  A string monitor does a compare between a string you provide and the value of the chosen attribute. You can specify that the trap is generated if there is a match between the value and the string you provide, or you can specify that the trap is generated if the value differs from the string you provide.

# Configuring a Counter Monitor

To set up a JMX counter monitor, do the following:

1. Invoke the Administration Console (if it is not already running).
2. Select SNMP→Traps→Monitors→SNMP Counter Monitors in the left pane. This invokes the counter monitor table. This table lists all the counter monitors that you have already configured.
3. To create a new counter monitor, select the Create a new Counter Monitor link to invoke the Counter Monitor screen.
4. On the Counter Monitor page, enter a name for the monitor instance in the Name field.

   BEA Systems recommends that you choose a name that indicates the resource that is being monitored.
5. Enter values in the Monitored MBean Type, Monitored Attribute Name, and (optionally) Monitored MBean Name fields.

   For example, if you want to monitor the `ActiveConnectionsHighCount` attribute of the `JDBCConnectionPoolRuntime` MBean for a JDBC connection pool name MyPool:

   - In the MBean Type field, enter `JDBCConnectionPool Runtime`.
   - In the MBean Name field, enter MyPool.
   - In the Attribute Name field, enter `ActiveConnectionsHighCount`.
6. In the Polling Interval field, enter the frequency in seconds at which you want WebLogic Server to check the attribute's value.

   For testing purposes, consider entering a small value, such as 10.

   A value of 0 means that the monitor never polls the attribute, effectively disabling this monitor.
7. Enter data in the remaining fields as described in the next section, “Typical Configurations for Counter Monitors” on page 2-10.
8. Click Create.
9. Click the Servers tab.

10. From the Available column, select the servers on which you want to monitor the selected attribute. Then click the right arrow to move the selected servers to the Chosen list.

    If you are configuring a monitor for a domain-wide resource, such as a JDBC Connection Pool, select the Administration Server.

11. Click Apply.

12. Restart the Administration Server.

### Typical Configurations for Counter Monitors

The following list describes how to achieve typical configurations of a Counter Monitor instance by entering data on the Counter Monitor page:

- To send a trap when the observed attribute exceeds a threshold, enter a threshold values in the Threshold field.
- To send a trap when the observed attribute exceeds the threshold and then increase the threshold by an offset value, enter a threshold in the Threshold field and an offset value in the Offset field.

  Each time the observed attribute exceeds the new threshold, the threshold is increased by the offset value. For example, if you set Threshold to 1000 and Offset to 2000, when the observed attribute exceeds 1000, the Counter Monitor sends a notification and increases the threshold to 3000. When the observed attribute exceeds 3000, the Counter Monitor sends a notification and increases the threshold again to 5000.
- To specify a maximum value for the threshold, enter a value in the Modulus field. When the threshold reaches the value specified by the modulus, the threshold is returned to the value that was specified through the latest call to the monitor's setThreshold method, before any offsets were applied. For example, if the original Threshold is set to 1000 and the Modulus is set to 5000, when the Threshold exceeds 5000, the monitor sends a notification and resets the Threshold to 1000.

## Configuring a Gauge Monitor

To set up a JMX gauge monitor, do the following:

1. Invoke the Administration Console (if it is not already running).
2. Select SNMP→Traps→Monitors→SNMP Gauge Monitors in the left pane. This invokes the gauge monitor table. This table lists all the gauge monitors that you have already configured.
3. To create a new gauge monitor, select the Create a new Gauge Monitor link to invoke the Gauge Monitor screen. You will need to fill out the fields on this screen as follows:
   - Name — Enter a name for this monitor. You might want to choose a name that gives some idea of what it is monitoring.
   - Monitored MBean Type — This is the type of the MBean that includes the attribute that you wish to monitor.
   - Monitored MBean Name — This is the name of the MBean that includes the attribute that you want to monitor.
   - Monitored Attribute Name — The name of the attribute that you want to monitor.
   - Polling Interval — This is the frequency in seconds that the agent is to check the attribute value.
   - Threshold High — A trap will be generated if the attribute value is equal to or greater than the integer value you enter here.
   - Threshold Low — A trap will be generated if the attribute value is equal to or less than the integer value you enter here.
   - Enabled Servers — Select the servers on which you want to monitor the selected attribute.
4. Select Apply to create the new gauge monitor.
5. Restart the Administration Server to activate the new monitor.

# Configuring a String Monitor

To set up a JMX string monitor, do the following:

1. Invoke the Administration Console (if it is not already running).

2. Select SNMP→Traps→Monitors→SNMP String Monitors in the left pane. This invokes the string monitor table. This table lists all the string monitors that you have already configured.
3. To create a new string monitor, select the Create a new String Monitor link to invoke the String Monitor screen. Fill out the fields on this screen as follows:
   - Name — Enter a name for this monitor. You might want to choose a name that gives some idea what it is monitoring.
   - String To Compare — This is the string that the monitor will compare to the attribute value to determine whether to generate a trap.
   - Notify Differ — If checked, a trap is generated if the value of the attribute differs from the value entered in the String To Compare field.
   - Notify Match — If checked, a trap is generated if the value of the attribute matches the value you entered in the String To Compare field.
   - Monitored MBean Type — This is the type of the MBean that includes the attribute that you wish to monitor.
   - Monitored MBean Name — This is the name of the MBean that includes the attribute that you want to monitor.
   - Monitored Attribute Name — The name of the attribute that you want to monitor.
   - Polling Interval — This is the frequency in seconds that the agent is to check the attribute value.
   - Enabled Servers — Select the servers on which you want to monitor the attribute specified in the Monitored Attribute Name field.
4. Select Apply to create the new string monitor.
5. Restart the Administration Server to activate the new string monitor.

# Variables Included with Monitor Trap

A JMX monitor polls for a specified threshold or condition and the agent generates a monitor trap when the specified threshold is crossed, or the specified condition occurs. The WebLogic SNMP agent includes the following name/value pairs in the variable bindings of each monitor trap:

- `trapTime` — The time at which the trap was generated.
- `trapServerName` — The local server whose attribute value generated the trap.
- `trapMonitorType` — Either `CounterMonitor`, `StringMonitor`, or `GaugeMonitor`.
- `trapMonitorThreshold` — An ASCII representation of the threshold that triggered the trap.
- `trapMonitorValue` — An ASCII representation of the value that triggered the trap.
- `trapMBeanName` — The name of the MBean that contained the attribute being monitored.
- `trapMBeanType` — The type of the MBean that contained the attribute being monitored.
- `trapAttributeName` — The name of the attribute whose value triggered the trap.

# Disabling Trap Generation

When you create an entry for a particular type of trap such as a log filter trap or JMX monitor trap, generation of such traps is only activated once the Administration Server is restarted. However, for any trap request that you have created, you can de-activate the trap generation dynamically via the Administration Console (or the `weblogic.Admin` command line interface).

When you enable trap generation for a particular type of trap, you create an entry in the table for that type of trap that is displayed in the Administration Console. To de-activate that trap, simply delete the entry in the trap table. Thus, if you have created a JMX counter monitor to poll for a specified condition, you can turn off that monitor by deleting the entry for that counter monitor in the table at SNMP→Traps→Monitors→SNMP Counter Monitors.

# 3 Using Multiple SNMP Agents

This section discusses the following topics:

- SNMP Agent as Proxy for Other Agents
- Configuring an SNMP Proxy

The original SNMP management model allowed for only a single, monolithic agent to carry out all management responsibilities on a given network node (IP address). This solution was not flexible enough to provide for effective management of increasingly complex systems. In addition to the agents typically provided by computer manufacturers for hardware and operating system information, agents are also produced by vendors of other products, such as agents for SQL database systems. Complex and heterogeneous systems thus require the ability to accommodate multiple agents on a single network node.

## SNMP Agent as Proxy for Other Agents

This weakness of the original SNMP model led to the concept of an SNMP master agent that acts as a proxy for other SNMP agents. The WebLogic SNMP agent can function as a master agent in this sense. To use the master agent functionality of the WebLogic SNMP agent, you can assign branches of the registration tree (OID tree) as the responsibility of other SNMP agents. Each of these will be a branch that encompasses the private MIB (or some part of that MIB) which the target agent is designed to manage.

**Note:** You cannot use the WebLogic SNMP agent as a proxy for SNMP agents in other WebLogic Server domains. For example, WebLogic domainA's SNMP agent cannot proxy requests to domainB's SNMP agent. This limitation is in effect because all WebLogic SNMP agents use the same MIB root.

Instead of proxying requests to multiple WebLogic Server domains, you can place all of your server instances in a single domain and send requests directly to each Managed Server. See "Specifying the Target Server in Management Requests" on page 1-5.

The WebLogic SNMP agent listens for requests from SNMP managers and then fans out these requests to other SNMP agents on the Administration Server machine, if the attribute requested has an OID falling under the branch of the OID tree assigned to one of those other agents. By default the WebLogic SNMP agent listens for management requests on port 161. If the WebLogic SNMP agent is to proxy for other SNMP agents, then those other agents must be configured to listen for SNMP management requests on a port other than the port that the WebLogic SNMP agent is using to receive requests from SNMP managers.

# Configuring an SNMP Proxy

To configure the WebLogic SNMP agent to proxy for another SNMP agent, do the following:

1. Invoke the Administration Console (if it isn't running already).
2. Select SNMP→SNMP Proxies in the left pane. This invokes the SNMP Proxies table, which lists entries for all the SNMP agents you have configured the WebLogic SNMP agent to proxy for.
3. To create a new proxy, select the Create a new SNMP Proxy link to invoke the SNMP Proxy configuration screen. Fill out the fields on this screen as follows:
   - Name — Enter a name for the proxy in this field. This should be descriptive of the agent that the requests will be forwarded to, such as "OracleDBAgent."
   - Port — Enter a port number for communication with the other SNMP agent. The agent being proxied for must be configured to expect SNMP

management requests on this port number. This must be a port number other than the port being used by the WebLogic SNMP agent for communication with SNMP managers.

- OID Root — This is an absolute OID that designates the root, or top node, of the part of the OID tree being assigned to that agent.
- Community — This is the community name that the other agent expects in requests from SNMP managers.
- Timeout — This is the interval, in seconds, that the WebLogic SNMP proxy agent waits for a response to requests forwarded to another SNMP agent. If this interval elapses without a response from the other agent, the WebLogic SNMP agent will send an appropriate error to the requesting manager.

4. Click Apply to create the new proxy.
5. Restart the Administration Server so that your changes can take effect.

# A Sources of SNMP Information

This appendix lists sources of additional information about Simple Network Management Protocol, including the following:

- Reference Books
- Standards and Drafts
- Obtaining RFCs

## Reference Books

If you need additional information about MIBs, agents, or the SNMP protocol, refer to these books:

- Comer, Douglas; Internetworking with TCP/IP, Vol. 2; Prentice-Hall, Englewood Cliffs, New Jersey, 1991
- Leinwand, Allan and Fang, Karen; Network Management: A Practical Perspective; Addison-Wesley, Reading, Massachusetts, 1993
- Rose, Marshall T.; The Simple Book: An Introduction to Management of TCP/IP-based Internets; Prentice-Hall, Englewood Cliffs, New Jersey, 1991
- Rose, Marshall T.; The Open Book: A Practical Perspective on Open Systems Interconnection; Prentice-Hall, Englewood Cliffs, New Jersey, 1989

- Miller, Mark; Managing Internetworks with SNMP, M & T Books
- Stallings, William; SNMP, SNMPv2 and CMIP: The Practical Guide to Network Management Standards, Addison-Wesley, Reading, Massachusetts, 1993

# Standards and Drafts

The SNMP protocol has been defined through a series of Requests for Comments (RFCs). The following standards and drafts are available.

**Figure 3-1 SNMP RFCs**

| RFC Number | Description |
|---|---|
| 052 | IAB Recommendations |
| 1089 | SNMP over Ethernet |
| 1109 | Ad-hoc Review |
| 1155 | Structure of Management Information |
| 1156 | Management Information Base (MIB-I) |
| 1157 | SNMP Protocol |
| 1161 | SNMP over OSI |
| 1187 | Bulk table retrieval |
| 1212 | Concise MIB definitions |
| 1213 | Management Information Base (MIB-II) |
| 1214 | OSI MIB |
| 1215 | Traps |
| 1227 | SNMP Multiplex (SMUX) |

| RFC Number | Description |
|---|---|
| 1228 | SNMP-DPI |
| 1229 | Generic-interface MIB extensions |
| 1230 IEEE 802.4 | Token Bus MIB |
| 1231 IEEE 802.5 | Token Ring MIB |
| 1239 | Reassignment of MIBs |
| 1243 | AppleTalk MIB |
| 1248 | OSPF MIB |
| ISO 8824 | ASN.1 |
| ISO 8825 | BER for ASN.1 |

# Obtaining RFCs

You can obtain Requests for Comments in the following ways:

- Download them from almost anywhere on the Internet
- Obtain them from SRI International

  Mailing Address: SRI International, EJ291, DDN Network Information Center, 333 Ravenswood Ave., Menlo Park CA 94025

  Phone: +1.800.235.3155

  e-mail: MAIL-SERVER@nisc.sri.com. Leave the subject field blank. In the body, enter: SEND RFCnnnn.TXT-1

  FTP: ftp://ftp.nisc.sri.com/rfc/rfcNNNN.txt