

8

Monitoring a Domain

ORACLE®

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to:

- Configure and access WebLogic Server logs
- Enable WebLogic Server debugging output
- Monitor WebLogic Server health and performance
- Monitor JDBC data sources
- Access diagnostic charts in the Monitoring Dashboard



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

WebLogic Server Logs

The subsystems of an instance of WebLogic Server publish information about themselves into logs.

Log	Description
Server log	Used by server subsystems to record events
Standard out	Some server log messages are printed to standard out.
Domain log	Some server messages are gathered by the administration server for inclusion into the domain-wide log.
Access log	Used by the HTTP subsystem to track HTTP communication
Audit log	Tracks security requests. Requires configuring an Auditing provider (not configured by default).



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Each WebLogic Server instance writes the messages from its subsystems and applications into a server log file that is located on the local host computer.

In addition to writing messages to a log file, each server instance prints a subset of its messages to standard out. Usually, standard out is the shell (command prompt) in which you are running the server instance. However, some operating systems enable you to redirect standard out to some other location. By default, a server instance prints only messages of a NOTICE severity level or higher to standard out.

Each server instance also forwards a subset of its messages for the administration to collect and place in the domain-wide log file. By default, servers forward messages of severity level NOTICE or higher. While you can modify the type of messages that are forwarded, servers can never forward messages of the DEBUG severity level.

The HTTP subsystem keeps a log of all HTTP transactions in a text file. The default location and rotation policy for HTTP access logs is the same as the server log. You can set the attributes that define the behavior of HTTP access logs for each server.

WebLogic Server Logs

Log	Description
Transaction log	<ul style="list-style-type: none">Contains information about transactions being managed by WebLogic ServerIs used by that server when recovering from crashesIs in binary format
JMS Server log	<ul style="list-style-type: none">Is enabled when a JMS Server is createdMessage destinations must be specifically enabled.Contains information on basic message lifecycle events



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

The WebLogic Auditing provider records information from a number of security requests, which are determined internally by the WebLogic Security Framework. The WebLogic Auditing provider also records the event data associated with these security requests, and the outcome of the requests. Configuring an Auditing provider is optional. The default security realm does not have an Auditing provider configured.

Each server has a transaction log, which stores information about committed transactions managed by the server that may not have been completed. WebLogic Server uses the transaction log when recovering from system crashes or network failures. You cannot directly view the transaction log; the file is in a binary format. The Transaction Manager uses the default persistent store to store transaction log files. You can change where the default store is located.

JMS logging is enabled by default when you create a JMS Server, however, you must specifically enable it on message destinations in the JMS modules targeted to this JMS server (or on the JMS template used to create the destinations). JMS server log files contain information on basic message lifecycle events, such as message production, consumption, and removal. When a JMS destination is configured with message logging enabled, then each of the basic message lifecycle events generate a message log event in the JMS message log file.

WebLogic Server Log Locations

Directory	Description
<code>domainname</code>	
<code>servers</code>	
<code>AdminServer</code>	Admin server (named <i>AdminServer</i>) directory
<code>logs</code>	
<code>AdminServer.log</code>	The server log file for <i>AdminServer</i>
<code>domainname.log</code>	The domain log
<code>server1</code>	Directory for managed server named <i>server1</i>
<code>logs</code>	
<code>server1.log</code>	The server log file for <i>server1</i>
<code>access.log</code>	HTTP subsystem log
<code>jmsServers</code>	
<code>jmsservername</code>	
<code>jms.messages.log</code>	JMS lifecycle events of the JMS Server called <i>jmsservername</i> created on <i>server1</i>



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Note that the administration server also has an `access.log` file, which is not shown here. The auditing log location is also not shown. If the WebLogic Auditing provider is configured, the audit log is here:

`domainpath/domainname/servers/servername/logs/
DefaultAuditRecorder.log`.

The location of the transaction log (also not shown) in the default store is:

`domainpath/domainname/servers/servername/data/store/default/_WLS_SERVERNAMExxxxxx.DAT`. (Where `xxxxxx` is a generated count.)

The JMS subsystem is enabled once JMS Servers are created. There is a log file for each JMS Server.

Log Message Severity Levels

Severity levels from low to high impact:

Severity	Description
TRACE	Used for messages that are part of WebLogic Diagnostic Framework
DEBUG	Messages from enabled "debug flags"
INFO	Normal operation information
NOTICE	More important operation information
WARNING	Something suspicious occurred, but it might not affect normal operation.
ERROR	A user level error has occurred, but the system or application can handle it with no interruption and limited degradation of service.



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

There is more on debug flags later in this lesson.

Log Message Severity Levels

Severity	Description
CRITICAL	A system or service level error has occurred. The system can recover, but there may be momentary loss or permanent degradation of service.
ALERT	A particular service is unusable, while other parts of the system still function. Automatic recovery is not possible. Immediate attention of an administrator is needed.
EMERGENCY	The server is unusable. This indicates a severe system failure.

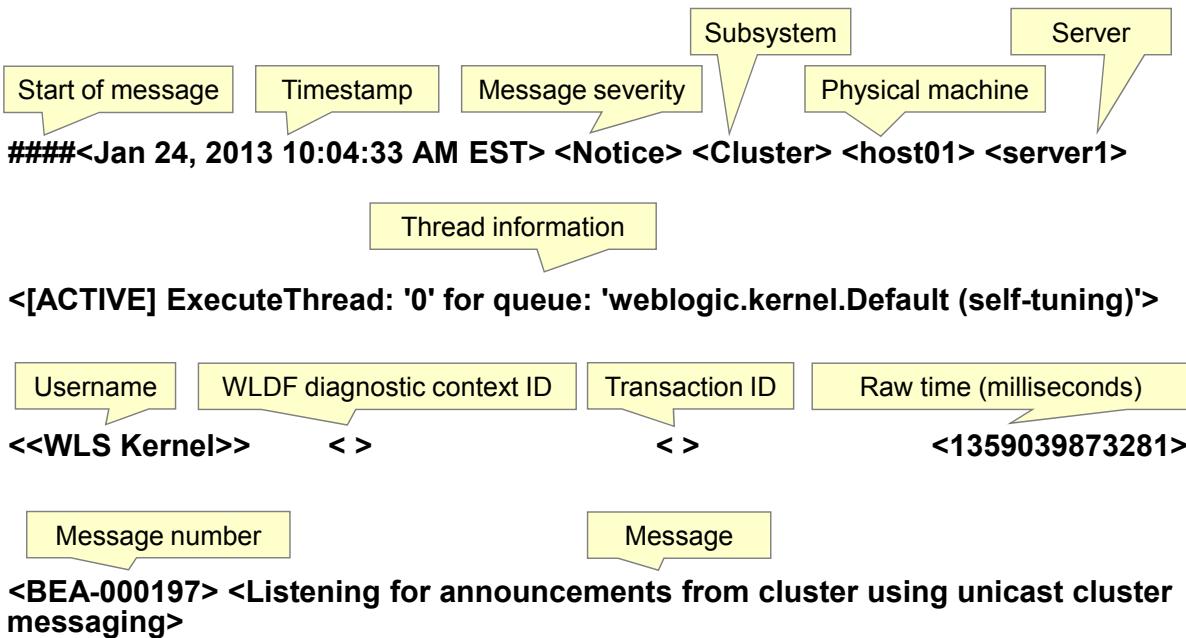


ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Understanding Log File Entries

Log message format:



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Log files can be viewed in any text editor.

In this example, extra blanks are placed in the log entry for readability. The username being used is WLS Kernel, an internal ID used by the server itself. Also, there is no WebLogic Diagnostic Framework (WLDF) context ID (it is blank), nor a transaction ID (also blank).

Accessing the Logs from the Admin Console

The first screenshot shows the 'Domain Structure' tree on the left with 'wlsadmin' selected. Under 'wlsadmin', 'Diagnostics' is expanded, and 'Log Files' is highlighted with a green circle labeled '1'. A mouse cursor is hovering over 'Log Files'. The main panel shows a table titled 'Log Files' with one row: 'ServerLog' (Type: Server Log, Server: AdminServer). A green circle labeled '2' is on the 'View' button next to 'ServerLog'. The second screenshot shows the 'Server Log Entries' table with one entry: Date: Jan 24, 2013 3:40:24 PM EST, Subsystem: Health, Severity: Info, Message ID: BEA-310002, Message: 12% of the total memory in the server is free. A green circle labeled '3' is on the 'View' button next to the log entry.

Name	Type	Server
ServerLog	Server Log	AdminServer

Date	Subsystem	Severity	Message ID	Message
Jan 24, 2013 3:40:24 PM EST	Health	Info	BEA-310002	12% of the total memory in the server is free.

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

To access logs from the WebLogic Server administration console:

1. Under Domain Structure, select **Diagnostics** and then **Log Files**.
2. Select the log of interest and click **View**.
3. View log entries. To see more details, select an entry and click **View**.

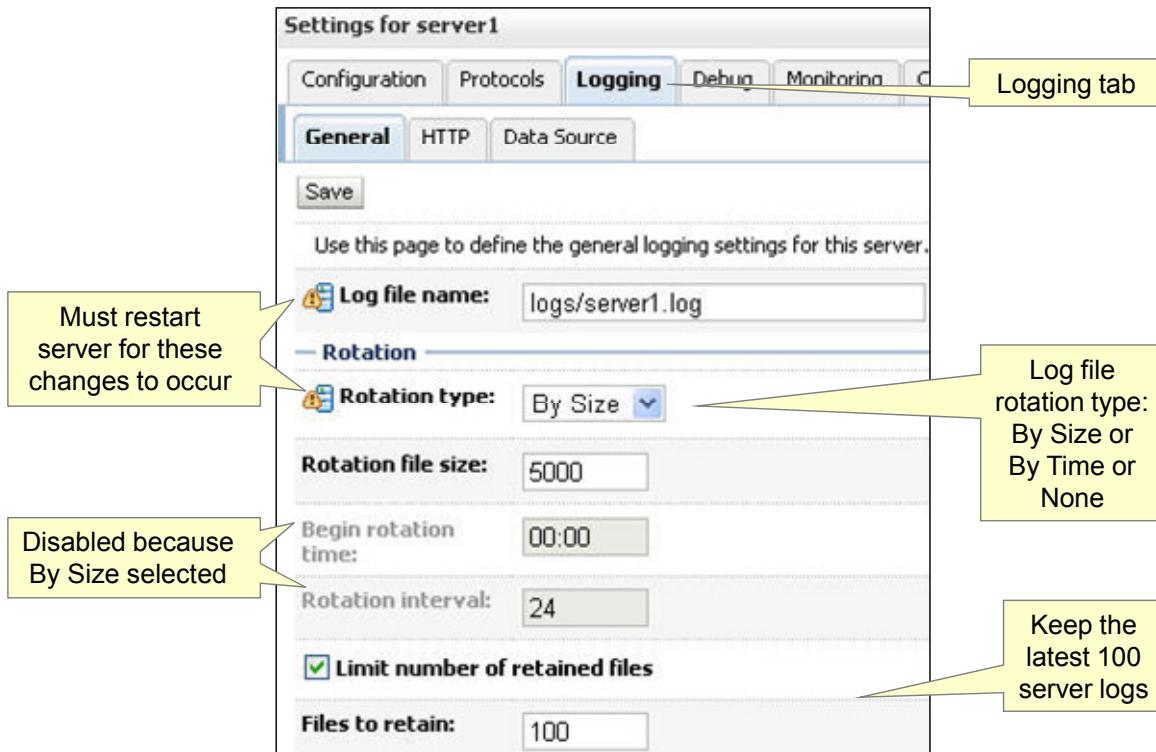
The logs available for viewing in the administration console are:

- Server logs
- Domain logs
- HTTP access logs
- JMS Server logs

The logs available for viewing in the administration console are:

- WebLogic Diagnostic Framework (WLDF) diagnostic logs:
 - **Data source profile log:** Data source diagnostic data that in earlier versions of WebLogic Server was written to the events data archive. What is contained in the data source profile log is configured at the data source level. In the administration console, select the particular data source and then the **Configuration > Diagnostics** tabs. The configuration of the log file itself is done at the server level. In the administration console, select the particular server and then click the **Logging > Data Source** tab. The default location and name of the file is:
domainpath/domainname/servers/servername/logs/datasource.log
 - **Events data archive:** Diagnostic data from WLDF instrumentation. WLDF instrumentation is a mechanism for adding diagnostic code to instances of WebLogic Server or applications to trigger actions at specific code locations and record data from those actions in the archive. This data is placed in the WLDF archive (see below).
 - **Harvested data archive:** Diagnostic data from a WLDF “harvest.” WLDF data can be collected by a WLDF artifact called a data harvester, which is configured to periodically collect diagnostic data and store it in this archive. Harvesters are configured in a WLDF diagnostic module. The default location of the WLDF archive file is:
domainpath/domainname/servers/servername/data/store/
diagnostics/WLS_DIAGNOSTICSxxxxxx.DAT
(Where *xxxxxx* is a generated count.)

Configuring Server Logging



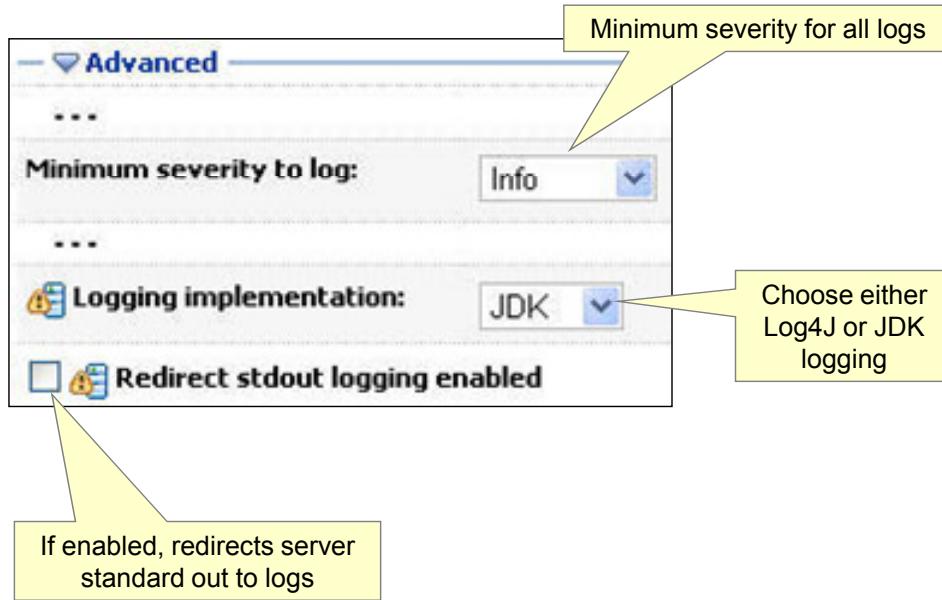
ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

After you click **Lock & Edit** in the Change Center, in the Domain Structure of the administration console, expand **Environment** and click **Servers**. In the Servers table, click the name of the server you want to configure. Click the **Logging > General** tab. The available options include:

- **Log file name:** The name of the file that stores the server log. The default is to place the file in the `log` directory under the server directory and name it the server's name `.log`. If you change it and specify a relative path, it is relative to the server's main directory.
- **Rotation type**
 - **None:** Messages accumulate in a single file. You must erase the contents of the file when it grows too large. Note that Oracle WebLogic Server sets a threshold size limit of 500 MB before it forces a hard rotation to prevent excessive log file growth.
 - **By Size:** When the log file reaches the size that you specify in "Rotation file size," the server renames the file to `servername.lognnnnn`.
 - **By Time:** At each time interval that you specify in "Begin rotation time" and "Rotation interval," the server renames the file to `servername.lognnnnn`.
- **Limit number of retained files:** After the server reaches the **File to retain** limit, it deletes the oldest log file and creates a new log file with the latest suffix.

Configuring Server Logging



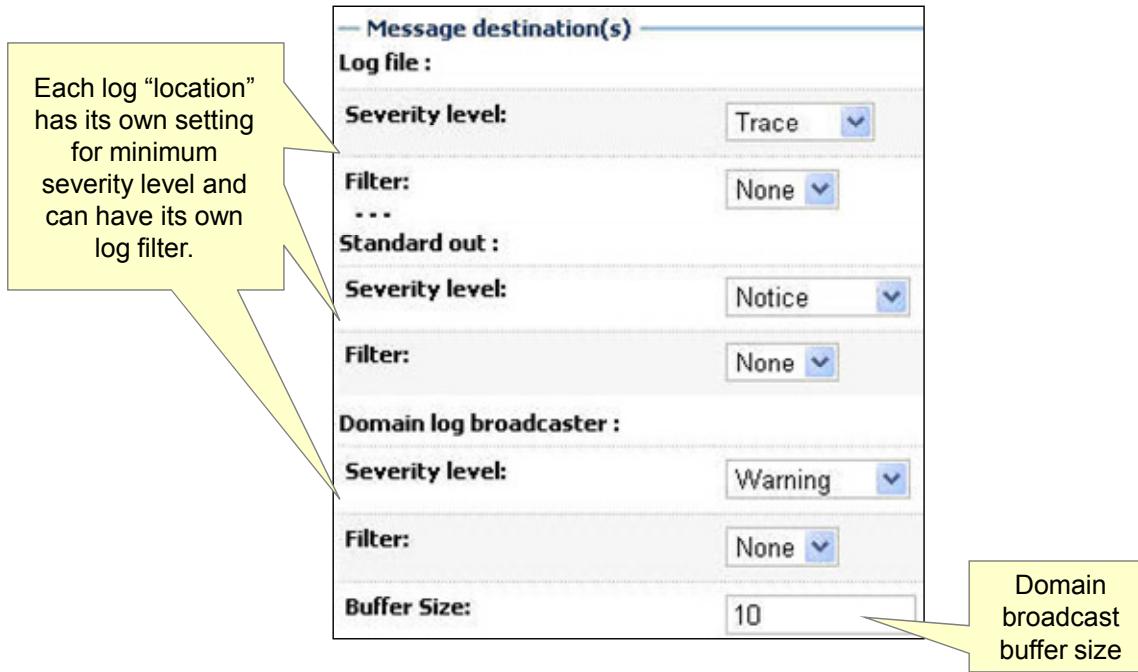
ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

The available Advanced options include:

- **Minimum severity to log:** The minimum severity of log messages going to all log destinations. By default all messages are published.
- **Logging implementation:** Specifies whether the server logging is based on a Log4J implementation or the default, the logging based on the Java Logging APIs in the JDK
- **Redirect stdout logging enabled:** When enabled, redirects the standard out of the JVM in which a WebLogic Server instance runs to the logging system

Configuring Server Logging



ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

The available Advanced options include:

- **Log file:** The server log attributes:
 - **Severity level:** The minimum severity of log messages for this log. By default all messages go.
 - **Filter:** Specifies the filter configuration for this log. A filter configuration defines simple filtering rules to limit the volume of log messages written.
- **Standard out:** The standard out attributes:
 - **Severity Level** and **Filter**. (Same explanations as before.) The default severity for standard out is NOTICE.
- **Domain log broadcaster:** The domain log (from this server) attributes:
 - **Severity Level** and **Filter**. (Same explanations as before.) The default severity for the domain log is also NOTICE.
 - **Buffer Size:** Specifies the size of the buffer for the log messages that are sent to the domain log. The buffer is maintained on the Managed Server and is broadcast to the domain log when it gets full. If you notice performance issues due to a high rate of log messages being generated, set this value higher.

Error Messages Reference

Use the WebLogic Server online Error Messages Reference document to obtain more information about a specific log message based on its ID.

Oracle® WebLogic Server Error Messages Reference
12c Release 1 (12.1.2)
Part Number E26117-01

[Home](#) [Book List](#) [Contents](#) [Contact Us](#)

[View PDF](#)

[Previous](#)

1 BEA-000001 to BEA-2160002

BEA-000001: Server must be started by Node Manager when consensus leasing is enabled.
Cause: The server was not started by Node Manager.
Action: Start the server using Node Manager.
Level: 1
Type: INTERNAL_ERROR
Impact: ConsensusLeasing



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

For a detailed description of log messages if you only have a message number, use the online document titled *Oracle WebLogic Server Error Messages Reference*.

Log Filters

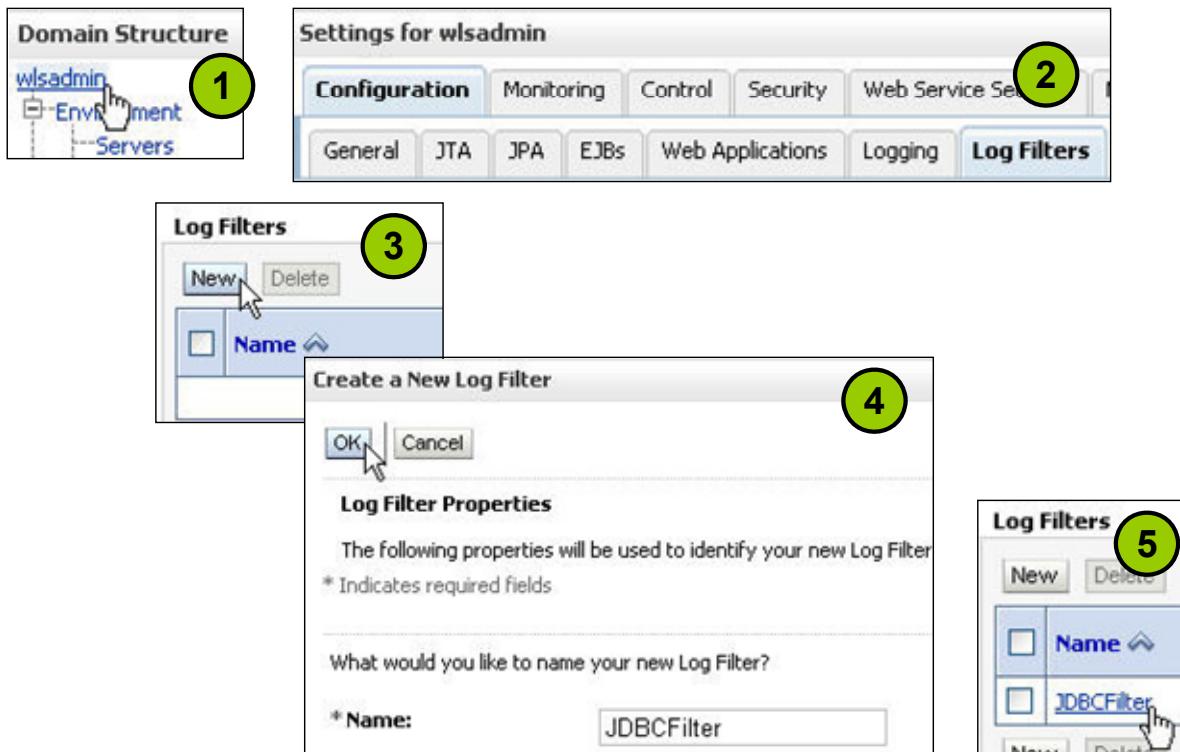
- Provide finer tuned control of the log messages that are published
- Are based on the values of message attributes
- Are created at the domain level
- Can be applied to different log message destinations:
 - Server log
 - Server log memory buffer
 - Server standard out
 - Domain log broadcaster



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Log filters provide control over the log messages that are published. A filter uses custom logic to evaluate the log message content and accept or reject a message. For example, you can filter messages of a certain severity level from a particular subsystem. Only the log messages that satisfy the filter criteria are published. You can create separate filters for the messages that each server instance either writes to its server log file, standard out, memory buffer, or broadcasts to the domain-wide log.

Creating a Log Filter

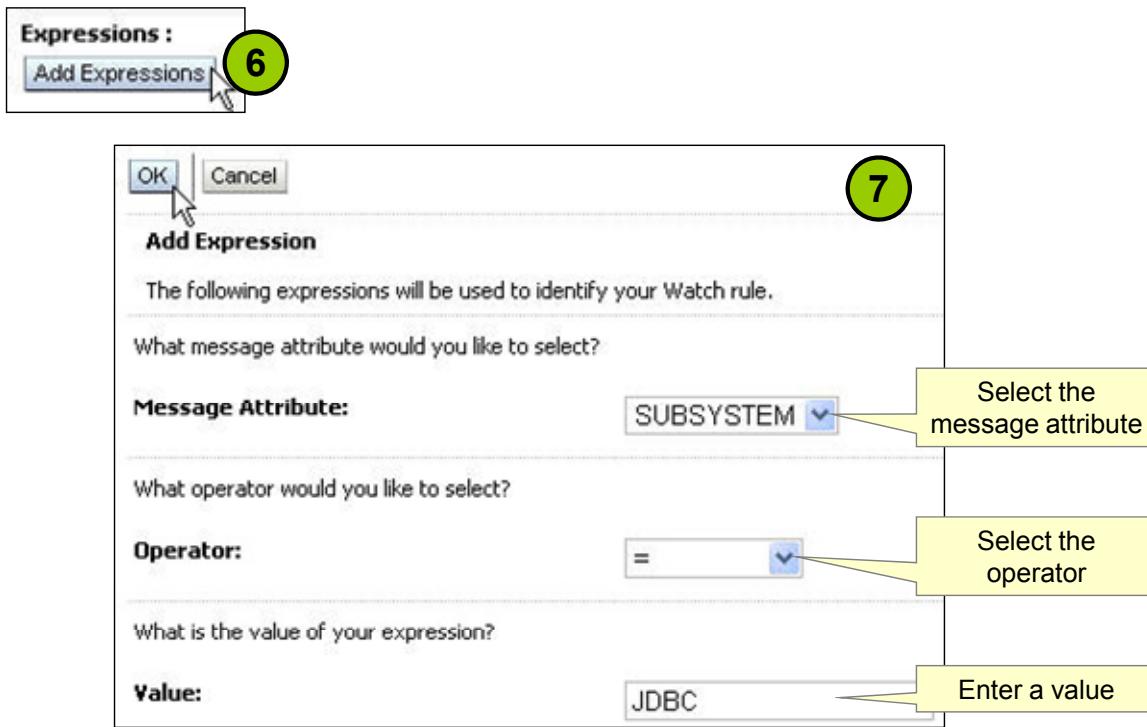


ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

1. After locking the configuration, in the Domain Structure, select the name of the domain.
2. Click the **Configuration > Log Filters** tab.
3. Click the **New** button.
4. On the Create a New Log Filter page, enter a name for the filter and click **OK**.
5. The new log filter appears in the Log Filters table. To configure a filter expression, click the log filter name.

Creating a Log Filter



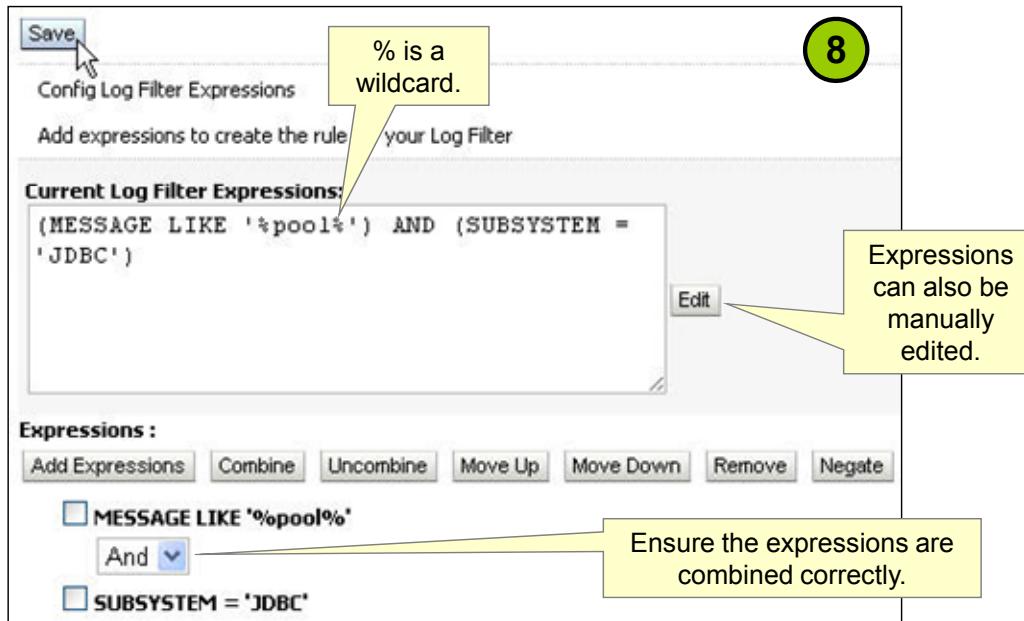
ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

6. On the Configuration page, click the **Add Expressions** button.
7. Specify an expression as part of the criteria for qualifying messages. A filter expression defines simple filtering rules to limit the volume of log messages that are written to a particular log destination. Select a Message Attribute, an Operator, and a Value. Then click **OK**. Continue to click the **Add Expressions** button and saving expressions until you have entered all the expressions needed.

Note: The names of the subsystems can be found by looking through the “Impact” values in the error messages in *Oracle WebLogic Server Error Messages Reference*.

Creating a Log Filter



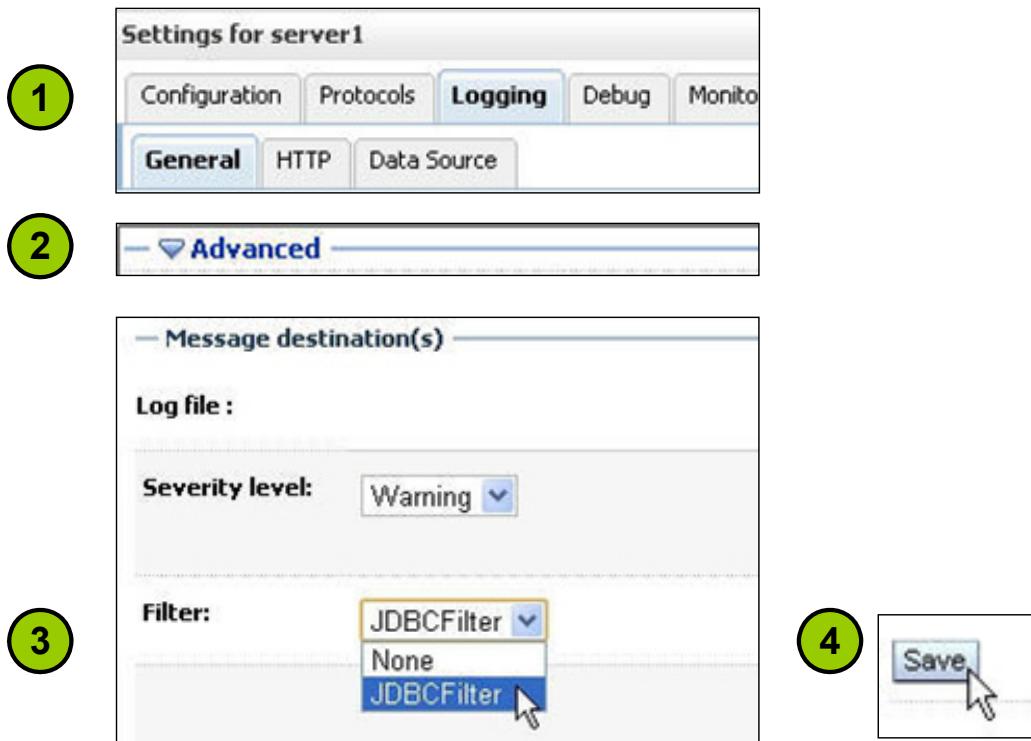
ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

8. After the expressions have been entered, ensure that the Boolean operators that connect them are correct. Use the drop-down list to choose **And** or **Or**. Then click the **Save** button. Finally, click **Activate Changes** in the Change Center.

Log filter expressions are WLDF queries. For more information about the WLDF Query Language see the appendix titled “WLDF Query Language” in the *Configuring and Using the Diagnostics Framework for Oracle WebLogic Server* document.

Applying a Log Filter



ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

1. After the configuration has been locked, in the Domain Structure, expand **Environment** and select **Servers**. In the Servers table, click the name of the server that you want to configure. Click the **Logging > General** tab.
2. Click **Advanced**.
3. Under the **Message destination(s)** section, specify an existing filter for messages going to any of the four log message destinations:
 - Log file (the server log, being filtered in the slide)
 - Standard out
 - Domain log broadcaster
 - Memory buffer
4. Click the **Save** button. Then, in the Change Center, click **Activate Changes**.

Subsystem Debugging

- Various WebLogic Server subsystems have the ability to generate detailed log messages to facilitate debugging.
- You can enable debugging on specific servers for individual subsystems.

Settings for server1

Debug settings for this Server

	Debug Scopes and Attributes	State
<input type="checkbox"/>	default	Disabled
<input type="checkbox"/>	weblogic	Disabled
<input checked="" type="checkbox"/>	application	Enabled

ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

1. After the configuration has been locked, in the Domain Structure, expand **Environment** and select **Servers**. In the Servers table, click the name of the server you want to configure.
2. Click the **Debug** tab.
3. Select one or more available debugging scopes by using the supplied check boxes. Then click **Enable** or **Disable**. For convenience, a **Clear** button is also provided to deselect the debug scopes or attributes that are currently selected.

Debug Scopes

- Debug flags (attributes) for WebLogic Server subsystems are organized into *scopes*.
 - You can enable entire debug scopes or individual attributes.
 - When a parent scope is enabled, all child scopes and attributes are also enabled, unless they are overridden.

<input type="checkbox"/>	weblogic	Disabled
<input checked="" type="checkbox"/>	application	Enabled
<input checked="" type="checkbox"/>	classloader	
<input type="checkbox"/>	cluster	Enabled
<input type="checkbox"/>	leasing	Enabled (Inherited)
<input type="checkbox"/>	databaseless	Enabled (Inherited)
<input type="checkbox"/>	DebugConsensusLeasing	Enabled (Inherited)

Scopes When in doubt about which scope or attribute to choose, enable a parent category.

Attribute

The Oracle logo is displayed on a red horizontal bar.

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Changing debugging is “dynamic,” in that you can enable debugging while a server is running. Many debug flags can also be set as command-line arguments when starting a server. Some examples:

- -Dweblogic.debug.DebugCluster=true (cluster debugging)
- -Dweblogic.debug.DebugJDBCSQL=true (SQL debugging)

Debug Scopes: Examples

Subsystem	Scopes (weblogic.*)
JDBC	jdbc.connection, jdbc.internal, jdbc.sql
Cluster	core.cluster
Deployment	deploy, ejb.deployment
Applications	application.library, ejb.caching, ejb.invoke, ejb.pooling, servlet, servlet.internal, servlet.internal.session
Transactions	transaction.recovery, transaction.twopc, transaction.xa
Security	security, security.ldap, security.ssl

All “debug scope” messages are the DEBUG severity level, so ensure the log location severity level is set appropriately.



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

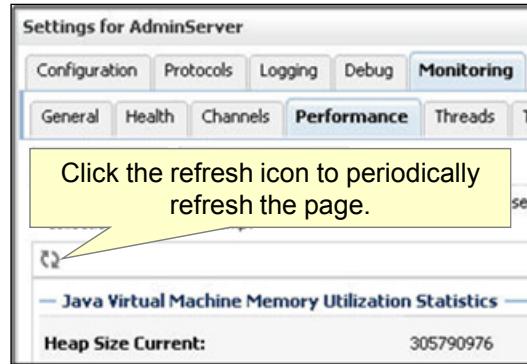
Some examples of attributes under a scope:

- **weblogic.jdbc.connection.DebugJDBCCConn**: Traces all connection reserve and release operations in data sources as well as all application requests to get or close connections
- **weblogic.jdbc.sql.DebugJDBCSQL**: Prints information about all JDBC methods invoked, including their arguments and return values, and thrown exceptions
- **weblogic.core.cluster.DebugCluster**: Prints information about basic cluster lifecycle events

Admin Console: Monitoring Domain Resources

The administration console can monitor domain resources:

- Servers
- Clusters
- Machines
- Deployments
- JDBC data sources
- And more



Use the Domain Structure to locate the type of resource. Select a particular instance. Then click the **Monitoring** tab. The **Monitoring** tab of some elements have subtabs.

- When data is displayed in a table, use the **Customize this table** link to modify the columns displayed.

ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

- **Servers:** The subtabs under the Monitoring tab are: General (state, version, OS, and so on), Health, Channels (statistics on hosts/ports), Performance (Java memory information), Threads (Java thread information), Timers (internal timers used by the server), Workload (information on Work Managers), Jobs (Job Scheduler information), Security (invalid logins and locked out users), Default Store (information on the default persistent store used by JMS, JTA, WLDF, and others), JMS (information on JMS connections and servers), SAF (information on Store and Forward agents), JDBC (data sources), and JTA (transactions).
- **Clusters:** Information about the servers in the cluster, how often they drop out of the cluster, failover data, and more
- **Machines:** Monitoring of the Node Manager under the machine
- **Deployments:** The state and other information about deployed applications
- **JMS destinations (queues and topics):** Information about messages on the destination
- **JDBC data sources:** Information about connections and capacity

If a monitoring page has a refresh icon, click it to have the administration console periodically poll the resource and update the display. To change the rate at which this occurs, update the Refresh Interval under the admin console's User Preferences.

Monitoring the Domain

The domain itself has a **Monitoring** tab, which can show you an overview of the domain's health, servers, clusters, and migration.

Settings for wlsadmin

Configuration Monitoring Control Security Web Service Security

Health Servers Clusters Migration

This page allows you to monitor health information for this domain.

Health: All Filter

Health Information

Server/Subsystem Name	State	Health	Reason
AdminServer	RUNNING	OK	
server1	RUNNING	OK	
server2	RUNNING	OK	
server3	SHUTDOWN	Not reachable	

ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

To access domain monitoring, in the Domain Structure, click the domain name, and then click the **Monitoring** tab. Its subtabs are **Health**, **Servers**, **Clusters**, and **Migration**.

The options for the Health filter are:

- **All:** Show all servers.
- **OK:** Show only servers that are functioning without any problems.
- **Warning:** Show servers that have issued warnings and that might have problems in the future.
- **Overloaded:** Show servers that are overloaded; these servers have gone below their free memory threshold percentage. (This threshold is set for a server in the admin console under the **Configuration > Overload** tabs. The field is called “Free Memory Percent Low Threshold.”)
- **Critical:** Show servers that are about to fail. Action must be taken immediately or a server in this state will soon fail.
- **Failed:** Show servers that have failed. A failed server’s Health will display as “Not reachable.”

Monitoring All Servers

The servers table in the admin console lists all the servers in a domain. The information displayed can be customized so you can use this table to see the information important to you.

Servers (Filtered - More Columns Exist)

Click the **Lock & Edit** button in the Change Center to activate all the buttons on this page.

	Name	Cluster	Machine	State	Health	Heap Free Current	Heap Size Current
<input type="checkbox"/>	AdminServer(admin)		machine1	RUNNING	OK	165433584	342294528
<input type="checkbox"/>	server1	cluster1	machine1	RUNNING	OK	173135168	349372416
<input type="checkbox"/>	server2	cluster1	machine2	RUNNING	OK	245126888	331153408

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

To access the servers table, in the Domain Structure, expand **Environment** and select **Servers**. The servers table is on the **Configuration** tab.

The data in the servers table can be customized (as many tables can in the administration console). Click **Customize this table** and you can filter the data displayed and adding or subtracting the attributes that are shown. You can also sort the data in the tables by clicking the column headers.

Monitoring Server Health

The admin console server health monitoring page shows the state of the server's subsystems and deployments.

Settings for server1

Configuration Protocols Logging Debug Monitoring

General Health Channels Performance Threads T

Use this page to monitor health information for this server.

Server Health: OK

Reason: (No value specified)

Health information details

Showing 1 to 10 of 11 Previous | Next

Subsystem	Health	Reason
JDBC	✓ OK	
JTA	✓ OK	

ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

To access the server health information, in the Domain Structure, expand **Environment** and select **Servers**. In the servers table, select the server of interest by clicking its name. Click the **Monitoring > Health** tab.

Monitoring Server Performance

The admin console server performance monitoring page shows information on the JVM.

Settings for server1

Configuration Protocols Logging Debug Monitoring

General Health Channels Performance Threads

Garbage Collect Dump Thread Stacks

Java Virtual Machine Memory Utilization Statistics

Heap Size Current:	320471040
Heap Free Current:	123103200
Heap Free Percent:	58
Heap Size Max:	477233152



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

To access the server health information, in the Domain Structure expand **Environment** and select **Servers**. In the servers table, select the server of interest by clicking its name. Click the **Monitoring > Performance** tab.

Monitoring Data Source Health

The admin console data source monitoring lets you view data source state and many statistics about its health and activity.

Settings for datasource1

Monitoring

Statistics Testing

Many data source attributes are available when you customize the table.

Customize this table

Deployed Instances of this Data Source (Filtered - More Columns Exist)

Showing 1 to 1 of 1 Previous | Next

Server	Enabled	State	JDBC Driver
server1	true	Running	oracle.jdbc.xa.client.OracleXADatasource

ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

To access data source monitoring, in the Domain Structure expand **Environment**, then **Services**, and select **Data Sources**. In the data sources table, select the data source of interest by clicking its name. Click the **Monitoring > Statistics** tab.

Many health, activity, and performance data source attributes are available to display by using the **Customize this table** link.

Example Data Source Performance Attributes

Customize the data source monitoring table to display performance data. For example:

Attribute	Description
Active Connections Current Count	The number of database connections currently in use
Current Capacity	The total number of connections in the connection pool
Failed Reserve Request Count	The running total of connection requests that could not be fulfilled
Leaked Connection Count	The number of connections reserved but not returned to the connection pool
Number Available	The number of connections idle and available for use



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

The following are some of the performance attributes available under data source monitoring :

- **Active Connections Current Count:** The number of connections currently in use by applications
- **Current Capacity:** The current count of JDBC connections in the connection pool in the data source
- **Failed Reserve Request Count:** The cumulative, running count of requests for a connection from this data source that could not be fulfilled
- **Leaked Connection Count:** The number of leaked connections. A leaked connection is a connection that was reserved from the data source but was not closed. Because it was not closed, it was not returned to the connection pool.
- **Number Available:** The number of database connections that are currently idle and available to be used by applications in this instance of the data source

JMX, MBeans, Managing, and Monitoring

WebLogic Server manages and monitors its resources by using the Java Management Extensions (JMX) API.

- JMX provides a standardized way of managing and monitoring resources through objects called MBeans (managed beans).
- WebLogic Server provides a large set of MBeans for all the resources that it manages and monitors.
 - These MBeans are used by WebLogic Server tools like the administration console, WLST, and the Monitoring Dashboard.



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Configuration MBeans contain information about the configuration of WebLogic Server resources. They represent the information that is stored in the domain's XML configuration documents. Each instance of WebLogic Server has an in-memory representation of its configuration as a collection of these read-only Configuration MBeans.

In addition to the read-only Configuration MBeans, the administration server maintains a collection of editable Configuration MBeans. To edit them, you use a JMX client (either the administration console, WLST, or Enterprise Manager Cloud Control). This client goes through the administration server to use its editable Configuration MBeans.

Runtime MBeans contain information about the runtime state of a server and its resources. They generally contain only data about the current state of a server or resource, and they do not persist this data. When you shut down a server instance, all runtime statistics and metrics from the runtime MBeans are destroyed. It is these runtime MBeans that are used by the Monitoring Dashboard to get real-time data.

Monitoring Dashboard

The Monitoring Dashboard:

- Is accessible from a link on the administration console home page
- Graphically presents current or historic WebLogic Server diagnostic data
 - Multiple graph types are available
- Allows you to monitor WebLogic Server MBean attributes
 - From active runtime MBeans (*polled metrics*)
 - From an archive collected by WLDF (*collected metrics*)



The dashboard opens in a new window or tab



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

The Monitoring Dashboard provides views and tools for graphically presenting diagnostic data. The underlying functionality for generating, retrieving, and persisting diagnostic data is provided by the WebLogic Diagnostics Framework (WLDF).

You can launch the Monitoring Dashboard from the administration console, or you can run it independently. To launch it from the admin console, go to the Home page and under “Charts and Graphs” click the **Monitoring Dashboard** link. The dashboard opens in its own window (or tab). To access the Monitoring Dashboard directly, use the URL:

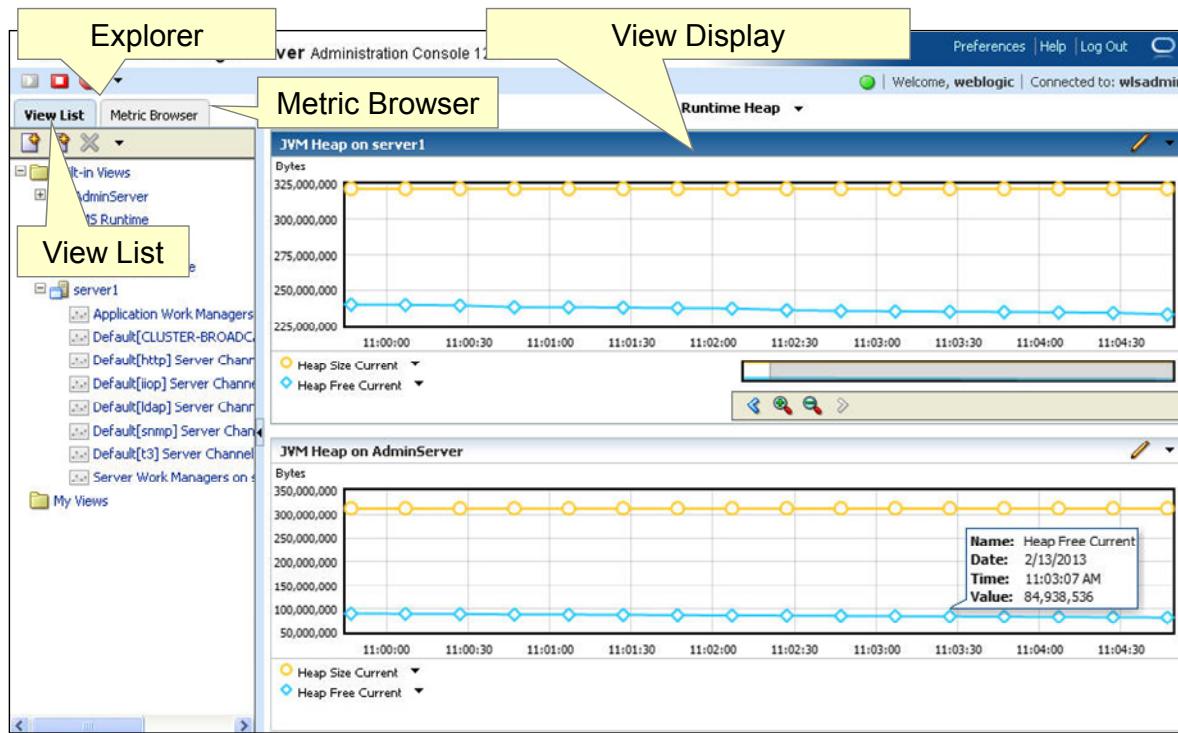
`http : admin_host : admin_port / console / dashboard`

The dashboard, like the admin console, requires you to log in with administrative credentials.

The diagnostic data displayed by the Monitoring Dashboard consists of runtime MBean attributes. These values are referred to in the Monitoring Dashboard as *metrics*. The dashboard obtains metrics from two sources:

- Directly from active runtime MBean instances. These metrics are referred to as *polled metrics*.
- From Archive data that has been collected by a WLDF (by a WLDF artifact called a Harvester). These metrics are referred to as *collected metrics*.

Monitoring Dashboard Interface



ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

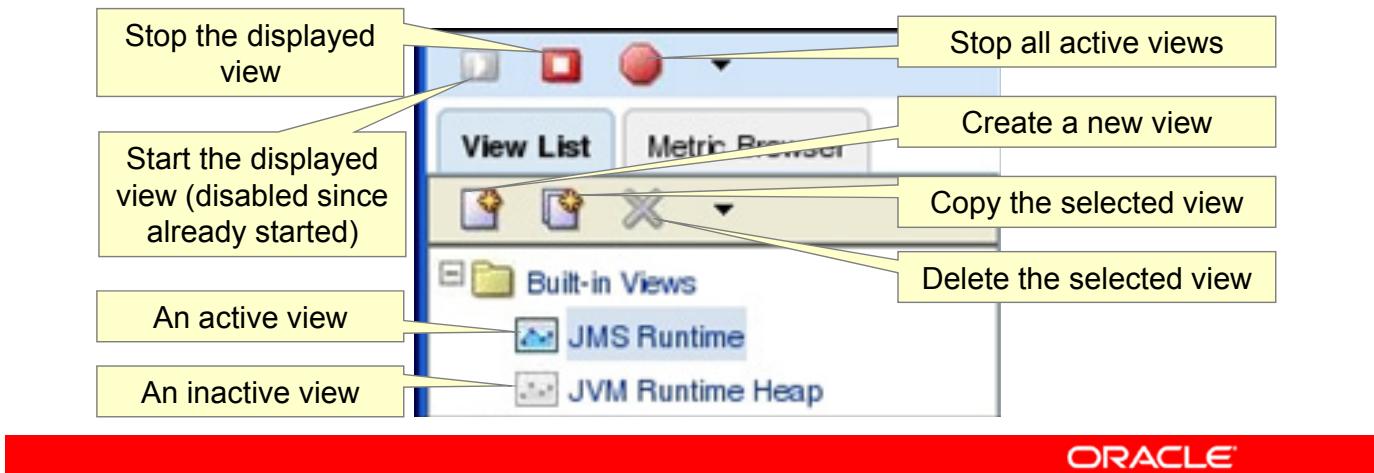
The Monitoring Dashboard has two main panels: the explorer panel and the view display panel.

The explorer panel has two tabs:

- **View List:** A list of existing built-in and custom views. It also contains controls for creating, copying, renaming, and deleting views.
- **Metric Browser:** A way of navigating to and selecting the specific MBean instance attributes whose metric values you want to display in a chart in a view

Views

- Are a way to organize your charts and graphs
- Typically display metrics that are related in some way
- Are individually started (to collect data) and stopped
- Continue to collect data even when not being displayed



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

ORACLE

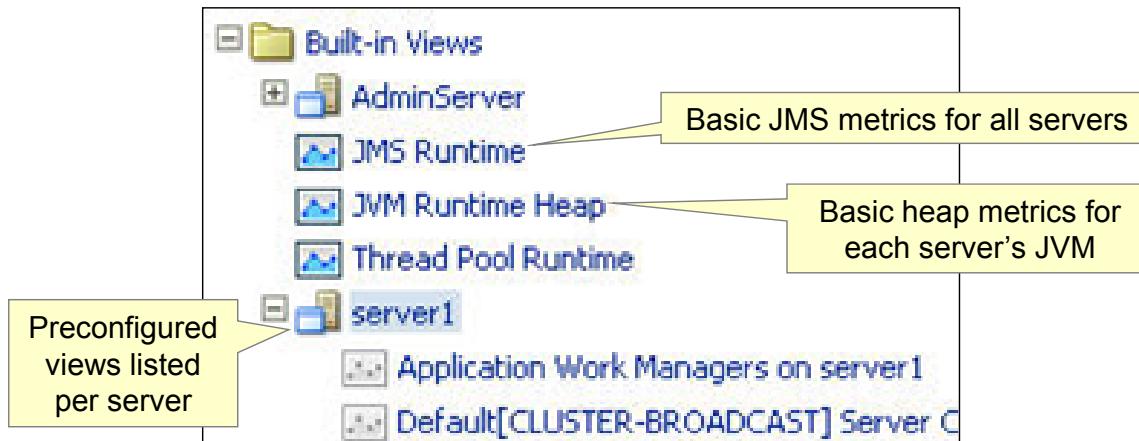
The View List tab lists views. A view is a collection of one or more charts that display captured monitoring and diagnostic data. You can access, create, and update views from the View List tab. When you click the name of a view on the View List tab, that view is displayed in the View Display on the right.

The dashboard uses icons to indicate the status of a view. A gray icon indicates that the view is inactive and data polling is not occurring for the charts in that view. A color icon indicates that the view is active and data polling is occurring for all charts in that view (this is true whether or not the view is currently displayed in the View Display).

To start the data collection for a view, click the view name in the list and click the green Start button above the tabs. To stop data collection, click the red-and-white rectangular Stop button. To stop all active views, click the red octagonal Stop All button.

Built-in Views

- The dashboard defines built-in views for some of the more critical runtime performance metrics.
- Built-in views cannot be modified (or deleted), but they can be copied and the copy modified.



ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

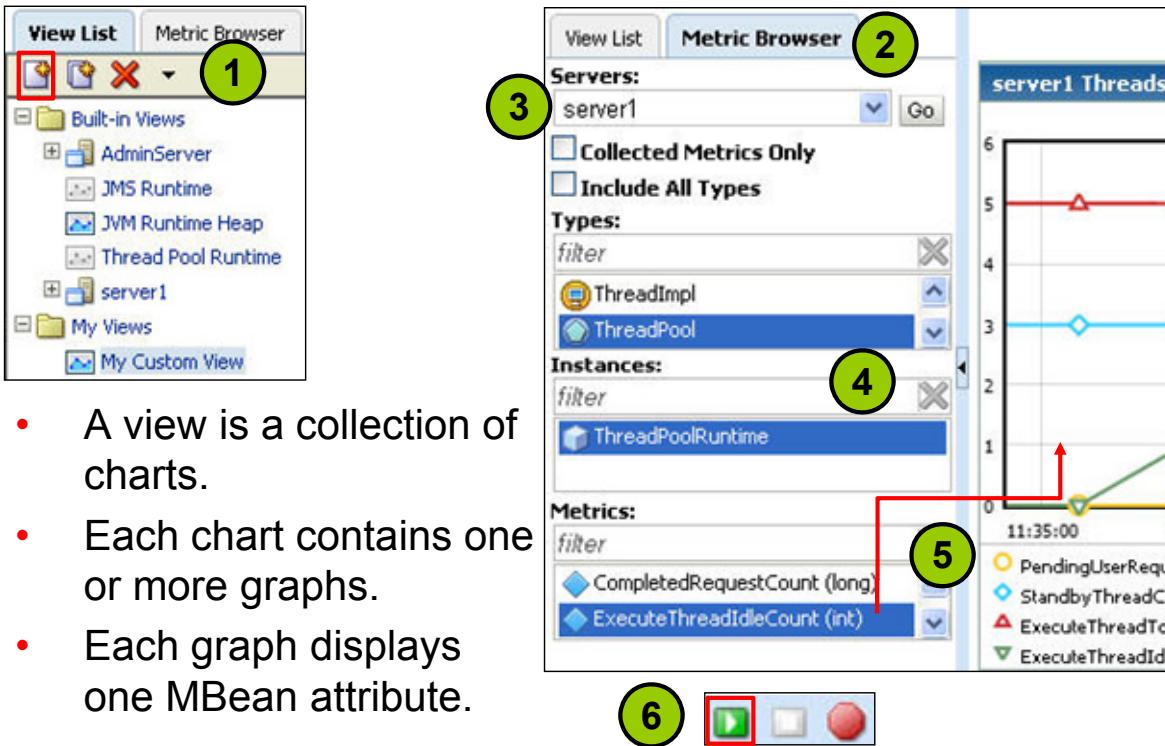
The built-in views are a set of predefined views of available runtime metrics for all running WebLogic Server instances in the domain. These views display some of the more critical runtime WebLogic Server performance metrics and serve as examples of the dashboard's chart and graph capabilities.

You cannot modify a built-in view, but you can copy it. This copy is now one of your custom views. As a custom view, the copy can be modified, renamed, saved, and later deleted.

Built-in views also cannot be deleted.

Custom views are available only to the user who created them and only within the current domain.

Creating a Custom View



- A view is a collection of charts.
- Each chart contains one or more graphs.
- Each graph displays one MBean attribute.

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

ORACLE

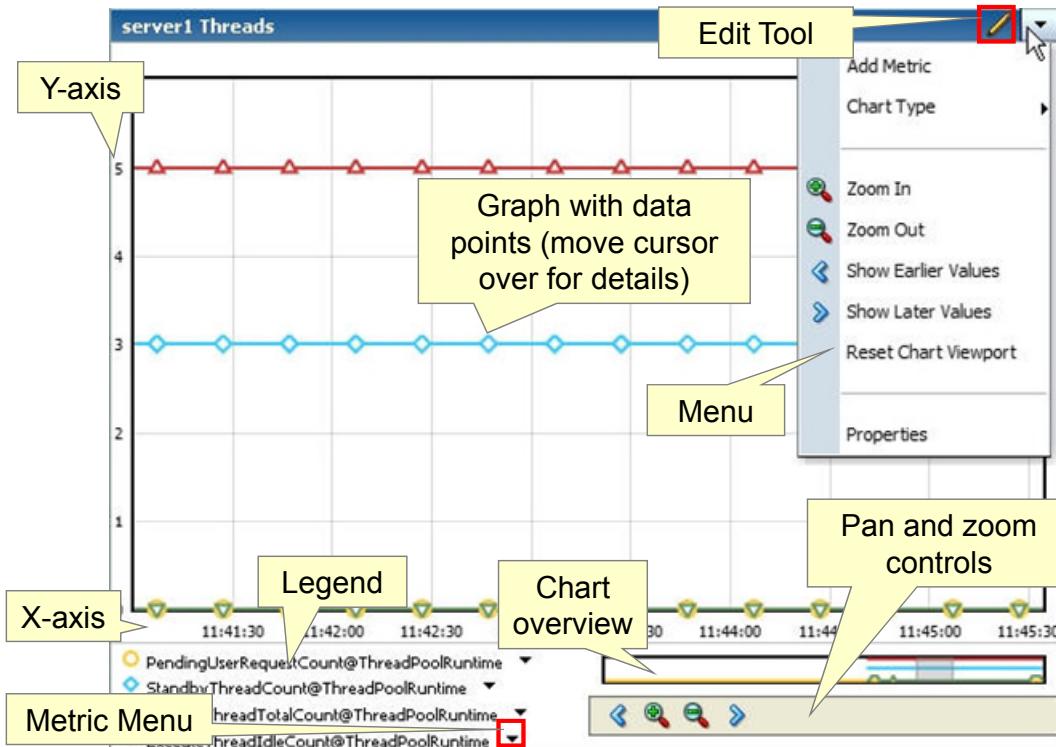
A custom view is any view created by a user. Custom views are available only to the user who created them. You can access a custom view again when needed.

To create a new custom view with a chart and graphs:

1. Click the **View List** tab. Then click the New View button. A new view appears in the list named New View. Replace the default name with something meaningful. Also, a new empty view appears in the View Display area. To add charts to the custom view, use the drop-down menu above the View Display area and click New Chart, or just drag in the first metric (MBean attribute) and a new chart is created for you.
2. To add graphs to a chart, first click the **Metric Browser** tab.
3. Select a server in the Servers drop-down list and click **Go**.
4. Select an MBean type and an MBean instance.
5. In the Metrics list for that instance, drag an MBean attribute to a chart. Note that a view may have as many charts as you like and a chart can graph as many metrics as you like. Also, if a metric is dragged into a view that contains no charts, the dashboard automatically creates a new chart to contain the graph.
6. When the metrics are in place, click the green Start button to start collecting data.

To delete a custom view, select the name of the view and click the Delete (red "X") button.

Anatomy of a Chart



ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

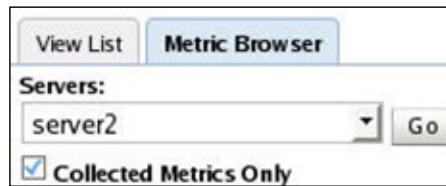
A chart contains one or more graphs that show data points over a specified time span. A chart also includes a legend that lists the sources of data for each graph along with their associated icons and colors.

When working with a view, you can do the following:

- Add charts to views
- Add graphs to charts
- Pan and zoom
- Edit labels and legends by using the Edit Tool
- Start and stop data collection for charts in a view

Current or Historical Data

- To view real-time metrics, no set up is needed. When a view is started, the runtime MBean instances are polled.
- To view historical (collected) metrics, WLDF must have been previously configured to collect data. Metrics collected by WLDF are placed into the diagnostic archive.
To view harvested data:
 1. In the View List, click the New View button.
 2. In the Metric Browser, select a Server.
 3. To see only harvested data, select Collected Metrics Only.
 4. Drag some attribute from the Metrics list to the new view.



ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

The Monitoring Dashboard displays two kinds of diagnostic metrics: real-time data directly from active runtime MBeans (called *polled metrics*) and historical data collected by a previously configured WLDF artifact called a Harvester (called *collected metrics*).

Note that with polled metrics, if polling has been taking place long enough for old data to be purged, a view will not contain all data from the time polling started.

If a WLDF Harvester was configured to harvest data for a particular metric, that historical data is available and can be displayed. Harvesters, or metric collectors, are configured within Diagnostic Modules. To create a Diagnostic Module, from the Domain Structure expand **Diagnostics** and select **Diagnostic Modules**, and then create one by clicking the **New** button. Select the new module and click the **Configuration > Collected Metrics** tabs to set up a collector. Select the **Enabled** check box to enable this collector. Set the period (in milliseconds) between collections in the **Sampling Period** field. Define the metric to collect by clicking the **New** button and use the Create a Metric wizard to select the **MBean Server location**, the **MBean Type**, and its **Attributes**. Target the Diagnostic Module to servers from which you want it to collect data. Harvested data is placed into the diagnostic archive, which can either be a WLDF file store or WLDF JDBC store. By default, the file store is used. The file can be found here: `server_name/data/store/diagnostics`.

Quiz

Which list of severity levels is in order from bad to worse?

- a. ERROR, CRITICAL, ALERT, EMERGENCY
- b. ALERT, ERROR, CRITICAL, EMERGENCY
- c. ERROR, ALERT, CRITICAL, EMERGENCY
- d. ERROR, CRITICAL, EMERGENCY, ALERT



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Answer: a

Quiz

A log filter can be applied to only one log message destination at a time.

- a. True
- b. False



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Answer: b

Summary

In this lesson, you should have learned how to:

- Configure and access WebLogic Server logs
- Enable WebLogic Server debugging output
- Monitor WebLogic Server health and performance
- Monitor JDBC data sources
- Access diagnostic charts in the Monitoring Dashboard



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Practice 8-1 Overview: Working with WebLogic Server Logs

This practice covers the following topics:

- Accessing the server log by using the admin console
- Creating and applying a log filter



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Practice 8-2 Overview: Monitoring WebLogic Server

This practice covers the following topics:

- Monitoring a server by using the admin console and the Monitoring Dashboard
- Monitoring JDBC data sources by using the admin console
- Enabling debugging by using the admin console



Copyright © 2014, Oracle and/or its affiliates. All rights reserved.