Oracle® Enterprise Manager

Oracle Database Metric Reference Manual

24ai Release 1 (24.1)
F97200-03
May 2025

Oracle Enterprise Manager Oracle Database Metric Reference Manual, 24ai Release 1 (24.1)

F97200-03

Contents

2 Cluster Database

Preface

This manual is a compilation of the Oracle Database metrics provided in Oracle Enterprise Manager.

In addition to this manual, information on the database-related target metrics is available in the following manuals:

- *Oracle Engineered Systems Metric Reference Manual*
- *Oracle Grid Infrastructure Metric Reference Manual*

Audience

This document is intended for Oracle Enterprise Manager users interested in Oracle Database metrics.

Documentation Accessibility

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

Access to Oracle Support

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

Related Resources

For more information, see the documents and other resources in Oracle Enterprise Manager Documentation Sets and Other Resources.

Conventions

The following text conventions are used in this document:

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

How to Use This Manual

The *Oracle Enterprise Manager Oracle Database Metric Reference Manual* (hereafter referred to as the *Oracle Database Metric Reference Manual*) lists the Oracle Database metrics that Enterprise Manager monitors.
To collect and monitor Oracle Database metrics, the Oracle Database plug-in must be deployed to an agent. The Oracle Database plug-in depends on the EM Platform plug-in.

For information on plug-ins, see Plug-ins Included In This Release in *Enterprise Manager Release Notes*.

This preface describes:

Structure of the Metric Reference Manual

The metrics in each chapter are in alphabetical order according to category.

Metric Information

Where available, each metric includes the following information:

- Description

  Explanation following the metric name. This text defines the metric and, where available, provides additional information pertinent to the metric.
- Metric Summary Table

  Where available, this table lists the target version, default collection frequency, default warning threshold, default critical threshold, and alert text for the metric.
- Data Source

  How the metric is calculated. In some metrics, data source information is not available.
- User Action

  Suggestions of how to solve the problem causing the alert.

Examples of Metric Summary Tables

This section provides examples of Metric Summary tables that you will see in the *Oracle Database Metric Reference Manual*.

When default thresholds are not defined for a metric, only the target version and default collection frequency are available.

Target Version	Collection Frequency
All versions	Every 15 minutes

The following table shows a metric where the server evaluation frequency is the same as the collection frequency.

Target Version	Evaluation and Collection Frequency
All versions	Every 10 minutes

Definitions of Columns in Metric Summary Tables

As previously mentioned, the Metric Summary table is part of the overall metric information. The following table provides descriptions of columns in the Metric Summary table.

Column Header	Column Definition
Target Version	Version of the target, for example, 19c. Note that if "**All versions**" is mentioned, the metric is available for target versions 19c, 21c, and 23ai in Oracle Enterprise Manager 24ai Release 1.
Evaluation and Collection Frequency	The rate at which the metric is collected and evaluated to determine whether it has crossed its threshold. The evaluation frequency is the same as the collection frequency.
Collection Frequency	The rate at which the Management Agent collects data. The default collection frequency for a metric comes from the Enterprise Manager default collection file for that target type.
Default Warning Threshold	Value that indicates whether a warning alert should be initiated. If the evaluation of the warning threshold value returns a result of TRUE for the specified number of consecutive occurrences defined for the metric, an alert triggers at the warning severity level.
Default Critical Threshold	Value that indicates whether a critical alert should be initiated. If the evaluation of the critical threshold value returns a result of TRUE for the specified number of consecutive occurrences defined for the metric, an alert triggers at the critical severity level.
Alert Text	Message indicating why the alert was generated. Words that display between percent signs (%) denote variables.

Abbreviations and Acronyms

To reduce the page count in this document, the following abbreviations and acronyms are used:

Abbreviation/Acronym	Name
Agent	Oracle Management Agent
Listener	Oracle Listener

About Metrics, Thresholds, and Alerts

A metric is a unit of measurement used to determine the health of a target. It is through the use of metrics and associated thresholds that Enterprise Manager sends out alerts notifying you of problems with the target.

Thresholds are boundary values against which monitored metric values are compared.

When a threshold is reached, Enterprise Manager generates an alert. An alert is an indicator signifying that a particular condition has been encountered and is triggered when one of the following conditions is true:

- A threshold is reached.
- An alert has been cleared.
- The availability of a monitored service changes. For example, the availability of an application server changes from up to down.
- A specific condition occurs. For example, an alert is triggered whenever an error message is written to a database alert log file.

Alerts are detected through a polling-based mechanism by checking for the monitored condition from a separate process at regular, predefined intervals.

Accessing Metrics

To access metrics from the Enterprise Manager Console, use the All Metrics page:

1. From the Enterprise Manager Console, choose the target.
2. From the target's home page, select the target type name, then **Monitoring**, and then **All Metrics**.

Editing Metrics

Out of the box, Enterprise Manager comes with default thresholds for critical metrics. Enterprise Manager generates alerts when warning and critical thresholds are reached, letting you know of impending problems so that you can address them in a timely manner.

To better suit the monitoring needs of your organization, you can edit the thresholds provided by Enterprise Manager and define new thresholds.

When defining thresholds:

- Choose acceptable values to avoid unnecessary alerts, while still being notified of issues in a timely manner.
- Adjust your metric thresholds based on metric trends. One of the more important actions you can perform with your monitoring system is to track metric trends for some period of time so you can make informed decisions about what metrics are important as well as what levels your thresholds should be set at.
- Set the number of occurrences appropriately. If some events occur only once or twice, for example, you might not need to be notified of them. You can set the number of occurrences of a metric that must be reached before you are notified.

To modify metric thresholds:

1. From the Enterprise Manager console, right-click the target name, select **Monitoring**, and then **All Metrics**.
2. From the **All Metrics** page, select the metric that you want to modify.
3. Click **Modify Thresholds**.
4. In the Modify Thresholds window, you can set values for settings such as:
    - Warning Threshold
    - Critical Threshold

- Occurrences Before Alert

> **Note:**
>
> You must have at least OPERATOR privilege on the target to make changes. Without OPERATOR privilege, the content of the Metric Threshold table is read-only.

5. Click **Save Thresholds** to upload the new metric settings to the Management Repository.

Specifying Multiple Thresholds

The Specifying Multiple Thresholds functionality enables you to define various subsets of data that can have different thresholds. By specifying multiple thresholds, you can refine the data used to trigger alerts, which are one of the key benefits of using Enterprise Manager. The key in specifying multiple thresholds is to determine how the comparison relates to the metric threshold as a whole, and what benefit will be realized by defining a more stringent or lax threshold for that particular device, mount point, and so on.

1

Database Instance

This chapter provides information about the Database Instance metrics.
The metric information includes some or all of the following: metric name, description, target version, default collection frequency, default warning threshold, default critical threshold, and alert text.

Alert Log

> **Note:**
>
> Oracle recommends using the DB Alert Log metrics instead of the Alert Log metrics.
>
> For information about the DB Alert Log metrics, see DB Alert Log.

The metrics in this category are used to create alerts by parsing the database alert log, for example, data block corruption, terminated session, and so on. The Alert Log metrics raise an alert containing the Error text and, when relevant, a link to the trace file for each ORA error that is reported in the alert log that matches the warning or critical thresholds defined for each category of error returned by the metric as defined in Metrics and Policy Settings but does not match the Alert Log Filter Expression.

> **Note:**
>
> The Alert Log and Alert Log Error Status metrics only return ORA errors from the Alert log. If the error is not an ORA error it will not be recognized by this metric. If you need to alert for non-ORA errors in the Alert Log it is suggested that you create a UDM for these purposes. See My Oracle Support Note 735137.1 for details.

Alert Log Filter Expression

The Alert Log Filter Expression is used (at the discretion of the Enterprise Manager administrator responsible for that target) to prevent errors that can be ignored resulting in alerts being raised in Enterprise Manager. It is a Perl regular expression that is used to filter all rows returned by the Alert Log metric.

The filtering takes place during the retrieval of errors from the Alert log and therefore no errors that match the expression are considered by either the Alert Log metric or, by definition, the Alert Log Error Status metric. Only those errors that do not match the Alert Log Filter Expression are compared against the Alert Log metric thresholds or counted for the Alert Log Error Status metric.

You can configure the Alert Log Filter Expression from several locations in Enterprise Manager for each target. For example, to configure the Alert Log Filter Expression, do one of the following:

- Click the link next to 'Alert Log' under 'Diagnostic Summary' from the DB Target home page and then click **Generic Alert Log Error Monitoring Configuration** under **Related Links**.
- Use any of the Metrics and Policy Settings pages for configuring the thresholds for each category of each metric.

> **Note:**
>
> The Alert Log Filter Expression is set at target level. No matter which page you use to configure it, you are configuring the same expression.

Alert Log Error Trace File

This metric reports the name of the trace file (if any) associated with the logged error.

Target Version	Collection Frequency
All versions	Every 15 minutes

Data Source

The following command is the data source for this metric where `$ORACLE_HOME` refers to the home of the Oracle Management Agent:

```
$ORACLE_HOME/sysman/admin/scripts/alertlog.pl
```

User Action

No user action is required.

Alert Log Name

This metric reports the name of the alert log file.

Target Version	Collection Frequency
All versions	Every 15 minutes

Data Source

The following command is the data source for this metric where `$ORACLE_HOME` refers to the home of the Oracle Management Agent:

```
$ORACLE_HOME/sysman/admin/scripts/alertlog.pl
```

User Action

No user action is required.

Archiver Alert Log Error

This metric signifies that an archiver error has occurred on the database being monitored, since the last sample time.

If the database is running in ARCHIVELOG mode, an alert is displayed when there is an archiver error (ORA-00257 and ORA-16038) and messages are written to the ALERT file. The ALERT file is a special trace file containing a chronological log of messages and errors.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	Not Defined	ORA-	The archiver error occurred at time/line number: %timeLine%.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Time/Line Number object.

If warning or critical threshold values are currently set for any Time/Line Number object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Time/Line Number object, use the Edit Thresholds page.

Data Source

The following command is the data source for this metric where $ORACLE_HOME refers to the home of the Oracle Management Agent:

$ORACLE_HOME/sysman/admin/scripts/alertlog.pl

User Action

Examine the ALERT log and archiver trace file for additional information. However, the most likely cause of this message is that the destination device is out of space to store the redo log file. Verify the device specified in the initialization parameter ARCHIVE_LOG_DEST is set up properly for archiving.

> **Note:**
>
> This event does not automatically clear because there is no automatic way of determining when the problem has been resolved. Therefore, you must manually clear the event after the problem is fixed.

Data Block Corruption Alert Log Error

This metric signifies that the database being monitored has generated a corrupted block error to the ALERT file since the last sample time. The ALERT file is a special trace file containing a chronological log of messages and errors. An alert event is triggered when data block corrupted messages (ORA-01157, ORA-01578, and ORA-27048) are written to the ALERT file.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	Not Defined	ORA-	A data block was corrupted at time/line number: %timeLine%.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Time/Line Number object.

If warning or critical threshold values are currently set for any Time/Line Number object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Time/Line Number object, use the Edit Thresholds page.

Data Source

The following command is the data source for this metric where $ORACLE_HOME refers to the home of the Oracle Management Agent:

$ORACLE_HOME/sysman/admin/scripts/alertlog.pl

User Action

Examine the ALERT log for additional information.

> **Note:**
>
> This event does not automatically clear because there is no automatic way of determining when the problem has been resolved. Therefore, you must manually clear the event after the problem is fixed.

Generic Alert Log Error

This metric signifies that the database being monitored has generated errors to the ALERT log file since the last sample time. The ALERT log file is a special trace file containing a chronological log of messages and errors. An alert event is triggered when Oracle Exception (ORA-006xx) messages are written to the ALERT log file. A warning is displayed when other ORA messages are written to the ALERT log file.

- For all supported databases monitored by Enterprise Manager release 10.2.0.4 Management Agent:

  Alert Log Filter - up to 1024 characters

  Warning or Critical Threshold - up to 256 characters

- For all supported databases monitored by Enterprise Manager release 10.2.0.5 Management Agent:

  Alert Log Filter - up to 4000 characters

  Warning or Critical Threshold - up to 4000 characters

Archiver error (ORA-00257) and data block corrupted (ORA-01578) messages are sent out as separate metrics.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	ORA-0*(600?|7445|4[0-9][0-9][0-9])[^0-9]	Not Defined	ORA-error stack (%errCodes%) logged in %alertLogName%.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Time/Line Number object.

If warning or critical threshold values are currently set for any Time/Line Number object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Time/Line Number object, use the Edit Thresholds page.

Data Source

The following command is the data source for this metric where $ORACLE_HOME refers to the home of the Oracle Management Agent:

$ORACLE_HOME/sysman/admin/scripts/alertlog.pl

User Action

Examine the ALERT log for additional information.

> **Note:**
>
> This event does not automatically clear because there is no automatic way of determining when the problem has been resolved. Therefore, you must manually clear the event after the problem is fixed.

Media Failure Alert Log Error

This metric represents the media failure alert log error. An alert event is triggered when messages ORA-01242 and ORA-01243 are written to the ALERT file.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	Not Defined	ORA-	Media failure was detected at time/line number: %timeLine%.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Time/Line Number object.

If warning or critical threshold values are currently set for any Time/Line Number object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Time/Line Number object, use the Edit Thresholds page.

Data Source

Not available.

User Action

No user action is required.

Session Terminated Alert Log Error

This metric signifies that a session terminated unexpectedly since the last sample time. The ALERT file is a special trace file containing a chronological log of messages and errors. An alert is displayed when session unexpectedly terminated (ORA-00603) messages are written to the ALERT file.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	ORA-	Not Defined	A session was terminated at time/line number: %timeLine%.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Time/Line Number object.

If warning or critical threshold values are currently set for any Time/Line Number object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Time/Line Number object, use the Edit Thresholds page.

Data Source

The source for this metric is $ORACLE_HOME/sysman/admin/scripts/alertlog.pl where $ORACLE_HOME refers to the home of the Oracle Management Agent.

User Action

Examine the ALERT log and the session trace file for additional information.

> **Note:**
>
> This event does not automatically clear because there is no automatic way of determining when the problem has been resolved. Therefore, you must manually clear the event after the problem is fixed.

Alert Log Error Status

> **Note:**
>
> Oracle recommends that you use DB Alert Log Error Status metrics instead of Alert Log Error Status metrics.
>
> For information about the DB Alert Log Error Status metrics, see DB Alert Log Error Status.

The metrics in this category count the number of errors returned in each category by the Alert Log Error metric after the Alert Log Filter expression has been taken into account but without taking the thresholds of the Alert Log Error metric into account and raises an alert if the number is greater than that specified in the Warning or Critical thresholds for that category. Therefore, it is possible for no alert to be raised by the Alert Log Error metric but still for the Alert Log Error Status metric to fire (even if the thresholds defined for the Alert Log Error metric are not matched). For more information on the Alert Log Filter Expression, see Alert Log Filter Expression.

Archiver Alert Log Error Status

This metric reflects the number of Archiver alert log errors witnessed the last time Enterprise Manager scanned the Alert Log.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	0	Not Defined	Archiver errors have been found in the alert log.

Data Source

The source of this metric is the Alert Log metric.

User Action

Examine the Alert Log.

Data Block Corruption Alert Log Error Status

This metric reflects the number of Data Block Corruption alert log errors witnessed the last time Enterprise Manager scanned the Alert Log.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	0	Not Defined	Data block corruption errors have been found in the alert log.

Data Source

The source of this metric is the Alert Log metric.

User Action

Examine the Alert Log.

Generic Alert Log Error Status

This metric reflects the number of Generic alert log errors witnessed the last time Enterprise Manager scanned the Alert Log.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	0	Not Defined	%value% distinct types of ORA- errors have been found in the alert log.

Data Source

The source of this metric is the Alert Log metric.

User Action

Examine the Alert Log.

Media Failure Alert Log Error Status

This metric represents the media failure alert log error status.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	0	Not Defined	Media failure errors have been found in the alert log.

Data Source

Not available.

User Action

No user action is required.

Session Terminated Alert Log Error Status

This metric reflects the number of Session Terminated alert log errors witnessed the last time Enterprise Manager scanned the Alert Log.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	0	Not Defined	Session terminations have been found in the alert log.

Data Source

The source of this metric is the Alert Log metric.

User Action

Examine the Alert Log.

Archive Area

This metric category contains metrics that track the utilization of the database archived redo log destinations.

If the database is running in ARCHIVELOG mode, these metrics examine the space utilization in the database archived redo log destinations (as specified in the LOG_ARCHIVED_DEST_n initialization parameters). If the database is not running in ARCHIVELOG mode, these metrics are not applicable and the collections do not run. For each archived redo log destination, this metric category returns the total, used, and free space. The methodology used to collect this information is different depending on whether the destinations are configured to use a conventional filesystem or an ASM diskgroup.

> **Note:**
>
> For databases that are configured to archive to the Fast Recovery Area, the Archive Area metrics (Archive Area Used (%), Archive Area Used (KB), Free Archive Area (KB), and Total Archive Area (KB)) are not applicable. Instead, use the Recovery Area Free Space (%) metric to monitor Fast Recovery Area usage.

The formulas used to calculate all the metrics in this metric group depend on the following conditions:

- Whether there is a quota configured in the LOG_ARCHIVE_DEST_n parameter setting.
- Whether the archived redo log destination is configured to use an ASM diskgroup or a regular filesystem location.

Applying these conditions yields three possible archive area scenarios that must be accommodated by these metrics:

- Quota is set
- No quota set
  - Archive area on regular filesystem
  - Archive area on ASM diskgroup

Archive Area Used (%)

The Archive Area Used (%) metric returns the percentage of space used in the archived redo log destination. If the space used is more than the threshold value given in the threshold arguments, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 15 Minutes	80	Not Defined	%value%%% of archive area %archDir% is used.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Archive Area Destination object.

If warning or critical threshold values are currently set for any Archive Area Destination object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Archive Area Destination object, use the Edit Thresholds page.

Data Source

The formula used for each of the three archive area scenarios is as follows:

- Quota is set: Regardless of whether the destination is using ASM or a conventional filesystem location, if the QUOTA_SIZE attribute is specified in the associated LOG_ARCHIVE_DEST_n parameter (meaning there is a quota specified for the destination), the percentage is calculated using the following formula:

  ```
  Archive Area Used (%) = (QUOTA_USED/QUOTA_SIZE) * 100
  ```

- No quota is set:
  - Archive area on regular filesystem: Free and used space in the archive area is determined by running the UNIX `df -k` command against the filesystem on which the archive area resides.
  - Archive area on ASM diskgroup: The space used in the archive area is calculated by first determining the total diskgroup size (minus space required for redundancy management) and dividing that by the diskgroup redundancy factor (1 for External, 2 for Normal, 3 for High) to arrive at the "Total Safely Usable" space. Then, the "Safely Usable Free" space is determined, which is the free space that can be safely utilized taking mirroring and redundancy needs into account. The SQL used to determine these values is as follows:

    ```
    select (((NVL(dg.total_mb,0) -
    NVL(dg.required_mirror_free_mb,0))*1024)/
    decode(dg.type,'EXTERN',1,'NORMAL',2,'HIGH',3,1))
    TotalSafelyUsable,NVL(dg.usable_file_mb,0)*1024 SafelyUsableFree from
    V$ASM_DISKGROUP_STAT dg
    where state in ('CONNECTED', 'MOUNTED') and name='$diskGroup'";
    ```

Using the values from this query, the Archive Area Used (%) is calculated as follows:

```
Archive Area Used (%) = [(TotalSafelyUsable - SafelyUsableFree)/
TotalSafelyUsable] * 100
```

User Action

Verify that the database archived redo log destination parameters are configured properly. For more information, see Specifying Archive Destinations in *Oracle Database Administrator's Guide*.

Archive Area Used (KB)

This metric returns the total space used (in KB) on the device containing the archived redo log destination directory.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The formula used for each of the three archive area scenarios is as follows:

- Quota is set: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula:

  ```
  Archive Area Used (KB) = QUOTA_USED
  ```

- No quota is set:
  - Archive area on regular filesystem: Same underlying methodology as described above for the Archive Area Used (%) metric.
  - Archive area on ASM diskgroup: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula (referencing the values from the SQL query in the Archive Area Used (%) metric):

    ```
    Archive Area Used (KB) = TotalSafelyUsable - SafelyUsableFree
    ```

User Action

Verify that the database archived redo log destination parameters are configured properly. For more information, see Specifying Archive Destinations in *Oracle Database Administrator's Guide*.

Free Archive Area (KB)

This metric returns the free space (in KB) on the device containing the archived redo log destination directory.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	Not Defined	Not Defined	Archive area %archDir% has %value% free KB remaining.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Archive Area Destination object.

If warning or critical threshold values are currently set for any Archive Area Destination object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Archive Area Destination object, use the Edit Thresholds page.

Data Source

The formula used for each of the three archive area scenarios is as follows:

- Quota is set: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula:

```
Free Archive Area (KB) = QUOTA_SIZE - QUOTA_USED
```

- No quota is set:
  - Archive area on regular filesystem: Same underlying methodology as described above for the Archive Area Used (%) metric.
  - Archive area on ASM diskgroup: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula (referencing the values from the SQL query in the Archive Area Used (%) metric):

```
Free Archive Area (KB) = SafelyUsableFree
```

User Action

Verify that the database archived redo log destination parameters are configured properly. For more information, see Specifying Archive Destinations in *Oracle Database Administrator's Guide*.

Total Archive Area (KB)

This metric returns the total space (in KB) on the device containing the archived redo log destination directory.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The formula used for each of the three archive area scenarios is as follows:

- Quota is set: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula:

```
Total Archive Area (KB) = QUOTA_SIZE
```

- No quota is set:
  - Archive area on regular filesystem: Same underlying methodology as described above for the Archive Area Used (%) metric.
  - Archive area on ASM diskgroup: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula (referencing the values from the SQL query in the Archive Area Used (%) metric):

```
Total Archive Area (KB) = TotalSafelyUsable
```

User Action

Verify that the database archived redo log destination parameters are configured properly. For more information, see Specifying Archive Destinations in *Oracle Database Administrator's Guide*.

Availability Notifications (Server Generated Alert)

This section provides information on the metrics in the Availability Notifications (Server Generated Alert) category.

Target Version	Evaluation and Collection Frequency
All versions	N/A

Metric Name	Description
Database Down	Notifies when a database is down.

Collect SQL Response Time

The metrics in the this category represent the SQL response time.

SQL Response Time (%)

This metric represents the SQL response time.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

Not available.

User Action

No user action is required.

Data Failure

Enterprise Manager uses the metrics in this category to alert you to checker failures reported in the alert log. It contains the number of checker failures detected. It also generates a critical alert by default when these problems are found in the alert log.

The alert log file provides this data. It is collected using the perl script $ORACLE_HOME/sysman/admin/scripts/alertlog.pl where $ORACLE_HOME refers to the home of the Management Agent.

Alert Log Name

This metric reports the name of the alert log file.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

No user action is required.

Data Failure Detected

This metric signifies that a database health checker has detected one or more persistent data failures. Examples of data failures include missing files, corrupt files, inconsistent files, and corrupt blocks. The alert shows the number of data failures detected by a checker run. Details of individual data failures can be accessed from the Perform Recovery page in Enterprise Manager.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.[1]	Checker run found %numberOfFailures% new persistent data failures.

[1] After an alert is triggered for this metric, it must be manually cleared.

Setting Thresholds

To edit the thresholds for any of the following metrics, from the Enterprise Manager UI, right-click the target name, select **Monitoring**, then **Metric and Collection Settings**. The following settings provide examples of some of the possible settings:

- Warning Threshold: Not Defined; Critical Threshold: .*

In this case, the Management Agent generates a critical error alert in Enterprise Manager when a data failure occurs.

- Warning Threshold: .*; Critical Threshold: Not Defined

  In this case, the Management Agent generates a warning alert in Enterprise Manager when a data failure occurs.

- Warning Threshold: Not Defined; Critical Threshold: Not Defined

  In this case, the Management Agent does not generate an alert in Enterprise Manager when a data failure occurs.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Details of individual data failures can be accessed from the Perform Recovery page in Enterprise Manager.

> **Note:**
>
> This event does not automatically clear because there is no automatic way of determining when the problem has been resolved. Therefore, you must manually clear the event after the problem is fixed.

Data Guard Fast-Start Failover

This section provides information on the metrics in the Data Guard Fast-Start Failover category, which generates an alert to notify of a new primary database after a fast-start failover occurred.

Target Version	Evaluation and Collection Frequency
All versions	Every 5 minutes

Metric Name	Description
Fast-Start Failover Occurred	Indicates whether a fast-start failover occurred in the last 15 minutes.
Last Fast-Start Failover Reason	The reason why the fast-start failover occurred.
Last Fast-Start Failover Time	The timestamp of the last fast-start failover.

Data Source

The data source for this metric is the v$fs_failover_stats view.

Data Guard Fast-Start Failover Observer – Oracle Database 11gR2 to 18c

The metrics in this category monitor the status of a fast-start failover observer in the Data Guard configuration.

Observer Status

This metric generates a critical alert on the primary database if the fast-start failover (FSFO) configuration is in an unobserved condition, indicating that FSFO is not currently possible.

Table 1-1 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR2, 11gR202, 12c, 12cR102, 12cR2, 18c	Every 1 minute	Not Defined	Error	The Data Guard fast-start failover observer status is %value%.

User Action

If the Data Guard configuration was configured in Enterprise Manager to use the automatic Observer restart feature, the alert will clear after a new observer process is restarted. Otherwise, determine the cause of the unobserved condition, and restart the Observer process if necessary.

Data Guard Fast-Start Failover Observers – Oracle Database 19c and later

This section provides information on the metrics in the Data Guard Fast-Start Failover Observers category for Oracle Database 19c and later. These metrics monitor the status of all the fast-start failover observers in the Data Guard configuration.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
19c and later	Every 5 minutes	Not Defined	Error	The Data Guard fast-start failover observer status is %value%.

Metric Name	Description	Data Source
Is Master Observer	Indicates whether the observer is the master observer (YES or NO).	v$fs_failover_observers

Metric Name	Description	Data Source
Is Observer Registered	Indicates whether the observer is registered (YES or NO).	v$fs_failover_observers
Observer Host	The server on which the observer is running.	v$fs_failover_observers
Observer Log File	The observer log file. **Note**: This metric is only available for Oracle Database 21c. For Oracle Database 19c, this metric column is empty.	v$fs_failover_observers
Observer Name	The name of the observer.	v$fs_failover_observers
Observer State File	The observer state file. **Note**: This metric is only available for Oracle Database 21c. For Oracle Database 19c, this metric column is empty.	v$fs_failover_observers
Observer Status	The status of the observer. **Note**: This metric is obtained from the **Observer Host**, **Pinging Active Target**, and **Pinging Primary** metrics.	v$fs_failover_observers
Pinging Active Target	Indicates whether the observer is currently connected to the active target (YES or NO).	v$fs_failover_observers
Pinging Primary	Indicates whether the observer is currently connected to the primary database (YES or NO).	v$fs_failover_observers

Data Guard Performance

The metrics in this category report on Data Guard performance.

Apply Lag (seconds)

This metric displays (in seconds) how far the standby is behind the primary.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Table 1-2 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	The standby database is approximately %value% seconds behind the primary database.

Data Source

The data source for this metric is the following command:

v$dataguard_stats('apply lag')

Estimated Failover Time (seconds)

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

This metric shows the approximate number of seconds required to failover to this standby database. This accounts for the startup time, if necessary, plus the remaining time required to apply all the available redo on the standby. If a bounce is not required, it is only the remaining apply time.

Table 1-3 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	The estimated time to failover is approximately %value% seconds.

Data Source

The data is derived from the following formula:

v$dataguard_stats ('estimated startup time','apply finish time','standby has been open')

Redo Apply Rate (KB/second)

This metric displays the Redo Apply Rate in KB/second on this standby.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Table 1-4 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	The redo apply rate is %value% KB/sec.

Redo Generation Rate (KB/second)

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Table 1-5 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	The redo generation rate is %value% KB/sec.

Transport Lag (seconds)

The approximate number of seconds of redo not yet available on this standby database or Far Sync instance. This may be because the redo has not yet been shipped or there may be a gap.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Data Source

The data is derived from the following formula:

v$dataguard_stats('transport lag')

Table 1-6 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	The standby database is approximately %value% seconds behind the primary database.

Transport Lag Data Refresh Time

Transport Lag metrics are computed based on data that is periodically received from the primary database. An unchanging value in this column across multiple queries indicates that the standby database or the Far Sync instance is not receiving data from the primary database.

Data Source

DATUM_TIME in v$dataguard_stats

Data Guard Status

The metrics in this category check the status, data not received, and data not applied for the databases in the Data Guard configuration.

Data Guard Status

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Use the Data Guard Status metric to check the status of each database in the Data Guard configuration.

By default, a critical and warning threshold value is set for this metric column. Alerts will be generated when threshold values are reached. You can edit the value for a threshold as required. Certain ORA- errors can be ignored by providing the ORA- errors in the Data Guard Status Filter expression.

Table 1-7 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Warning	Error	The Data Guard status of %dg_name% is %value%.

User Action

1. Check the Edit Properties General page for the primary and standby databases for detailed information.
2. Examine the database alert logs and the Data Guard broker logs for additional information.

Database Files

The metrics in this category represent the average file read time and average file write time for the database files.

Average File Read Time (centi-seconds)

This metric represents the average file read time, measured in hundredths of a second.

Target Version	Server Evaluation Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each File Name object.

If warning or critical threshold values are currently set for any File Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each File Name object, use the Edit Thresholds page.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Average File Write Time (centi-seconds)

This metric represents the average file write time, measured in hundredths of a second.

Target Version	Server Evaluation Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each File Name object.

If warning or critical threshold values are currently set for any File Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each File Name object, use the Edit Thresholds page.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Database Job Status

The metrics in this category represent the health of database jobs registered through the DBMS_SCHEDULER interface.

Broken Job Count

The Oracle Server job queue is a database table that stores information about local jobs such as the PL/SQL call to execute for a job such as when to run a job. Database replication is also managed by using the Oracle job queue mechanism using jobs to push deferred transactions to remote master sites, to purge applied transactions from the deferred transaction queue or to refresh snapshot refresh groups.

A job can be broken in two ways:

- Oracle failed to successfully execute the job after a specified number of attempts (defined in the job).
- The job is explicitly marked as broken by using the procedure DBMS_ JOB.BROKEN.

This metric checks for broken DBMS jobs. A critical alert is generated if the number of broken jobs exceeds the value specified by the threshold argument.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	0	Not Defined	%value% job(s) are broken.

Data Source

The data is derived from the following formula:

```
SUM(broken)
  FROM (SELECT DECODE(broken, 'N', 0, 1) broken
         FROM dba_jobs
  UNION ALL
         SELECT DECODE(STATE, 'BROKEN', 1, 0) broken
          FROM dba_scheduler_jobs
```

User Action

From the Enterprise Manager console, check the Scheduler Job History page or query the ALL_SCHEDULER_JOB_RUN_DETAILS view for error information.

Correct the problem that is preventing the job from running. Force immediate re-execution of the job by calling DBMS_SCHEDULER.RUN.

Failed Job Count

The Oracle Server job queue is a database table that stores information about local jobs such as the PL/SQL call to execute for a job such as when to run a job. Database replication is also managed by using the Oracle job queue mechanism using jobs to push deferred transactions to remote master sites, to purge applied transactions from the deferred transaction queue or to refresh snapshot refresh groups.

If a job returns an error while Oracle is attempting to execute it, the job fails. Oracle repeatedly tries to execute the job doubling the interval of each attempt. If the job fails after a specified number of times (specified in the job definition), Oracle automatically marks the job as broken and no longer tries to execute it.

This metric checks for failed DBMS jobs. An alert is generated if the number of failed job exceeds the value specified by the threshold argument.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	0	Not Defined	%value% job(s) are broken.

Data Source

The data is derived from the following formula:

```
SELECT SUM(failed)
          FROM (SELECT DECODE(NVL(failures,0), 0, 0, 1) failed
                  FROM dba_jobs
        UNION ALL
```

```
                SELECT
DECODE(STATUS ,'FAILED',DECODE(STATE,'BROKEN',0,'DISABLED',0,1),0) failed FROM (SELECT
all_jobs.OWNER,
                                all_jobs.JOB_NAME,
                                all_runs.STATUS,
                                all_jobs.STATE
                             FROM (SELECT OWNER,
                                          JOB_NAME, MAX(ACTUAL_START_DATE) AS START_DATE
                                     FROM DBA_SCHEDULER_JOB_RUN_DETAILS
                                 GROUP BY OWNER,JOB_NAME) last_run ,
DBA_SCHEDULER_JOB_RUN_DETAILS all_runs,
                                    DBA_SCHEDULER_JOBS all_jobs
                            WHERE all_runs.OWNER(+)=all_jobs.OWNER
                              AND all_runs.JOB_NAME(+)=all_jobs.JOB_NAME
                              AND last_run.OWNER(+)=all_jobs.OWNER
                              AND last_run.JOB_NAME(+)=all_jobs.JOB_NAME
                              AND all_runs.ACTUAL_START_DATE=last_run.START_DATE))
```

User Action

From the Enterprise Manager console, check the Scheduler Job History page or query the ALL_SCHEDULER_JOB_RUN_DETAILS view for error information. Correct the problem that is preventing the job from running.

Database Limits

The metrics in this category represent the percentage of resource limitations at which the Oracle Server is operating.

Current Logons Count

This metric represents the current number of logons.

> **Note:**
>
> Unlike most metrics, which accept thresholds as real numbers, this metric can only accept an integer as a threshold.

Target Version	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text

Data Source

The data is derived from the current logins.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Current Open Cursors Count

This metric represents the current number of opened cursors.

> **Note:**
>
> Unlike most metrics, which accept thresholds as real numbers, this metric can only accept an integer as a threshold.

Target Version	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 minutes	Not Defines	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text

Data Source

The data is derived from the current open cursors.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Lock Limit Usage (%)

The DML_LOCKS initialization parameter specifies the maximum number of DML locks. The purpose of DML locks is to guarantee the integrity of data being accessed concurrently by multiple users. DML locks prevent destructive interference of simultaneous conflicting DML and/or DDL operations.

This metric checks for the utilization of the lock resource against the values (percentage) specified by the threshold arguments. If the percentage of all active DML locks to the limit set in the DML_LOCKS initialization parameter exceeds the values specified in the threshold arguments, then a warning or critical alert is generated.

If DML_LOCKS is 0, this test fails to register. A value of 0 indicates that enqueues are disabled.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	Not Defined	Not Defined	%target% has reached %value%%% of the lock limit.

Data Source

The data is derived from the following formula:

```
SELECT resource_name name,
  100*DECODE(initial_allocation, ' UNLIMITED', 0, current_utilization /
```

```
  initial_allocation) usage
FROM v$resource_limit
WHERE LTRIM(limit_value)
  != '0' AND LTRIM(initial_allocation) != '0' AND resource_name = 'dml_locks'
```

User Action

Increase the DML_LOCKS instance parameter by 10%.

Process Limit Usage (%)

The PROCESSES initialization parameter specifies the maximum number of operating system user processes that can simultaneously connect to a database at the same time. This number also includes background processes utilized by the instance.

This metric checks for the utilization of the process resource against the values (percentage) specified by the threshold arguments. If the percentage of all current processes to the limit set in the PROCESSES initialization parameter exceeds the values specified in the threshold arguments, then a warning or critical alert is generated.

Target Version	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

Depending on your release, one of the following derives the data:

The fifth column provides process usage. You can get this information from the inst_perf.xmlp file located in the plug-in installation directory (plugins/oracle.sysman.db.agent.plugin_*version*/metadata).

```
SELECT /*+ ORDERED */
    TO_CHAR( FROM_TZ( CAST(m.end_time AS TIMESTAMP),
           TO_CHAR(systimestamp, 'tzr') )
    AT TIME ZONE sessiontimezone,
    'YYYY-MM-DD HH24:MI:SS'),
    SUM(CASE WHEN a.internal_metric_name = 'logons'
        THEN m.value ELSE 0 END) logons,
    SUM(CASE WHEN a.internal_metric_name = 'opencursors'
        THEN m.value ELSE 0 END) opencursors,
    SUM(CASE WHEN a.internal_metric_name = 'user_limit'
        THEN m.value ELSE 0 END) user_limit,
    SUM(CASE WHEN a.internal_metric_name = 'process_usage'
        THEN m.value ELSE 0 END) process_usage,
    SUM(CASE WHEN a.internal_metric_name = 'session_usage'
        THEN m.value ELSE 0 END) session_usage
   FROM v$alert_types a, v$threshold_types t, v$sysmetric m
  WHERE a.internal_metric_category = 'Database_Resource_Usage'
    AND a.reason_id = t.alert_reason_id
    AND t.metrics_id = m.metric_id
    AND m.group_id = 2
    AND :1 != 'BASIC'
    AND m.end_time <= SYSDATE
   GROUP BY m.end_time
```

```
ORDER BY m.end_time ASC
```

User Action

Verify that the current PROCESSES instance parameter setting has not exceeded the operating system-dependent maximum. Increase the number of processes to be at least 6 + the maximum number of concurrent users expected to sign in to the instance.

Session Limit Usage (%)

The SESSIONS initialization parameter specifies the maximum number of concurrent connections that the database will allow.

This metric checks for the utilization of the session resource against the values (percentage) specified by the threshold arguments. If the percentage of the number of sessions, including background processes, to the limit set in the SESSIONS initialization parameter exceeds the values specified in the threshold arguments, then a warning or critical alert is generated.

Target Version	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

```
SELECT resource_name name,
  100*DECODE(initial_allocation, ' UNLIMITED', 0, current_utilization) != '0'
  AND resource_name = 'sessions'
```

User Action

Increase the SESSIONS instance parameter. For XA environments, confirm that SESSIONS is at least 2.73 * PROCESSES. For shared server environments, confirm that SESSIONS is at least 1.1 * maximum number of connections.

User Limit Usage (%)

The LICENSE_MAX_SESSIONS initialization parameter specifies the maximum number of concurrent user sessions allowed simultaneously.

This metric checks whether the number of users logged on is reaching the license limit. If the percentage of the number of concurrent user sessions to the limit set in the LICENSE_MAX_SESSIONS initialization parameter exceeds the values specified in the threshold arguments, then a warning or critical alert is generated. If LICENSE_MAX_SESSIONS is not explicitly set to a value, the test does not trigger.

> **Note:**
>
> This metric is most useful when session licensing is enabled. Refer to the *Oracle Server Reference Manual* for more information on LICENSE_MAX_SESSIONS and LICENSE_MAX_USERS.

> **Note:**
>
> Unlike most metrics, which accept thresholds as real numbers, this metric can only accept an integer as a threshold.

Target Version	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text

Data Source

The data is derived from the following formula:

```
SELECT 'user' name,
  100*DECODE(session_max, 0, 0, sessions_current/session_max) usage
  FROM v$license
```

User Action

This typically indicates that the license limit for the database has been reached. You must acquire additional licenses, then increase LICENSE_MAX_ SESSIONS to reflect the new value.

Database Replay

The metrics in this category show the current status (on/off) of database workload capture and replay.

Workload Capture Status

This metric shows if the database workload capture is in progress.

This metric is available for all versions.

Data Source

The source of the data is the server-generated alert triggered by the target database when a capture is started.

User Action

No user action is required.

Workload Replay Status

This metric shows if database workload replay is in progress.

This metric is available for all versions.

Data Source

The source of the data is the server-generated alert triggered by the target database when a replay is started.

User Action

No user action is required.

Database Replay Client

The metrics in this category show the resource usage of the replay clients during database workload replay.

Average I/O Latency (milliseconds)

This metric reflects the average response time for a single I/O for a database replay client.

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All versions	null

Data Source

The source of the data is the server-generated alert triggered by the target database when an alarming condition is detected in a replay client.

User Action

Run the calibrate utility of the replay client and restart a replay with the suggested number of replay clients, distributed between machines with the necessary capacity.

Replay Threads (%) Performing I/O

This metric represents the number of replay client connections performing I/O operation concurrently.

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager. The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All versions	null

Data Source

The source of the data is the server-generated alert triggered by the target database when an alarming condition is detected in a replay client.

User Action

Run the calibrate utility of the replay client and restart a replay with the suggested number of replay clients, distributed between machines with the necessary capacity.

Replay Threads (%) Using CPU

This metric represents the number of replay client connections using the CPU concurrently.

Target Version	Collection Frequency
All versions	null

Data Source

The source of the data is the server-generated alert triggered by the target database when an alarming condition is detected in a replay client.

User Action

Run the calibrate utility of the replay client and restart a replay with the suggested number of replay clients, distributed between machines with the necessary capacity.

Database Scheduler Jobs

This section provides information on the metrics in the Database Scheduler Jobs category, which report the current status of DBMS jobs registered through the DBMS_SCHEDULER interface. Using these metrics, you can monitor long running jobs and obtain alerts on individual jobs.

Elapsed Running Time (in Minutes)

The duration of time the current DBMS job has been running, in minutes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	> 5 Minutes	> 30 Minutes	DBMS job %job_name% for %owner% has been running for %value% minutes.

Failure Count

The number of times the DBMS job has failed during the last collection period.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	> 0	Not Defined	DBMS job %job_name% for %owner% has failed %value% time(s) during last collection period.

State

The current state of the DBMS job.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	DISABLED	BROKEN	DBMS job %job_name% for %owner% is in %state% state.

Database Services

The metrics in this category include the service CPU time and service response time.

Service CPU Time (per user call) (microseconds)

This metric represents the average CPU time, in microseconds, for calls to a particular database service.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	10 minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Service Name object.

If warning or critical threshold values are currently set for any Service Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Service Name object, use the Edit Thresholds page.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Service Response Time (per user call) (microseconds)

This metric represents the average elapsed time, in microseconds, for calls to a particular database service.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	10 minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Service Name object.

If warning or critical threshold values are currently set for any Service Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Service Name object, use the Edit Thresholds page.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Database Vault Attempted Violations - Command Rules

The metrics in this category monitor violation attempts against the Database Vault database.

Database Vault Attempted Violations Count - Command Rules

This metric is used to enable Database Vault Security Analyst to keep a watch on the violation attempts against the Database Vault database. Database Vault Security Analysts can pick the command rules that they would like to get alerted on and even further filter them based on the different types of attempts by mentioning different thresholds to match SQL commands causing the violations.

This metric is not enabled out of the box. You must enable it from Metrics and Policy Settings page. By default, this metric is collected every 1 hour, but you can change the collection frequency.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Hour	Not Defined	Not Defined	%ACTION_OBJECT_NAME% got violated at %VIOLATIONTIMESTAMP%

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Database Vault Command Rule and Violation Time objects.

If warning or critical threshold values are currently set for any unique combination of Database Vault Command Rule and Violation Time objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Database Vault Command Rule and Violation Time objects, use the Edit Thresholds page.

Data Source

The attempted violations are picked up from the target's database vault audit trail. Only audit entries related to a command rule, which represent failed attempts to execute a SQL, are selected. Specified thresholds should match the SQL command causing command rule violation.

User Action

To know more about the violations, for example, the command that was violated, which database user triggered the violation, what action triggered this violation, and at what time this violation happened, login to the target's Database Vault Home Page and use the Attempted Violations charts.

Database Vault Attempted Violations - Realms

The metrics in this category monitor the violation attempts against the Database Vault database.

Database Vault Attempted Violations - Realms

This metric is used to enable Database Vault Security Analyst to keep a watch on the violation attempts against the Database Vault database. Database Vault Security Analysts can pick the realms that they would like to get alerted on and even further filter them based on the different types of attempts by mentioning different thresholds to match SQL commands causing the violations.

This metric is not enabled out of the box. You must enable it from Metrics and Policy Settings page. By default, this metric is collected every 1 hour, but you can specify the collection frequency.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Hour	Not Defined	Not Defined	%ACTION_OBJECT_NAME% got violated at %VIOLATIONTIMESTAMP%

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Database Vault Realm and Violation Time objects.

If warning or critical threshold values are currently set for any unique combination of Database Vault Realm and Violation Time objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Database Vault Realm and Violation Time objects, use the Edit Thresholds page.

Data Source

The attempted violations are picked up from the target's Database Vault audit trail. Only audit entries related to realms, which represent failed attempts to execute a SQL, are selected. Specified thresholds should match the SQL command causing command rule violation.

User Action

To know more about the violations, for example, the realm that was violated, which database user triggered the violation, what action triggered this violation, and at what time this violation happened, login to the target's Database Vault Home Page and use the Attempted Violations charts.

Database Vault Configuration Issues - Command Rules

The metrics in this category track users' actions and raise alerts when there is a misconfiguration on a command rule that requires administrator attention.

DV (Command Rule) - Configuration Issue Count

After the Database Vault policies are defined and configured to protect the database, further user actions over the course of time can disturb these configurations. This metric tracks the users' actions and raises an alert when there is a misconfiguration on a Command Rule that needs administrator attention. This metric is enabled out of the box. By default this metric is collected every 1 hour, but you can change the collection frequency.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Hour	Not Defined	0	%ACTION_OBJECT_NAME% has configuration issues.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Database Vault Command Rule object.

If warning or critical threshold values are currently set for any Database Vault Command Rule object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Database Vault Command Rule object, use the Edit Thresholds page.

Data Source

The configuration issues are picked from scanning the realm and command rule definitions.

User Action

To know the cause of the command rule misconfiguration, navigate to the target's Database Vault Home page, launch Database Vault Administrator, and view the Database Vault Configuration Issues Reports. These alerts are automatically cleared when the configuration issue is resolved.

Database Vault Configuration Issues - Realms

The metrics in this category track users' actions and raise alerts when there is a misconfiguration on a realm that requires administrator attention.

Database Vault Configuration Issues Count - Realms

After the Database Vault policies are defined and configured to protect the database, further user actions over the course of time can disturb these configurations. This metric tracks the users' actions and raises an alert when there is a misconfiguration on a Realm that needs administrator attention. This metric is enabled out of the box. By default this metric is collected every 1 hour, but you can change the collection frequency.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Hour	Not Defined	0	%ACTION_OBJECT_NAME% has configuration issues.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Database Vault Realm object.

If warning or critical threshold values are currently set for any Database Vault Realm object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Database Vault Realm object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

Data Source

The configuration issues are picked from scanning the realm and rule set definitions.

User Action

To know the cause of the realm misconfiguration, navigate to the target's Database Vault Home page, launch Database Vault Administrator, and view the Database Vault Configuration Issues Reports. These alerts are automatically cleared when the configuration issue is resolved.

Database Vault Policy Changes

The metrics in this category track the Database Vault policies.

Database Vault Policy Changes Count

After the Database Vault policies are defined, further changes to it is tracked by this metric. On any changes to the Database Vault policies, this metric will raise an alert. This metric is enabled out of the box. By default this metric is collected every 1 hour, but you can change the collection frequency.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Hour	Not Defined	0	%POLICY_CATEGORY_NAMES% has Policy changes

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of DV Policy Change Category and DV Policy Change Time objects.

If warning or critical threshold values are currently set for any unique combination of DV Policy Change Category and DV Policy Change Time objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of DV Policy Change Category and DV Policy Change Time objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page

Data Source

The policy changes are picked up from scanning the records in the Database Audit Trail related to Database Vault Schemas.

User Action

To know more about the policy changes, for example which object was changed, which database user changed the policy, what was the user action, and at what time this policy was changed, login to the target's Database Vault Home Page and view the Policy Changes Report.

Datafile Allocation

This section provides information on the metrics in the Datafile Allocation category.

The allocated space is the current size of the datafile. A portion of this allocated space is used to store data while some may be free space. The metrics in this category check the amount of space used and the amount of space allocated to each datafile. The used space can then be compared to the allocated space to determine how much space is unused in the datafile. This metric is not intended for alerts. Rather it is intended for reporting. Historical views of unused allocated free space can help DBAs to correctly size their datafiles, eliminating wasted space.

Target Version	Evaluation and Collection Frequency
All versions	Every 24 hours

Metric Name	Description	Data Source
Allocated File Size (MB)	The space allocated to the datafile. This metric should be used in conjunction with the Used File Size (MB) metric to produce a historical view of the amount of space being used and unused in each datafile.	• Datafile: dba_data_files • Tempfile: dba_temp_files

Metric Name	Description	Data Source
Used File Size (MB)	The space used in the datafile. This metric should be used in conjunction with the Allocated File Size (MB) metric to produce a historical view of the amount of space being used and unused in each datafile.	• Datafile: Subtract allocated free space (dba_lmt_free_space and dba_dmt_free_space) from allocated file size (dba_data_files) • Tempfile: gv$temp_extent_pool.bytes_used

DB Alert Log

The metrics in this category are used to create alerts by parsing the database alert log, for example, data block corruption, terminated session, and so on. The DB alert log metrics raise an alert containing the error text and, when relevant, a link to the trace file for each ORA error that is reported in the alert log that matches the warning or critical thresholds defined for each category of error returned by the metric as defined in the Metrics and Policy Settings but does not match the Alert Log Filter Expression.

> **Note:**
>
> If there is more than one ORA-error with the same error code or combination of error codes in one collection, only one error is uploaded. Duplicates are eliminated. De-duplication of recurring events for the same issue into a single event across collections is done alone for this metric.

Alert Log Filter Expression

The Alert Log Filter Expression is used (at the discretion of the Enterprise Manager administrator responsible for that target) to prevent errors that can be ignored resulting in alerts being raised in Enterprise Manager. It is a Perl regular expression that is used to filter all rows returned by the Alert Log metric

The filtering takes place during the retrieval of errors from the Alert log and therefore no errors that match the expression are considered by either the Alert Log metric or, by definition, the Alert Log Error Status metric. Only those errors that do not match the Alert Log Filter Expression are compared against the Alert Log metric thresholds or counted for the Alert Log Error Status metric.

You can configure the Alert Log Filter Expression from several locations in Enterprise Manager for each target. To configure the Alert Log Filter Expression, do either of the following:

- Click the link next to **Alert Log** under **Diagnostic Summary** from the DB Target home page and then click **Generic Alert Log Error Monitoring Configuration** under **Related Links**.

- Use any of the **Metrics and Policy Settings** pages for configuring the thresholds for each category of each metric.

Archiver Alert Log Error

This metric signifies that an archiver error has occurred on the database being monitored, since the last sample time.

If the database is running in ARCHIVELOG mode, an alert is displayed when there is an archiver error (ORA-00257 and ORA-16038) and messages are written to the ALERT file. The ALERT file is a special trace file containing a chronological log of messages and errors.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes	Not Defined	ORA-	Archiver error occurred at time/line number: %timeLine%.

Data Source

The source of the data is `$ORACLE_HOME/sysman/admin/scripts/alertlog.pl` where `$ORACLE_HOME` refers to the home of the Oracle Management Agent.

User Action

Examine the ALERT log and archiver trace file for additional information. However, the most likely cause of this message is that the destination device is out of space to store the redo log file. Verify the device specified in the initialization parameter `ARCHIVE_LOG_DEST` is set up properly for archiving.

> **Note:**
>
> This event does not automatically clear because there is no automatic way of determining when the problem has been resolved. Therefore, you must manually clear the event after the problem is fixed.

Data Block Corruption Alert Log Error

This metric signifies that the database being monitored has generated a corrupted block error to the ALERT file since the last sample time. The ALERT file is a special trace file containing a chronological log of messages and errors. An alert event is triggered when data block corrupted messages (ORA-01157, ORA-01578, and ORA-27048) are written to the ALERT file.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes	Not Defined	ORA-	A data block was corrupted at time/line number: %timeLine%.

Data Source

The source of the data is `$ORACLE_HOME/sysman/admin/scripts/alertlog.pl` where `$ORACLE_HOME` refers to the home of the Oracle Management Agent.

User Action

Examine the ALERT log for additional information.

> **Note:**
>
> This event does not automatically clear because there is no automatic way of determining when the problem has been resolved. Therefore, you must manually clear the event after the problem is fixed.

Generic Alert Log Error

This metric signifies that the database being monitored has generated errors to the ALERT log file since the last sample time. The ALERT log file is a special trace file containing a chronological log of messages and errors. An alert event is triggered when Oracle Exception (ORA-006xx) messages are written to the ALERT log file. A warning is displayed when other ORA messages are written to the ALERT log file.

- For all supported databases monitored by Enterprise Manager release 10.2.0.4 Management Agent:

  Alert Log Filter - up to 1024 characters

  Warning or Critical Threshold - up to 256 characters

- For all supported databases monitored by Enterprise Manager release 10.2.0.5 Management Agent:

  Alert Log Filter - up to 4000 characters

  Warning or Critical Threshold - up to 4000 characters

Archiver error (ORA-00257) and data block corrupted (ORA-01578) messages are sent out as separate metrics.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes	ORA-0*(600?|7445|4[0-9][0-9][0-9])[^0-9]	Not Defined	ORA-error stack (%errCodes%) logged in %alertLogName%.

Media Failure Alert Log Error

This metric indicates that the database being monitored has generated a media failure error to the ALERT file since the last sample time.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes	Not Defined	ORA-	Media failure was detected at time/line number: %timeLine%.

Session Terminated Alert Log Error

This metric indicates that the database being monitored has generated a session terminated message to the ALERT file since the last sample time.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes	ORA-	Not Defined	A session was terminated at time/line number: %timeLine%.

DB Alert Log Error Status

The metrics in this category count the number of errors returned in each category by the DB Alert Log Error metric after the Alert Log Filter expression has been taken into account but without taking the thresholds of the DB Alert Log Error metric into account and raises an alert if the number is greater than that specified in the Warning or Critical thresholds for that category. Therefore, it is possible for no alert to be raised by the metrics in the DB Alert Log Error category but still for the DB Alert Log Error Status metric to fire (even if the thresholds defined for the DB Alert Log Error metric are not matched).

Archiver Alert Log Error Status

This metric reflects the number of Archiver alert log errors witnessed the last time Enterprise Manager scanned the Alert Log.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes	0	Not Defined	%value% distinct types of ORA- errors have been found in the alert log.

Data Block Corruption Alert Log Error Status

This metric reflects the number of Data Block Corruption alert log errors witnessed the last time Enterprise Manager scanned the Alert Log.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes	0	Not Defined	Data block corruption errors have been found in the alert log.

Generic Alert Log Error Status

This metric reflects the number of Generic alert log errors witnessed the last time Enterprise Manager scanned the Alert Log.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes	0	Not Defined	%value% distinct types of ORA- errors have been found in the alert log.

Media Failure Alert Log Error Status

This metric represents the media failure alert log error status.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes	0	Not Defined	Media failure errors have been found in the alert log.

Session Terminated Alert Log Error Status

This metric reflects the number of Session Terminated alert log errors witnessed the last time Enterprise Manager scanned the Alert Log.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes	0	Not Defined	Session terminations have been found in the alert log.

DB Managed by Single Instance

The metrics in this category collect the configuration information for an Oracle Database for Single Instance High Availability (HA) registration.

CRS Home Directory

This metric reports on the Oracle Home directory if a Single Instance HA is installed on the machine.

DB Managed by Single Instance HA

This metric indicates whether the database is managed by Single Instance HA. If the Oracle Database is not managed by Single Instance HA, indicates if a Single Instance HA is available for Oracle Database registration.

Deferred Transactions

The metrics in this category are associated with this distributed database's deferred transactions.

Deferred Transaction Count

Oracle uses deferred transactions to propagate data-level changes asynchronously among master sites in an advanced replication system as well as from an updatable snapshot to its master table.

This metric checks for the number of deferred transactions. An alert is generated if the number of deferred transactions exceeds the value specified by the threshold argument.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	100	Not Defined	Number of deferred transactions is %value%.

Data Source

The source of the data is the following formula:

```
SELECT count(*) FROM sys.deftran
```

User Action

When the advanced replication facility pushes a deferred transaction to a remote site, it uses a distributed transaction to ensure that the transaction has been properly committed at the remote site before the transaction is removed for the queue at the local site. If transactions are not being pushed to a given remote site, verify that the destination for the transaction was correctly specified. If you specify a destination database when calling *DBMS_DEFER_SYS.SCHEDULE_EXECUTION* using the *DBLINK* parameter or *DBMS_DEFER_SYS.EXECUTE* using the *DESTINATION* parameter, make sure the full database link is provided.

Wrong view destinations can lead to erroneous deferred transaction behavior. Verify the *DEFCALLEST* and *DEFTRANDEST* views are the definitions from the *CATREPC.SQL* not the ones from *CATDEFER.SQL*.

Deferred Transaction Error Count

Oracle uses deferred transactions to propagate data-level changes asynchronously among master sites in an advanced replication system as well as from an updatable snapshot to its master table. If a transaction is not successfully propagated to the remote site, Oracle rolls back the transaction, logs the transaction in the SYS.DEFERROR view in the remote destination database.

This metric checks for the number of transactions in SYS.DEFERROR view and raises an alert if it exceeds the value specified by the threshold argument.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	0	Not Defined	Number of deferred transactions with errors is %value%.

Data Source

The source of the data is the following formula:

```
SELECT count(*) FROM sys.deferror
```

User Action

An error in applying a deferred transaction may be the result of a database problem, such as a lack of available space in the table is to be updated or may be the result of an unresolved insert, update or delete conflict. The SYS.DEFERROR view provides the ID of the transaction that could not be applied. Use this ID to locate the queued calls associated with the transaction. These calls are stored in the SYS.DEFCALL view. You can use the procedures in the DBMS_DEFER_QUERY package to determine the arguments to the procedures listed in the SYS.DEFCALL view.

Dump Area

The metrics in this category check for the percentage of used space of the dump destination devices.

Dump Area Directory

This metric reports the directory represented by this metric index's dump destination.

Each server and background process can write to an associated trace file to log messages and errors.

Background processes and the ALERT file are written to the destination specified by BACKGROUND_DUMP_DEST. Trace files for server processes are written to the destination specified by USER_ DUMP_DEST.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The source of the data is the v$parameter view.

User Action

Verify the device specified in the initialization parameters BACKGROUND_DUMP_DEST, USER_DUMP_DEST, and CORE_DUMP_DEST are set up properly for archiving.

If the BACKGROUND_DUMP_DEST, USER_DUMP_DEST, and CORE_DUMP_DEST initialization parameters are set up correctly and this metric triggers, then free up more space in the destination specified by the dump destination parameters.

Dump Area Used (%)

This metric returns the percentage of used space of the dump area destinations.

If the space used is more than the threshold value given in the threshold arguments, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	95	Not Defined	%value%% of %dumpType% dump area is used.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Type of Dump Area object.

If warning or critical threshold values are currently set for any Type of Dump Area object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Type of Dump Area object, use the Edit Thresholds page.

Data Source

Calculated using the UNIX `df -k` command.

- Critical threshold: Percentage of free space threshold for critical alert.
- Warning threshold: Percentage of free space threshold for warning alert.

User Action

Verify the device specified in the initialization parameters BACKGROUND_DUMP_DEST, USER_DUMP_DEST, and CORE_DUMP_DEST are set up properly for archiving.

If the BACKGROUND_DUMP_DEST, USER_DUMP_DEST, and CORE_DUMP_DEST initialization parameters are set up correctly and this metric triggers, then free up more space in the destination specified by the dump destination parameters.

Dump Area Used (KB)

This metric represents the total space used (in KB) on the device containing the dump destination directory.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The the data is calculated using the UNIX `df -k` command.

User Action

Verify the device specified in the initialization parameters *BACKGROUND_DUMP_DEST, USER_DUMP_DEST*, and *CORE_DUMP_DEST* are set up properly for archiving.

If the *BACKGROUND_DUMP_DEST, USER_DUMP_DEST*, and *CORE_DUMP_DEST* initialization parameters are set up correctly and this metric triggers, then free up more space in the destination specified by the dump destination parameters.

Free Dump Area (KB)

Each server and background process can write to an associated trace file in order to log messages and errors. Background processes and the ALERT file are written to the destination specified by BACKGROUND_DUMP_DEST.

Trace files for server processes are written to the destination specified by USER_DUMP_DEST.

This metric checks for available free space on these dump destination devices. If the space available is less than the threshold value given in the threshold arguments, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	%value% free KB remains in %dumpType% dump area.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Type of Dump Area object.

If warning or critical threshold values are currently set for any Type of Dump Area object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Type of Dump Area object, use the Edit Thresholds page.

Data Source

The data is calculated using the UNIX `df -k` command.

User Action

Verify the device specified in the initialization parameters BACKGROUND_DUMP_DEST, USER_DUMP_DEST, and CORE_DUMP_DEST are set up properly for archiving.

If the BACKGROUND_DUMP_DEST, USER_DUMP_DEST, and CORE_DUMP_DEST initialization parameters are set up correctly and this metric triggers, then free up more space in the destination specified by the dump destination parameters.

Total Dump Area (KB)

This metric represents the total space (in KB) available on the device containing the dump destination directory.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data is calculated using the UNIX `df -k` command.

User Action

Verify the device specified in the initialization parameters BACKGROUND_DUMP_DEST, USER_DUMP_DEST, and CORE_DUMP_DEST are set up properly for archiving.

If the BACKGROUND_DUMP_DEST, USER_DUMP_DEST, and CORE_DUMP_DEST initialization parameters are set up correctly and this metric triggers, then free up more space in the destination specified by the dump destination parameters.

Efficiency

This metric category contains the metrics that have traditionally been considered to represent the efficiency of some resource. Interpreting the wait interface is generally accepted as a much more accurate approach to measuring efficiency, and is recommended as an alternative to these hit ratios.

Buffer Cache Hit (%)

This metric represents the data block buffer cache efficiency, as measured by the percentage of times the data block requested by the query is in memory.

Effective use of the buffer cache can greatly reduce the I/O load on the database. If the buffer cache is too small, frequently accessed data will be flushed from the buffer cache too quickly which forces the information to be re-fetched from disk. Because disk access is much slower than memory access, application performance will suffer. In addition, the extra burden imposed on the I/O subsystem could introduce a bottleneck at one or more devices that would further degrade performance.

This test checks the percentage of buffer requests that were already in buffer cache. If the value is less than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the ((DeltaLogicalGets - (DeltaPhysicalReads - DeltaPhysicalReadsDirect)) / DeltaLogicalGets) * 100 formula where:

- DeltaLogicalGets: difference in 'select value from v$sysstat where name='session logical reads'' between sample end and start
- DeltaPhysicalReads: difference in 'select value from v$sysstat where name='physical reads'' between sample end and start

User Action

A low buffer cache hit ratio means that the server must often go to disk to retrieve the buffers required to satisfy a query. The queries that perform the most physical reads lower the numerical value of this statistic. Typically queries that perform full table scans force large amounts of buffers into the cache, aging out other buffers that may be required by other queries later. The Top Sessions page sorted by Physical Reads will show the sessions performing the most reads and through further drilldown their associated queries can be identified. Similarly, the Top SQL page sorted by Physical Reads shows which SQL statements are performing the most physical reads. The statements performing the most I/O should be looked at for tuning.

The difference between the two is that the Top Sessions chart shows the sessions that are responsible for the physical reads at any given moment. The Top SQL view shows all SQL that is still in the cache. The top statement may not be executing currently, and thus not responsible for the current poor buffer cache hit ratio.

If the queries seem to be well tuned, the size of the buffer cache also determines how often buffers must be fetched from disk. The DB_BLOCK_BUFFERS initialization parameter determines the number of database buffers available in the buffer cache. It is one of the primary parameters that contribute to the total memory requirements of the SGA on the instance. The DB_BLOCK_BUFFERS parameter, together with the DB_BLOCK_SIZE parameter, controls the total size of the buffer cache. Because DB_BLOCK_SIZE can only be specified when the database is first created, normally the size of the buffer cache size is controlled using the DB_BLOCK_BUFFERS parameter.

Consider increasing the DB_BLOCK_BUFFERS initialization parameter to increase the size of the buffer cache. This increase allows the Oracle Server to keep more information in memory, thus reducing the number of I/O operations required to do an equivalent amount of work using the current cache size.

CPU Usage (per second)

This metric represents the CPU usage per second by the database processes, measured in hundredths of a second. A change in the metric value may occur because of a change in either workload mix or workload throughput being performed by the database. Although there is no *correct* value for this metric, it can be used to detect a change in the operation of a system. For example, an increase in Database CPU usage from 500 to 750 indicates that the database is using 50% more CPU. (*No correct value* means that there is no single value that can be applied to any database. The value is a characteristic of the system and the applications running on the system.)

Target Version	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. ADDM can help to identify database operations that are consuming CPU. ADDM reports are available from a number of locations including the Database Home page and Advisor Central.

CPU Usage (per transaction)

This metric represents the average CPU usage per transaction expressed as a number of seconds of CPU time. A change in this metric can occur either because of changing workload on the system, such as the addition of a new module, or because of a change in the way that the workload is performed in the database, such as changes in the plan for a SQL statement. The threshold for this metric should be set based on the actual values observed on your system.

Target Version	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. ADDM will provide information about which operations are using the CPU resources.

Cursor Cache Hit (%)

This metric represents the percentage of soft parses satisfied within the session cursor cache.

Target Version	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is the following formula:

```
session cursor cache hits / (parse count (total) - parse count (hard))
```

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Data Dictionary Hit (%)

This metric represents dictionary cache efficiency as measured by the percentage of requests against the dictionary data that were already in memory. It is important to determine whether the misses on the data dictionary are actually affecting the performance of the Oracle Server. The shared pool is an area in the SGA that contains the library cache of shared SQL requests, the dictionary cache, and the other cache structures that are specific to a particular instance configuration.

Misses on the data dictionary cache are to be expected in some cases. Upon instance startup, the data dictionary cache contains no data, so any SQL statement issued is likely to result in cache misses. As more data is read into the cache, the likelihood of cache misses should decrease. Eventually the database should reach a steady state in which the most frequently used dictionary data is in the cache. At this point, very few cache misses should occur. To tune the cache, examine its activity only after your application has been running.

This test checks the percentage of requests against the data dictionary that were found in the Shared Pool. If the value is less than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is (1 - Misses/Gets) * 100 where:

- Misses: select sum(getmisses) from v$rowcache
- Gets: select sum(gets) from v$rowcache

User Action

If the percentage of gets is below 90% to 85%, consider increasing SHARED_POOL_SIZE to decrease the frequency in which dictionary data is being flushed from the shared pool to make room for new data. To increase the memory available to the cache, increase the value of the initialization parameter SHARED_POOL_SIZE.

Database CPU Time (%)

This metric represents the percentage of database call time that is spent on the CPU. Although there is no *correct* value for this metric, it can be used to detect a change in the operation of a system, for example, a drop in Database CPU time from 50% to 25%. (*No correct value* means that there is no single value that can be applied to any database. The value is a characteristic of the system and the applications running on the system.)

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

Not available.

User Action

Investigate if the change is CPU usage by using Automatic Database Diagnostic Monitor (ADDM). ADDM reports are available from a number of locations including the Database Home page and Advisor Central. Examine the report for increased time spent in wait events.

Library Cache Hit (%)

This metric represents the library cache efficiency, as measured by the percentage of times the fully parsed or compiled representation of PL/SQL blocks and SQL statements are already in memory.

The shared pool is an area in the SGA that contains the library cache of shared SQL requests, the dictionary cache and the other cache structures that are specific to a particular instance configuration.

The shared pool mechanism can greatly reduce system resource consumption in at least three ways: Parse time is avoided if the SQL statement is already in the shared pool.

Application memory overhead is reduced, because all applications use the same pool of shared SQL statements and dictionary resources.

I/O resources are saved, because dictionary elements that are in the shared pool do not require access.

If the shared pool is too small, users will consume additional resources to complete a database operation. For library cache access, the overhead is primarily the additional CPU resources required to re-parse the SQL statement.

This test checks the percentage of parse requests where cursor already in cache If the value is less than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is the (DeltaPinHits / DeltaPins) * 100 formula where:

- DeltaPinHits: difference in 'select sum(pinhits) from v$librarycache' between sample end and start
- DeltaPins: difference in 'select sum(pins) from v$librarycache' between sample end and start

User Action

The Top Sessions page sorted by Hard Parses lists the sessions incurring the most hard parses. Hard parses occur when the server parses a query and cannot find an exact match for the query in the library cache. You can avoid hard parses by sharing SQL statements efficiently. The use of bind variables instead of literals in queries is one method to increase sharing.

By showing you which sessions are incurring the most hard parses, this page can identify the application or programs that are the best candidates for SQL rewrites.

Also, examine SQL statements that can be modified to optimize shared SQL pool memory use and avoid unnecessary statement reparsing. This type of problem is commonly caused when similar SQL statements are written which differ in space, case, or some combination of the two. You may also consider using bind variables rather than explicitly specified constants in your statements whenever possible.

The SHARED_POOL_SIZE initialization parameter controls the total size of the shared pool. Consider increasing the SHARED_POOL_SIZE to decrease the frequency in which SQL requests are being flushed from the shared pool to make room for new requests.

To take advantage of the additional memory available for shared SQL areas, you may also need to increase the number of cursors permitted per session. You can increase this limit by increasing the value of the initialization parameter OPEN_CURSORS.

Library Cache Miss (%)

This metric represents the percentage of parse requests where the cursor is not in the cache.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

```
The source of the data is the following formula:
1 - pinhits / pins
```

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Parallel Execution Downgraded 25% or more (per second)

Number of times per second parallel execution was requested and the degree of parallelism was reduced to 25% and more because of insufficient parallel execution servers.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is the following formula:

```
(parallel operations downgraded 25 to 50 percent
+ parallel operations downgraded 50 to 75 percent
+ parallel operations downgraded 75 to 99 percent)
/ time
```

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Parallel Execution Downgraded 50% or more (per second)

This metric reports the number of times per second parallel execution was requested and the degree of parallelism was reduced to 50% and more because of insufficient parallel execution servers.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is the following formula:

```
(parallel operations downgraded 50 to 75 percent
+ parallel operations downgraded 75 to 99 percent)
/ time
```

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Parallel Execution Downgraded 75% or more (per second)

This metric reports the number of times per second parallel execution was requested and the degree of parallelism was reduced to 75% or more because of insufficient parallel execution servers.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is the following formula:

```
(parallel operations downgraded 75 to 99 percent) / time
```

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Parallel Execution Downgraded to Serial (per second)

This metric reports the number of times per second parallel execution was requested but execution was serial because of insufficient parallel execution servers.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is the following formula:

parallel operations downgraded to serial / time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Parallel Execution Downgraded to Serial (per transaction)

This metric reports the number of times per transaction parallel execution was requested but execution was serial because of insufficient parallel execution servers.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 minutes	Not Defined	Not Defined	Not Defined

Data Source

The source of the data is the following formula:

parallel operations downgraded to serial / transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

PGA Cache Hit (%)

This metric represents the total number of bytes processed in the PGA versus the total number of bytes processed plus extra bytes read/written in extra passes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Redo Log Allocation Hit (%)

Redo log entries contain a record of changes that have been made to the database block buffers. The log writer (LGWR) process writes redo log entries from the log buffer to a redo log file. The log buffer should be sized so that space is available in the log buffer for new entries, even when access to the redo log is heavy. When the log buffer is undersized, user process will be delayed as they wait for the LGWR to free space in the redo log buffer.

The redo log buffer efficiency, as measured by the hit ratio, records the percentage of times users did not have to wait for the log writer to free space in the redo log buffer.

This metric monitors the redo log buffer hit ratio (percentage of success) against the values specified by the threshold arguments. If the number of occurrences is smaller than the values specified, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is the following formula:

```
100 * (redo_entries_delta - redo_space_requests_delta)
/redo_entries_delta
where:
```

- redo_enties_delta = difference between "SELECT value FROM v$sysstat WHERE name = 'redo entries'" at the beginning and ending of the interval
- redo_space_requests_delta = difference between "SELECT value FROM v$sysstat WHERE name = 'redo log space requests'" at the beginning and ending of the interval

User Action

The LOG_BUFFER initialization parameter determines the amount of memory that is used when buffering redo entries to the redo log file.

Consider increasing the LOG_BUFFER initialization parameter in order to increase the size of the redo log buffer. Redo log entries contain a record of the changes that have been made to the database block buffers. The log writer process (LGWR) writes redo log entries from the log buffer to a redo log. The redo log buffer should be sized so space is available in the log buffer for new entries, even when access to the redo log is heavy.

Response Time (per transaction)

This metric represents the time spent in database operations per transaction. It is derived from the total time that user calls spend in the database (DB time) and the number of commits and rollbacks performed. A change in this value indicates that either the workload has changed or that the database's ability to process the workload has changed because of either resource constraints or contention.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page. Changes in the response time per transaction will appear as increased time spent in the database, either on CPU or in wait events and ADDM will report the sources of contention for both hardware and software resources.

Row Cache Miss Ratio (%)

This metric represents the percentage of row cache miss ratio.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every10 Minutes	Not Defined	Not Defined	Management Agent generates alert message.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Sorts in Memory (%)

This metric represents the sort efficiency as measured by the percentage of times sorts were performed in memory as opposed to going to disk.

For best performance, most sorts should occur in memory because sorts to disks are less efficient. If the sort area is too small, extra sort runs will be required during the sort operation. This increases CPU and I/O resource consumption.

This test checks the percentage of sorts performed in memory rather than to disk. If the value is less than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is (DeltaMemorySorts / (DeltaDiskSorts + DeltaMemorySorts)) * 100 where:

- DeltaMemorySorts: difference in 'select value from v$sysstat where name='sorts (memory)'' between sample end and start
- DeltaDiskSorts: difference in 'select value from v$sysstat where name='sorts (disk)'' between sample end and start

User Action

The sessions that are performing the most sorts should be identified such that the SQL they are executing can be further identified. The sort area sizes for the database may be sized correctly, and the application SQL may be performing unwanted or excessive sorts. The sessions performing the most sorts are available through the Top Sessions page sorted by Disk Sorts.

Further drilldown into the session performing the most disk sorts with the Current SQL page shows you the SQL statement responsible for the disk sorts.

The Top SQL page sorted by Sorts provides a mechanism to quickly display the SQL statements in the cache, presented in sorted order by their number sort operations. This is an alternative to viewing a sort of current sessions. It allows you to view sort activity via SQL statements and contains cumulative statistics for all executions of that statement.

If excessive sorts are taking place on disk and the queries are correct, consider increasing the SORT_AREA_SIZE initialization parameter to increase the size of the sort area. A larger sort area allows the Oracle Server to maintain sorts in memory, reducing the number of I/O operations required to do an equivalent amount of work using the current sort area size.

Exadata Module Version Failure

This metric category provides information about the amount of times an Exadata module version error occurs.

Error Count

This metric displays the number of times that the error occurred.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 24 Hours	0	Not Defined	%errorCode% occurrences of %errorCount%.

Failed Logins

The metrics in this category check for the number of failed logins on the target database. This check is performed every interval specified by the collection frequency and returns the number of failed logins for the last 30 minutes. These metrics will only work for databases where the audit_trail initialization parameter is set to DB or XML and the session is being audited.

Failed Login Count

This metric checks for the number of failed logins on the target database. This check is performed every interval specified by the collection frequency and returns the number of failed logins for the last 30 minutes. This metric will only work for databases where the audit_trail initialization parameter is set to DB or XML and the session is being audited.

If the failed login count crosses the values specified in the threshold arguments, then a warning or critical alert is generated. Because it is important to know every time a significant number of failed logins occurs on a system, on every collection, this metric determines the number of failed login attempts in the last 30 minutes and overrides the current alert instead of a new alert. You can manually clear these alerts. They will not automatically cleared after the next collection.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 30 Minutes	150	300	There have been %value% failed login attempts in the last %failed_login_interval_min% minutes.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Time object.

If warning or critical threshold values are currently set for any Time object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Time object, use the Edit Thresholds page.

Data Source

The database stores login information in different views, based on the audit_trail setting. The database view used is DB or DB_EXTENDED: DBA_AUDIT_SESSION.

User Action

No user action is required.

Fast Recovery

The Fast Recovery metrics relate to the fast recovery area.

Fast Recovery Area

Formerly referred to as flash recovery area, the metrics in this category return an optional disk location that you can use to store recovery-related files such as control file and online redo log copies, archived redo log files, flashback logs, and RMAN backups.

Oracle Database and RMAN manage the files in the fast recovery area automatically. You can specify the disk quota, which is the maximum size of the fast recovery area.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The source of the data is the following formula:

```
SELECT value
  FROM v$parameter
  WHERE name='db_recovery_file_dest';
```

User Action

No user action is required.

Fast Recovery Area Size

This metric returns the Fast Recovery Area Size.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The source of the data is the following formula:

```
SELECT value
  INTO 1_fast_recovery_size
  FROM v$parameter
  WHERE name='db_recovery_file_dest_size';
```

User Action

No user action is required.

Flashback On

This metric returns whether or not flashback logging is enabled - YES, NO, or RESTORE POINT ONLY. For the RESTORE POINT ONLY option, flashback is ON but you can only flashback to guaranteed restore points.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The source of the data is the following formula:

```
SELECT flashback_on
  FROM v$database;
```

User Action

No user action is required.

Log Mode

This metric returns the log mode of the database - ARCHIVELOG or NOARCHIVELOG.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The source of the data is the following formula:

```
SELECT log_mode
  FROM v$database;
```

User Action

No user action is required.

Non-Reclaimable Fast Recovery Area (%)

This metric represents the percentage of space non-reclaimable (spaced used minus space reclaimable) in the fast recovery area.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The source of the data is the following formula:

```
Non-reclaimable = space used - space reclaimable

Space Used:
  SELECT SUM(PERCENT_SPACE_USED
     FROM v$fast_recovery_area_usage;

Space Reclaimable:
  SELECT SUM(PERCENT_SPACE_RECLAIMABLE)
     FROM v$fast_recovery_area_usage;
```

User Action

No user action is required.

Oldest Flashback Time

This metric returns the oldest point-in-time to which you can flashback your database.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The source of the data is the following formula:

```
SELECT to_char(oldest_flashback_time, 'YYYY-MM-DD HH24:MI:SS')
  FROM v$flashback_database_log;
```

User Action

No user action is required.

Reclaimable Fast Recovery Area (%)

This metric represents the percentage of space reclaimable in the fast recovery area.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The source of the data is the following formula:

```
Space Reclaimable:
  SELECT SUM(PERCENT_SPACE_RECLAIMABLE)
     FROM v$fast_recovery_area_usage;
```

User Action

No user action is required.

Usable Fast Recovery Area (%)

This metric represents the percentage of space usable in the fast recovery area. The space usable is composed of the space that is free in addition to the space that is reclaimable.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The source of the data is the following formula:

```
SELECT (CASE WHEN PERCENT_USED > 100 THEN 0 ELSE (100-PERCENT_USED) END)
PERCENT_FREE
   FROM (SELECT (SUM(PERCENT_SPACE_USED)-SUM(PERCENT_SPACE_RECLAIMABLE))
PERCENT_USED
   FROM V$FAST_RECOVERY_AREA_USAGE);
```

User Action

No user action is required.

Fragmented Text Indexes

This metric category represents the number of text indexes in the database fragmented beyond the warning and critical percentage thresholds specified by the user. The collection is disabled by default. Before enabling this metric and specifying a metric threshold for the number of text indexes, the "Evaluate and Fix Text Index Fragmentation" job should be submitted against the database target. The following details could be specified as part of the job parameters:

- Warning/Critical percentage threshold against which the text indexes are to be evaluated.
- List of text indexes to be evaluated (all indexes, specific schemas, or list of fully qualified names).

- List of text indexes to be fixed (all indexes, specific schemas, or list of fully qualified names). The scheduled DBMS job would attempt to fix the fragmented text indexes by optimizing (if warning threshold exceeded) or rebuilding them (if critical threshold exceeded, using shadow creation).
- The DBMS job schedule.

Fragmented Text Index count

This metric collects the total number of text indexes that have crossed the fragmentation percentage threshold specified by the user.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 hours	NA	NA	NA

Fragmented Text Index count crossing critical threshold

This metric collects the number of text indexes that have crossed the critical fragmentation percentage threshold specified by the user.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 hours	Not Defined	Not Defined	Fragmented Text Index count crossing critical threshold is %value%

Data Source

The fragmentation percentage for each index or index partition is derived by computing the data from DBA_IND_PARTITIONS, CTXSYS.CTX_INDEX_PARTITIONS, and its relevant text index metadata tables. The list of text indexes and the critical percentage threshold against which their fragmentation is to be evaluated are specified by the user as part of the "Evaluate and Fix Text Index Fragmentation" job.

User Action

A metric threshold could be set to generate incidents on the number of text indexes that have crossed the critical fragmentation threshold specified in "Evaluate and Fix Text Index Fragmentation" job. The scheduled DBMS job would automatically attempt to fix such text indexes (if they were specified in the fix list) by rebuilding them (using shadow creation). In addition, the incident also enables the user to fix the fragmented text indexes from the Enterprise Manager console.

Fragmented Text Index count crossing warning threshold

This metric collects the total number of text indexes that have crossed the warning fragmentation percentage threshold, but not the critical percentage threshold, specified by the user.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 hours	Not Defined	Not Defined	Fragmented Text Index count crossing warning threshold is %value%

Data Source

The fragmentation percentage for each index or index partition is derived by computing the data from DBA_IND_PARTITIONS, CTXSYS.CTX_INDEX_PARTITIONS, and its relevant text index metadata tables. The list of text indexes and the warning percentage threshold against which their fragmentation is to be evaluated are specified by the user as part of the "Evaluate and Fix Text Index Fragmentation" job.

User Action

A metric threshold could be set to generate incidents on the number of text indexes that have crossed the warning fragmentation threshold, but not the critical threshold, specified in "Evaluate and Fix Text Index Fragmentation" job. The scheduled DBMS job would automatically attempt to fix such text indexes (if they were specified in the fix list) by optimizing them. In addition, the incident also enables the user to fix the fragmented text indexes from the Enterprise Manager console.

Global Cache Statistics

The metrics in this category are associated with global cache statistics.

Global Cache Average CR Block Request Time (centi-seconds)

This metric represents the average time, measured in hundredths of a second, that CR block was received.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	1	2	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is the following formula:

global cache CR block receive time * 10 / global cache current blocks received

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Global Cache Average Current Block Request Time (centi-seconds)

This metric represents the average time, measured in hundredths of a second, to get a current block.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	1	2	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is the following formula:

global cache current block send time * 10 / global cache current blocks served

User Action

The required actions are specific to your site.

Global Cache Blocks Corrupt

This metric represents the number of blocks that encountered a corruption or checksum failure during interconnect over the user-defined observation period.

> **Note:**
>
> Unlike most metrics, which accept thresholds as real numbers, this metric can only accept an integer as a threshold.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	0	0	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is the following formula:

global cache blocks corrupted

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Global Cache Blocks Lost

This metric represents the number of global cache blocks lost over the user-defined observation period.

> **Note:**
>
> Unlike most metrics, which accept thresholds as real numbers, this metric can only accept an integer as a threshold.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	1	3	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The source of the data is global cache blocks lost.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

High Availability (RMAN Configuration)

This section provides information on the metrics in the High Availability (RMAN Configuration) category, which lists RMAN persistent configuration settings.

Target Version	Evaluation and Collection Frequency
All versions	Every 24 hours

Metric Name	Description
Conf Number	A unique key identifying the RMAN configuration record within the target database that owns it.
Name	The name or type of configuration.
Value	The `CONFIGURE` command setting. Example, `RETENTION POLICY TO RECOVERY WINDOW OF 10 DAYS`.
Container ID	The ID of the container to which the data (the Conf Number, Name, and Value) pertains. Possible values include: • `0`: This value is used for rows containing data that pertains to the entire Container Database (CDB). This value is also used for rows in non-CDBs. • `1`: This value is used for rows containing data that pertains to only the root. • `n`: This value is used where `n` is the applicable container ID for the rows containing data.

High Availability Backup

This section provides information on the metrics in the High Availability Backup category.

Target Version	Evaluation and Collection Frequency
All versions	Every 12 hours

Metric Name	Description	Data Source
Command ID	The unique command ID of the backup command corresponding to this backup job.	v$rman_backup_job_details
End Time	The end time of the last backup command in the job.	v$rman_backup_job_details
Size of Input Files	The sum of all input file sizes backed up by this job, expressed as a string.	v$rman_backup_job_details
Input Type	The input type, which contains one of the following values: DB FULL, RECVR AREA, DB INCR, DATAFILE FULL, DATAFILE INCR, ARCHIVELOG, CONTROLFILE, SPFILE.	v$rman_backup_job_details
Size of Output Files	The output size of all pieces generated by this job, expressed as a string.	v$rman_backup_job_details
Output Bytes Per Sec	The generation rate of the output pieces for this backup, expressed as a string.	v$rman_backup_job_details
Output Device Type	The media device type for this backup job.	v$rman_backup_job_details
Session Key	The session key for this backup job.	v$rman_backup_job_details
Session RECID	The session RECID for this backup job.	v$rman_backup_job_details
Session Stamp	The session stamp for this backup job.	v$rman_backup_job_details
Start Time	The start time of the first backup command in the job.	v$rman_backup_job_details
Status	The status of the backup job.	v$rman_backup_job_details
Time Taken	The time taken for this backup job.	v$rman_backup_job_details

High Availability Backup History

This section provides information on the metrics in the High Availability Backup History category.

> **Note:**
>
> These metrics are not enabled out of the box, and must be enabled on the **Metric and Collection Settings** page. By default, these metrics are collected every 12 hours, but you can change the collection frequency.

Target Version	Evaluation and Collection Frequency
12c and later	Every 12 hours

Metric Name	Description	Data Source
Oracle Storage Container	If the backup is on Oracle Cloud storage, this specifies the Cloud storage container where the backup is located.	v$backup_piece_details
Compressed	Indicates whether the backup piece is compressed.	v$backup_piece_details
Compression Ratio	If the backup is compressed, this specifies the compression ratio of the backup.	v$rman_backup_job_details
Container ID	If the backup is on Oracle Cloud storage, this specifies the ID of the Cloud storage container where the backup is located.	v$rman_backup_job_details
Elapsed Seconds	The number of seconds that the backup job took to complete.	v$rman_backup_job_details
Encrypted	Indicates whether the backup was encrypted (YES or NO).	v$backup_piece_details
End Time	The end time of the last backup command in the job.	v$rman_backup_job_details
Incremental Level	Indicates the incremental level of this backup set.	v$backup_set_details
Size of Input Files (bytes)	The sum of all input file sizes backed up by this job, in bytes.	v$rman_backup_job_details
Size of Input Files	The sum of all input file sizes backed up by this job, expressed as a string.	v$rman_backup_job_details
Input Type	The input type, which contains one of the following values: DB FULL, RECVR AREA, DB INCR, DATAFILE FULL, DATAFILE INCR, ARCHIVELOG, CONTROLFILE, SPFILE.	v$rman_backup_job_details
Keep	Indicates whether or not this backup set has a retention policy.	v$backup_set_details
Keep Options	The additional retention options for this backup set.	v$backup_set_details
Keep Until	Indicates the date after which the backup becomes obsolete.	v$backup_set_details
Media	The name of the media on which the backup piece resides.	v$backup_piece_details
Name	The unique command ID for the backup command corresponding to this backup job.	v$rman_backup_job_details
Size of Output Files (bytes)	The output size of all pieces generated by this job, in bytes.	v$rman_backup_job_details
Size of Output Files	The output size of all pieces generated by this job, expressed as a string.	v$rman_backup_job_details
Output Bytes Per Sec (bytes/sec)	The generation rate of the output pieces for this backup, in bytes/second.	v$rman_backup_job_details
Output Bytes Per Sec	The generation rate of the output pieces for this backup, expressed as a string.	v$rman_backup_job_details
Output Device Type	The media device type for this backup job.	v$rman_backup_job_details
Session Key	The session key for this backup job.	v$rman_backup_job_details
Session RECID	The session RECID for this backup job.	v$rman_backup_job_details

Metric Name	Description	Data Source
Session Stamp	The session stamp for this backup job.	v$rman_backup_job_details
Start Time	The start time of the first backup command in the job.	v$rman_backup_job_details
Status	The status of the backup job.	v$rman_backup_job_details
Tag	The tag for this backup job.	v$backup_piece_details
Time Taken	The time taken for this backup job.	v$rman_backup_job_details

High Availability Client Recovery Window

This section provides information on the metrics in the High Availability Client Recovery Window category.

Note that the data source for these metrics is the database control file, and the collection for these metrics is disabled by default. If the database is backing up to a Recovery Appliance, these metrics are not applicable and the collection should remain disabled. If the database is not backing up to a Recovery Appliance and you want to monitor the database recovery window, you can enable the collection. In this case, the data in these metrics is used as the source data for the related High Availability Recovery Window metric category. See High Availability Recovery Window.

Target Version	Evaluation and Collection Frequency
12c and later	Every 15 minutes

Metric Name	Description	Data Source
Disk Recovery Window (seconds)	The recovery window for disk backups.	v$disk_restore_range
Disk Unprotected Data Window (seconds)	The current amount of potential data loss for disk backups.	v$disk_restore_range
Last Complete Disk Backup Date	The latest point in time for which a complete disk backup is available for all datafiles in this database.	v$disk_restore_range
Last Complete Media Backup Date	The latest point in time for which a complete media backup is available for all datafiles in this database.	v$sbt_restore_range
Media Recovery Window (seconds)	The recovery window for media backups.	v$sbt_restore_range
Media Unprotected Data Window (seconds)	The current amount of potential data loss for media backups.	v$sbt_restore_range

High Availability Data Guard Target Summary

This section provides information on the metrics in the High Availability Data Guard Target Summary category.

Target Version	Evaluation and Collection Frequency
All versions	Every 24 hours

Metric Name	Description
Source Type	The role (Primary or Standby) of the database that was the source of the data row.
Row Type	The role (Primary or Standby) of the database to which the data in the row pertains.
Using Broker	Whether the Data Guard configuration for the database specified by the row is using Data Guard broker.
Active Standby	Whether the database specified by the row is an Active Data Guard standby database.
Database Unique Name	The value of the DB_UNIQUE_NAME initialization parameter for the database specified by the row.
Database ID	The value of DBID (as found in v$database) of the database specified by the row.
Primary Database Unique Name	The Database Unique Name of the primary database associated with the database specified by the row (if the database is a standby database).
Primary Database ID	The Database ID of the primary database associated with the database specified by the row.
Role	The Data Guard role of the database specified by the row.
Standby Database List	The list of standby databases associated with the database specified by the row (if the database is a primary database).
Protection Mode	The protection mode for the database specified by the row.
Fast-Start Failover Status	The fast-start failover status for the database specified by the row.
Status	The Data Guard status for the database specified by the row.
Redo Source	The Database Unique Name of the database that is shipping redo to the database specified by the row.
Data Guard Connect Identifier	The net connect identifier used to reach the database.

High Availability Disk Backup

This section provides information on the metrics in the High Availability Disk Backup category.

Target Version	Evaluation and Collection Frequency
All versions	Every 30 minutes

Metric Name	Description	Data Source
Time Since Last Successful Full Backup (hours)	The time since the last successful full disk backup, in hours. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful full database disk backup was %value% hours ago.	v$rman_backup_job_details
Time Since Last Successful Incremental Backup (hours)	The time since the last successful incremental disk backup, in hours. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful incremental database disk backup was %value% hours ago.	v$rman_backup_job_details
Time Since Last Successful Archived Log Backup (minutes)	The time since the last successful archived log disk backup, in minutes. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful archived log disk backup was %value% minutes ago.	v$rman_backup_job_details
Last Executed Full Backup Status	The status of the last executed full disk backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed full database disk backup status was %value%.	v$rman_backup_job_details
Last Executed Incremental Backup Status	The status of the last executed incremental disk backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed incremental database disk backup status was %value%.	v$rman_backup_job_details
Last Executed Archived Log Backup Status	The status of the last executed archived log disk backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed archived log disk backup status was %value%.	v$rman_backup_job_details
Recovery Window (seconds)	This column is obsolete and is not populated. This data is now available in the High Availability Client Recovery Window metric category.	v$disk_restore_range
Last Successful Archived Log Backup Date	The date of the last successful archived log disk backup.	v$rman_backup_job_details
Last Successful Archived Log Backup Size (bytes)	The size of the last successful archived log disk backup, in bytes.	v$rman_backup_job_details
Last Complete Backup Date	The latest point in time for which a complete disk backup is available for all datafiles.	v$disk_restore_range
Last Successful Full Backup Date	The date of the last successful full disk backup.	v$rman_backup_job_details
Last Successful Full Backup Size (bytes)	The size of the last successful full disk backup, in bytes.	v$rman_backup_job_details

Metric Name	Description	Data Source
Last Executed Archived Log Backup Date	The date of the last executed archived log disk backup.	v$rman_backup_job_details
Last Executed Full Backup Date	The date of the last executed full disk backup.	v$rman_backup_job_details
Last Executed Incremental Backup Date	The date of the last executed incremental disk backup.	v$rman_backup_job_details
Last Executed Incremental Level 0 Backup Status	The status of the last executed incremental level 0 disk backup.	v$rman_backup_job_details
Last Executed Incremental Level 1 Backup Status	The status of the last executed incremental level 1 disk backup.	v$rman_backup_job_details
Last Successful Incremental Backup Date	The date of the last successful incremental disk backup.	v$rman_backup_job_details
Last Successful Incremental Backup Size (bytes)	The size of the last successful incremental disk backup, in bytes.	v$rman_backup_job_details
Time Since Last Successful Incremental Level 0 Backup (hours)	The time since the last successful incremental level 0 disk backup, in hours.	v$rman_backup_job_details
Last Successful Incremental Level 0 Backup Size (bytes)	The size of the last successful incremental level 0 disk backup, in bytes.	v$rman_backup_job_details
Time Since Last Successful Incremental Level 1 Backup (hours)	The time since the last successful incremental level 1 disk backup, in hours.	v$rman_backup_job_details
Last Successful Incremental Level 1 Backup Size (bytes)	The size of the last successful incremental level 1 disk backup, in bytes.	v$rman_backup_job_details
Unprotected Data Window (seconds)	This column is obsolete and is not populated. This data is now available in the High Availability Client Recovery Window metric category.	v$disk_restore_range

High Availability Media Backup

This section provides information on the metrics in the High Availability Media Backup category.

Target Version	Evaluation and Collection Frequency
All versions	Every 30 minutes

Metric Name	Description	Data Source
Time Since Last Successful Full Backup (hours)	The time since the last successful full media backup, in hours. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful full database media backup was %value% hours ago.	v$rman_backup_job_details
Time Since Last Successful Archived Log Backup (minutes)	The time since the last successful archived log media backup, in minutes. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful archived log media backup was %value% minutes ago.	v$rman_backup_job_details
Time Since Last Successful Incremental Backup (hours)	The time since the last successful incremental media backup, in hours. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful incremental database media backup was %value% hours ago.	v$rman_backup_job_details
Last Executed Archived Log Backup Status	The status of the last executed archived log media backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed archived log media backup status was %value%.	v$rman_backup_job_details
Last Executed Full Backup Status	The status of the last executed full media backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed full database media backup status was %value%.	v$rman_backup_job_details
Last Executed Incremental Backup Status	The status of the last executed incremental media backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed incremental database media backup status was %value%.	v$rman_backup_job_details
Recovery Window (seconds)	This column is obsolete and is not populated. This data is now available in the High Availability Client Recovery Window metric category.	v$sbt_restore_range
Last Successful Archived Log Backup Date	The date of the last successful archived log media backup.	v$rman_backup_job_details
Last Successful Archived Log Backup Media	The name of the media on which the last successful archived log backup resides.	v$backup_piece_details
Last Successful Archived Log Backup Size (bytes)	The size of the last successful archived log media backup, in bytes.	v$rman_backup_job_details
Last Complete Backup Date	The latest point in time for which a complete media backup is available for all datafiles.	v$sbt_restore_range
Last Successful Full Backup Date	The date of the last successful full media backup.	v$rman_backup_job_details

Metric Name	Description	Data Source
Last Successful Full Backup Media	The name of the media on which the last successful full backup resides.	v$backup_piece_details
Last Successful Full Backup Size (bytes)	The size of the last successful full media backup, in bytes.	v$rman_backup_job_details
Last Executed Archived Log Backup Date	The date of the last executed archived log media backup.	v$rman_backup_job_details
Last Executed Full Backup Date	The date of the last executed full media backup.	v$rman_backup_job_details
Last Executed Incremental Backup Date	The date of the last executed incremental media backup.	v$rman_backup_job_details
Last Executed Incremental Level 0 Backup Status	The status of the last executed incremental level 0 media backup.	v$rman_backup_job_details
Last Executed Incremental Level 1 Backup Status	The status of the last executed incremental level 1 media backup.	v$rman_backup_job_details
Last Successful Incremental Backup Date	The date of the last successful incremental media backup.	v$rman_backup_job_details
Last Successful Incremental Backup Media	The name of the media on which the last successful incremental backup resides.	v$backup_piece_details
Last Successful Incremental Backup Size (bytes)	The size of the last successful incremental media backup, in bytes.	v$rman_backup_job_details
Time Since Last Successful Incremental Level 0 Backup (hours)	The time since the last successful incremental level 0 media backup, in hours.	v$rman_backup_job_details
Last Successful Incremental Level 0 Backup Media	The name of the media on which the last successful incremental level 0 backup resides.	v$rman_backup_job_details
Last Successful Incremental Level 0 Backup Size (bytes)	The size of the last successful incremental level 0 media backup, in bytes.	v$rman_backup_job_details
Time Since Last Successful Incremental Level 1 Backup (hours)	The time since the last successful incremental level 1 media backup, in hours.	v$rman_backup_job_details
Last Successful Incremental Level 1 Backup Media	The name of the media on which the last successful incremental level 1 backup resides.	v$rman_backup_job_details
Last Successful Incremental Level 1 Backup Size (bytes)	The size of the last successful incremental level 1 media backup, in bytes.	v$rman_backup_job_details
Unprotected Data Window (seconds)	This column is obsolete and is not populated. This data is now available in the High Availability Client Recovery Window metric category.	v$sbt_restore_range

High Availability Recovery Window

This section provides information on the metrics in the High Availability Recovery Window category.

Note that the data source for these metrics depends on the database backup destination. If the database is backing up to a Recovery Appliance, all of the data is sourced from the Recovery Appliance, and metrics that are applicable only in the Recovery Appliance case are noted. If the database is not backing up to a Recovery Appliance, all data is sourced from the High Availability Client Recovery Window metric category, which in turn gets its data from the local database control file. In this case, if there is no data for these metrics, it may be because the High Availability Client Recovery Window collection is disabled. See High Availability Client Recovery Window.

Target Version	Evaluation and Collection Frequency
All versions	Every 15 minutes

Metric Name	Description	Data Source
Recovery Appliance Downstream One	The name of the first downstream Recovery Appliance that is receiving replicated backups for this database. (Note that this is applicable only to databases backing up to a Recovery Appliance.)	–
Recovery Appliance Downstream Two	The name of the second downstream Recovery Appliance that is receiving replicated backups for this database. (Note that this applicable only to databases backing up to a Recovery Appliance.)	–
Final Change Number	The highest SCN to which this database can be recovered when using backups and redo logs available on the Recovery Appliance. (Note that this is applicable only to databases backing up to a Recovery Appliance.)	rc_database on Recovery Appliance
Last Complete Disk Backup	The latest point in time for which a complete disk backup is available for all datafiles in this database. If the database is backing up to a Recovery Appliance, this is based on backups contained in Recovery Appliance disk storage.	v$disk_restore_range, if the database is configured to backup to a disk. ra_database, if the database is configured to backup to a Recovery Appliance.
Last Complete Media Backup	The latest point in time for which a complete media backup is available for all datafiles in this database. If the database is backing up to a Recovery Appliance, this is based on backups copied by the Recovery Appliance to tape or Cloud storage.	v$sbt_restore_range
Recovery Appliance	The Recovery Appliance that this database is currently backing up to, if any.	–
Recovery Appliance Replication Server List	The list of replication servers configured for this database on the Recovery Appliance. (Note that this is applicable only to databases backing up to a Recovery Appliance.)	ra_protection_policy on Recovery Appliance
Disk Recovery Window Goal (seconds)	The recovery window goal specified within the Recovery Appliance protection policy applicable to this database. (Note that this is applicable only to databases backing up to a Recovery Appliance.)	ra_protection_policy on Recovery Appliance

Metric Name	Description	Data Source
Disk Unprotected Data Window Threshold (seconds)	The unprotected data window threshold specified within the Recovery Appliance protection policy applicable to this database. (Note that this is applicable only to databases backing up to a Recovery Appliance.)	ra_database on Recovery Appliance
Disk Unprotected Data Window (seconds)	The current amount of potential data loss for disk backups. If the database is backing up to a Recovery Appliance, this is the amount of data not present in backups contained in Recovery Appliance disk storage. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The disk unprotected data window is %value% seconds.	v$disk_restore_range, if the database is configured to backup to a disk. ra_database, if the database is configured to backup to a Recovery Appliance.
Disk Recovery Window (seconds)	The current database recovery window for disk backups. If the database is backing up to a Recovery Appliance, this is the current disk recovery window reported for this database by the Recovery Appliance, based on backups contained in Recovery Appliance disk storage. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The disk recovery window is %value% seconds.	v$disk_restore_range, if the database is configured to backup to a disk. ra_disk_restore_range, if the database is configured to backup to a Recovery Appliance.
Media Recovery Window (seconds)	The current database recovery window for media backups. If the database is backing up to a Recovery Appliance, this is the current media recovery window reported for this database by the Recovery Appliance, based on backups copied by the Recovery Appliance to attached tape or Cloud storage. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The media recovery window is %value% seconds.	v$sbt_restore_range
Media Unprotected Data Window (seconds)	The current amount of potential data loss for media backups. If the database is backing up to a Recovery Appliance, this is the amount of data not present in backups copied by the Recovery Appliance to tape or Cloud storage. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The media unprotected data window is %value% seconds.	v$sbt_restore_range

Incident

This metric category contains the metrics representing incidents, such as generic internal error, or access violation, as recorded in the database alert log file. Incidents refer to problems for which Automatic Diagnostic Repository (ADR) incidents are created. These type of problems usually require investigation, diagnostic data to be collected, and possibly interaction with Oracle Support for resolution. The alert log file has a chronological log of messages and errors.

Each metric signifies that the database being monitored has detected a critical error condition about the database and has generated an incident to the alert log file since the last sample time. The Support Workbench in Enterprise Manager contains more information about each generated incident.

> **Note:**
>
> For more information about Incident metrics and Operational Error metrics, sign in to My Oracle Support and search for the following Oracle Support note:
>
> *Database Alert log monitoring in 12c explained (Doc ID 1538482.1)*
>
> https://support.oracle.com/

Setting Thresholds for Incident Metrics

To edit the thresholds for any of the following metrics, from the Enterprise Manager UI, right-click the target name, select **Monitoring**, then **Metric and Collection Settings**. The following settings provide examples of some of the possible settings:

- Warning Threshold: Not Defined; Critical Threshold: .*

  In this case, the Management Agent generates a critical error alert in Enterprise Manager when the incident occurs.

- Warning Threshold: .*; Critical Threshold: Not Defined

  In this case, the Management Agent generates a warning alert in Enterprise Manager when the incident occurs.

- Warning Threshold: Not Defined; Critical Threshold: Not Defined

  In this case, the Management Agent does *not* generate an alert in Enterprise Manager when the incident occurs.

Access Violation

This metric signifies that the database has generated an incident due to some memory access violation. This type of incident is typically related to Oracle Exception messages such as ORA-3113 and ORA-7445. The database can also generate this type of incident when it detects a SIGSEGV or SIGBUS signals.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	An access violation detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Alert Log Error Trace File

This metric reports the name of the trace file (if any) associated with the logged incident.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

No user action is required.

Alert Log Name

This metric reports the name of the alert log file.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

No user action is required.

Cluster Error

This metric signifies that the database has generated an incident due to a member evicted from the group by a member of the cluster database. This type of incident is typically related to Oracle Exception message ORA-29740.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	A cluster error detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Deadlock

This metric signifies that the database has generated an incident due to a deadlock detected while trying to lock a library object. This type of incident is typically related to Oracle Exception message ORA-4020.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	A deadlock detected in $alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

File Access Error

This metric signifies that the database has generated an incident due to failure to read a file at the time.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	A file access error detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Generic Incident

This metric signifies that the database has generated an incident due to some database error.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	Incident (%adr_problemKey%) detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Generic Internal Error

This metric signifies that the database has generated an incident due to an internal database error. This type of incident is typically related to Oracle Exception message ORA-600 or ORA-0060*.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	Internal error (%adr_problemKey%) detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Impact

This metric reports the impact of an incident. For a Generic Internal Error incident, the impact describes how the incident may affect the database.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The source of the data is `$AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_n.n.n.n/scripts/alertlogAdr.pl`.

In the preceding directory path, `$AGENT_BASE` refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

No user action is required.

Incident ID

This metric reports a number identifying an incident. The Support Workbench in Enterprise Manager uses this ID to specify an incident.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

No user action is required.

Inconsistent DB State

This metric signifies that the database has generated an incident due to an inconsistent database state such an invalid ROWID. This type of incident is typically related to Oracle Exception message ORA-1410.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	An inconsistent DB state detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Internal SQL Error

This metric signifies that the database has generated an incident due to an internal SQL error. This type of incident is typically related to Oracle Exception message ORA-604.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	An internal SQL error detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Oracle Data Block Corruption

This metric signifies that the database has generated an incident due to an ORACLE data block corruption. This type of incident is typically related to Oracle Exception message ORA-1578.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	An Oracle data block corruption detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Out of Memory

This metric signifies that the database has generated an incident due to failure to allocate memory. This type of incident is typically related to Oracle Exception message ORA-4030 or ORA-4031.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	Out of memory detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Redo Log Corruption

This metric signifies that the database has generated an incident due to an error with the redo log. This type of incident is typically related to Oracle Exception message ORA-353, ORA-355, or ORA-356.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	A redo log corruption detected in %alertLogName% at time/line number: %timeLine%/

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Session Terminated

This metric signifies that the database has generated an incident due to an unexpected session termination. This type of incident is typically related to Oracle Exception message ORA-603.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	A session termination detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of incident as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the incident.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Interconnect

The metrics in this category collect the information about network interfaces used by cluster database instances as internode communication.

Interface Type

Cluster database instances should use private interconnects for internode communication. This metric monitors whether the network interface used by the cluster instance is a private one. If the network interface is known to be public, a critical alert is generated. If the network interface type is unknown, a warning alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 12 Hours	Unknown	Public	The instance is using interface '%if_name%' of type '%value%'.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Interface Name object.

If warning or critical threshold values are currently set for any Interface Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Interface Name object, use the Edit Thresholds page.

Data Source

The data is derived from the following views:

V$CLUSTER_INTERCONNECTS

V$CONFIGURED_INTERCONNECTS

User Action

Use oifcfg in the CRS home to correctly configure the private interfaces in OCR.

Interconnect Traffic

The metrics in this category monitor the internode data transfer rate of cluster database instances.

Transfer Rate (MB/s)

This metric collects the internode communication traffic of a cluster database instance. This is an estimation using the following formula:

```
(gc cr blocks received/sec + gc current blocks received/sec + gc cr blocks served/sec +
gc current blocks served/sec) * db_block_size
+
( messages sent directly/sec + messages send indirectly/sec + messages received/sec ) *
200 bytes
```

The critical and warning thresholds of this metric are not set by default. Users can set them according to the speed of their cluster interconnects.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	Not Defined

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Instance Name object.

If warning or critical threshold values are currently set for any Instance Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Instance Name object, use the Edit Thresholds page.

Data Source

The data is derived from the following views:

V$SYSSTAT

V$DLM_MISC

V$PARAMETER

User Action

No user action is required.

Invalid Objects

The metrics in this category represent number of invalid objects in the database.

Invalid Object Count

This metric represents the total invalid object count in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Invalid Object Count in the database is %value%

Data Source

The data is derived from the SYS.OBJ$ and SYS.USER$ tables.

User Action

The “Recompile Invalid Objects” corrective action could be setup against the incident to automatically attempt to recompile the invalid objects in the database. Some objects might need specific corrective steps to be performed manually before re-compilation.

Invalid Objects by Schema

The metrics in this category represent the number of invalid objects in each schema.

Invalid Object Count by Schema

This metric represents the total number of invalid objects per schema.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Invalid Object Count in %owner% schema is %value%

Multiple Thresholds

Different warning and critical threshold values could be set for each Invalid Object Owner (schema) object.

If warning or critical threshold values are currently set for any Invalid Object Owner object, those thresholds could be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Invalid Object Owner object, use the Edit Thresholds page.

Data Source

The data is derived from the SYS.OBJ$ and SYS.USER$ tables.

User Action

The “Recompile Invalid Objects” corrective action could be setup against the incident to automatically attempt to recompile the invalid objects in a schema. Some objects might need specific corrective steps to be performed manually before recompilation.

Messages Per Buffered Queue

The metrics in this category monitor the age and state of the first (top of the queue) message for each buffered queue in the database except for the system queues. Queues that are in the schema of SYS, SYSTEM, DBSNMP, and SYSMAN are defined as system level queues.

Average Age of Messages Per Buffered Queue (Seconds)

This metric provides the average age (in seconds) of the messages in the buffered queue for all nonsystem queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Average age of messages in %schema%.%queue_name% queue is %value% seconds.

First Message Age in Buffered Queue Per Queue (Seconds)

This metric gives the age (in seconds) of the first message in the buffered queue for all non-system queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Age of first message in %schema%.%queue_name% buffered queue is %value% seconds.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Schema Name and Queue Name objects.

If warning or critical threshold values are currently set for any unique combination of Schema Name and Queue Name objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Schema Name and Queue Name objects, use the Edit Thresholds page.

Data Source

This metric is calculated by finding the age of the first message in all the subscribers of the queue and then the oldest amongst all is taken.

The following views and tables are used for the calculation:

1. <SCHEMA>.AQ$<QUEUE_TABLE>
2. v$buffered_queues

User Action

When using buffered queues for storing and propagating messages, monitor this metric to get the age of first message in the queue.

Messages processed per buffered queue (%)

This metric gives the messages processed percentage per minute per buffered queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Messages processed for queue %schema%.%queue_name% is %value% percent.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Schema Name and Queue Name objects.

If warning or critical threshold values are currently set for any unique combination of Schema Name and Queue Name objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Schema Name and Queue Name objects, use the Edit Thresholds page.

Data Source

This is calculated as the percent of total number of messages processed per minute and total number of messages received per minute in the last collection interval per buffered queue.

User Action

When using queues for storing/propagating messages, monitor this metric to get the messages processed percent (or throughput) per minute in the last collection interval for the queue.

Messages Processed Per Buffered Queue (%) Per Minute

This metric gives the messages processed percentage per minute in the last interval per buffered queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Messages processed per minute in the last interval for queue %schema%.%queue_name% is %value% .

Spilled Messages

This metric displays the current number of overflow messages spilled to disk from the buffered queue.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Current number of overflow messages spilled to disk from the buffered queue %schema%.%queue_name% is %value%

Total Messages Processed per Buffered Queue per Minute

This metric gives the total number of messages processed per minute per buffered queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Total messages processed per minute in the last interval for queue %schema%.%queue_name% is %value% .

Total Messages Received per Buffered Queue per Minute

This metric gives the total number of messages received or enqueued into the buffered queue per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Total messages received per minute in the last interval for queue %schema%.%queue_name% is %value% .

Message Per Buffered Queue Per Subscriber

This metric category monitors the messages for buffered queues per subscriber in the database.

Average Age of Messages Per Buffered Queue Per Subscriber (Seconds)

This metric display's the average age of messages in the buffered queue per queue in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Average age of messages for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

First Message Age in Buffered Queue per Subscriber (Seconds)

This metric displays the age of the first message in the buffered queue per queue per subscriber in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Age of first message for subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

Messages Processed Per Buffered Queue Per Subscriber (%)

This metric gives the messages processed percentage for the buffered queue per subscriber. Messages processed percent is calculated as the percent of the total number messages processed or dequeued to the total number of messages received or enqueued.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Messages processed for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% percent.

Messages Processed Per Buffered Queue (%) Per Subscriber Per Minute

This metric gives the total number of messages processed per minute per buffered queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value%.

Total Messages Processed Per Buffered Queue Per Subscriber Per Minute

This metric gives the total number of messages processed per minute per buffered queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Total messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% .

Total Messages Received Per Buffered Queue Per Subscriber Per Minute

This metric gives the total number of messages received or enqueued into the queue per subscriber per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Total messages received per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% .

Messages Per Persistent Queue

The metrics in this category monitor the age and state of the first (top of the queue) message for each persistent queue in the database except for the system queues. Queues that are in the schema of SYS, SYSTEM, DBSNMP, and SYSMAN are defined as system level queues.

Age of the First Message in Persistent Queue Per Queue

This metric gives the age (in seconds) of the first message in the persistent queue for all non-system queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Age of first message in %schema%.%queue_name% queue is %value% seconds.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Schema Name and Queue Name objects.

If warning or critical threshold values are currently set for any unique combination of Schema Name and Queue Name objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Schema Name and Queue Name objects, use the Edit Thresholds page.

Data Source

This metric is calculated by finding the age of the first message in all the subscribers of the queue and then the oldest amongst all is taken.

The following views/tables are used for the calculation:

1. <SCHEMA>.AQ$_<QUEUE_TABLE>_S
2. <SCHEMA>.AQ$_<QUEUE_TABLE>_I
3. <SCHEMA>.AQ$<QUEUE_TABLE>

User Action

When using persistent queues for storing and propagating messages, monitor this metric to get the age of first message in the queue.

Average Age of Messages Per Persistent Queue (Seconds)

This metric displays the average age of messages in the persistent queue per queue in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Average age of messages in %schema%.%queue_name% queue is %value% seconds.

Messages Processed Per Persistent Queue (%)

This metric gives the messages processed percentage for the persistent queue. Messages processed percent is calculated as the percent of the total number messages processed or dequeued to the total number of messages received or enqueued.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Messages processed for queue %schema%.%queue_name% is %value% percent.

Messages Processed Per Persistent Queue (%) Per Minute

This metric gives the messages processed percentage per minute per persistent queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Messages processed per minute in the last interval for queue %schema%.%queue_name% is %value%

Total Messages Processed per Persistent Queue per Minute

This metric gives the total number of messages processed per minute per persistent queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Total messages processed per minute in the last interval for queue %schema%.%queue_name% is %value% .

Total Messages Received per Persistent Queue per Minute

This metric gives the total number of messages received or enqueued into the queue per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Total messages received per minute in the last interval for queue %schema%.%queue_name% is %value% .

Messages Per Persistent Queue Per Subscriber

The metrics in this category monitor the age and state of the first (top of the queue) message for each persistent queue per queue subscriber in the database except for the system queues. Queues that are in the schema of SYS, SYSTEM, DBSNMP, and SYSMAN are defined as system level queues.

Average Age of Messages Per Persistent Queue Per Subscriber (Seconds)

This metric display's the average age of messages in the persistent queue per queue in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Average age of messages for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

Age of the First Message in Persistent Queue Per Subscriber

This metric gives the age (in seconds) of the first message in the persistent queue per subscriber for all non-system queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Age of first message for subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

Messages Processed Per Persistent Queue Per Subscriber (%)

This metric gives the messages processed percentage for the persistent queue per subscriber. Messages processed percent is calculated as the percent of the total number messages processed or dequeued to the total number of messages received or enqueued.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Messages processed for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% percent.

Messages Processed Per Persistent Queue (%) Per Subscriber Per Minute

This metric gives the messages processed percentage per minute per persistent queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value%.

Total Messages Processed Per Persistent Queue Per Subscriber Per Minute

This metric gives the messages processed percentage per minute per persistent queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Total messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% .

Total Messages Received Per Persistent Queue Per Subscriber Per Minute

This metric gives the total number of messages received or enqueued into the queue per subscriber per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Total messages received per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% .

Memory Usage

The metric in this category provides information about the total memory used by the database instance.

Total Memory Usage (MB)

This metric displays the total amount of memory used in MB.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	Not Defined	Not Defined	Total memory usage is %value% MB.

Monitoring User Account

The metrics in this category provide visibility into potential problems with the Monitoring User account (for example DBSNMP) to prevent a lapse in monitoring.

Monitoring User Connectivity Issue

This metric monitors the expiry of the Monitoring User account password, and raises an alert when the password is not updated in Oracle Enterprise Manager's target configuration.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	ORA-	ORA-	Connection for monitoring user %USER_NAME% failed with error %PASSWORD_INVALID%.

Monitoring User Expiry

This metric monitors the potential expiry of the Monitoring User account.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 24 Hours	72	Not Defined	Monitoring user %USER_NAME% will expire in %ACCOUNT_EXPIRY_IN_HOURS% hours.

Database Monitoring User Privileges Check

This metric checks whether the Monitoring User account has monitoring privileges on par with the DBSNMP or SYSDBA user accounts. This is useful when using a non-DBSNMP user account.

The collection is disabled by default.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 48 Hours	Not Defined	FALSE	Monitoring user %USER_NAME% does not have sufficient monitoring privileges under %ROLE% role. It must have monitoring privileges equal or higher than DBSNMP user.

OCM Instrumentation

The metrics in this category determine whether the database has been instrumented with Oracle Configuration Manager (OCM). Oracle Configuration Manager is used to personalize the support experience by collecting configuration information and uploading it to the Oracle repository.When customer configuration data is uploaded on a regular basis, customer support representatives can analyze this data and provide better service to the customers. For example, when a customer logs a service request, he can associate the configuration data directly with that service request. The customer support representative can then view the list of systems associated with the customer and solve problems accordingly.

Instrumentation Present

This metric determines whether the database has been instrumented with Oracle Configuration Manager.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

This metric tests for the existence of the MGMT_DB_LL_METRICS package body owned by the ORACLE_OCM user.

User Action

No user action is required.

Need to Instrument with OCM

This metric determines that Oracle Configuration Manager needs to be instrumented in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	1	Not Defined	OCM Instrumentation should be installed in database. Please use $ORACLE_HOME/ccr/admin/scripts/installCCRSQL script with collectconfig parameter.

Data Source

This metric tests for the existence of the emCCR executable in the $ORACLE_HOME/ccr/bin/ directory. If the emCCR executable is present, then Enterprise Manager checks to see if the MGMT_DB_LL_METRICS package body, owned by the ORACLE_OCM user, exists in the Management Repository.

If the emCCR executable is present but the MGMT_DB_LL_METRICS package body is missing, then this metric returns 1, indicating that the database must be instrumented.

User Action

Install Oracle Configuration Manager (OCM) in the database.

OCM Configured

This metric determines how the Oracle Configuration Manager is configured.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

This metric tests for the existence of the emCCR executable in the $ORACLE_HOME/ccr/bin directory.

User Action

No user action is required.

Operational Error

This metric category contains the metrics representing errors that might affect the operation of the database, such as archiver error, or media failure as recorded in the database alert log file. These errors are not triggered by ADR incidents but are daily issues which you can handle without interaction with Oracle Support. The alert log file has a chronological log of messages and errors.

Each metric signifies that the database being monitored has detected a critical error condition that might affect the normal operation of the database and has generated an error message to the alert log file since the last sample time. The Support Workbench in Enterprise Manager might contain more information about the error.

> **Note:**
>
> For more information about Incident metrics and Operational Error metrics, sign in to My Oracle Support and search for the following Oracle Support note:
>
> *Database Alert log monitoring in 12c explained (Doc ID 1538482.1)*
>
> https://support.oracle.com/

Setting Thresholds for Operational Errors

To edit the thresholds for any of the following metrics, from the Enterprise Manager UI, right-click the target name, select Monitoring, then Metric and Collection Settings. The following settings provide examples of some of the possible settings:

- Warning Threshold: Not Defined; Critical Threshold: .*

  In this case, the Management Agent generates a critical error alert in Enterprise Manager when the error occurs.

- Warning Threshold: .*; Critical Threshold: Not Defined

  In this case, the Management Agent generates a warning alert in Enterprise Manager when the error occurs.

- Warning Threshold: Not Defined; Critical Threshold: Not Defined

  In this case, the Management Agent does not generate an alert in Enterprise Manager when the error occurs.

Alert Log Error Trace File

This metric reports the name of the trace file (if any) associated with the logged error.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

No user action is required.

Alert Log Name

This metric reports the name of the alert log file.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

No user action is required.

Archiver Error

This metric signifies that an archiver error has occurred on the database being monitored, since the last sample time.

If the database is running in ARCHIVELOG mode, an alert is displayed when alert log (log.xml) contains the entry of type='ERROR',group='Archiver Error'. It detects the following information from log.xml:

```
lertlogAdr.pl: Tue Oct 24 16:09:29 2017: DEBUG:

20 error types defined.
Error type pattern:
 incident,type=["']INCIDENT_ERROR["'],group=["']Generic Internal Error["']
 incident,type=["']INCIDENT_ERROR["'],group=["']Session Terminated["']
 incident,type=["']INCIDENT_ERROR["'],group=["']Internal SQL Error["']
 incident,type=["']INCIDENT_ERROR["'],group=["']Access Violation["']
 incident,type=["']INCIDENT_ERROR["'],group=["']Redo Log Corruption["']
 incident,type=["']INCIDENT_ERROR["'],group=["']File Access Error["']
 incident,type=["']INCIDENT_ERROR["'],group=["']Inconsistent DB State["']
 incident,type=["']INCIDENT_ERROR["'],group=["']Data Block Corruption["']
 incident,type=["']INCIDENT_ERROR["'],group=["']Deadlock["']
 incident,type=["']INCIDENT_ERROR["'],group=["']Out of Memory["']
 incident,type=["']INCIDENT_ERROR["'],group=["']Cluster Error["']
 incident,level=["'][12]["'],type=["']INCIDENT_ERROR["']
 dataFailure,type=["']ERROR["'],group=["']DRA["']
 operational,type=["']ERROR["'],group=["']Archiver Error["']
 operational,type=["']ERROR["'],group=["']Data Block Corruption["']
 operational,type=["']ERROR["'],group=["']Media Failure["']
 operational,level=["'][12]["'],type=["']ERROR["']
 operational,level=["']*["'],type=["']ERROR["']
 operational,level=["']*["'],type=["']WARNING["']
 operational,level=["']*["'],type=["']*["']
```

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	Archiver error detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of error as Critical. For information about modifying threshold values, see Setting Thresholds for Incident Metrics.

Data Source

The source of the data is `$AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_n.n.n.n/scripts/alertlogAdr.pl`.

In the preceding directory path, `$AGENT_BASE` refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the error. However, the most likely cause of this message is that the destination device is out of space to store the

redo log file. Verify the device specified in the initialization parameter *ARCHIVE_LOG_DEST* is set up properly for archiving.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Data Block Corruption

This metric signifies that the database being monitored has generated a corrupted block error (ORA-01157 or ORA-27048) to the alert file since the last sample time. The alert file is a special trace file containing a chronological log of messages and errors. An alert event is triggered when data block corrupted messages are written to the alert file.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	A datablock corruption detected in %alertLogName% at time/ line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of error as Critical. For information about modifying threshold values, see "Setting Thresholds for Incident Metrics".

Data Source

The source of the data is $AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_*n.n.n.n*/scripts/alertlogAdr.pl.

In the preceding directory path, $AGENT_BASE refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the error.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Generic Operational Error

This metric signifies that the database being monitored has generated some error that may affect the normal operation of the database to the alert file since the last sample time. The alert

file is a special trace file containing a chronological log of messages and errors. An alert event is triggered when data block corrupted messages are written to the alert file.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	Operational error (%errorCodes%) detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of error as Critical. For information about modifying threshold values, see Setting Thresholds for Operational Error Metrics in *Oracle Grid Infrastructure Metric Reference Manual*.

Data Source

The source of the data is `$AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_n.n.n.n/scripts/alertlogAdr.pl`.

In the preceding directory path, `$AGENT_BASE` refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the error.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

Media Failure

This metric signifies that the database being monitored has generated a media failure error (ORA-01242 or ORA-01243) to the alert file since the last sample time. The alert file is a special trace file containing a chronological log of messages and errors. An alert event is triggered when data block corrupted messages are written to the alert file.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	.*	Media Failure detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

By default, Enterprise Manager reports this type of error as Critical. For information about modifying threshold values, see Setting Thresholds for Operational Error Metrics in *Oracle Grid Infrastructure Metric Reference Manual*.

Data Source

The source of the data is `$AGENT_BASE/plugins/oracle.sysman.db.agent.plugin_n.n.n.n/scripts/alertlogAdr.pl`.

In the preceding directory path, `$AGENT_BASE` refers to the home of the Oracle Management Agent and *n.n.n.n* refers to the release version of the Oracle Database plug-in, such as plug-in release 13.1.0.0.

User Action

Use Support Workbench in Enterprise Manager to examine the details of the error.

> **Note:**
>
> This event does not clear automatically because there is no automatic way of determining when the problem has been resolved. Therefore, you must clear the event manually after the problem is fixed.

User-Defined Error

This metric displays the user-defined error.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	Error (%errorCodes%) detected in %alertLogName% at time/line number: %timeLine%.

User-Defined Text

This metric displays user-defined text. You can use this metric to raise alerts for custom text found in the XML alert log.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	Matching text (%errorCodes%) detected in %alertLogName% at time/line number: %timeLine%.

Multiple Thresholds

Use the thresholds to define the custom text (regular expression). An alert will be raised for any entry in the XML alert log with text matching the custom text entered here.

User-Defined Warning

This metric displays the user-defined warning.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	Warning (%errorCodes%) detected in %alertLogName% at time/line number: %timeLine%.

Operating System Audit Records

This metrics in this category check target database OS audit trail files. It checks for aud, bin, and .xml file extensions in either a user-configured location or the default location.

Size of Audit Files (MB)

This metric displays the cumulative size of audit files. Due to different reasons, if audit trails can't be written to the database, then they are written to the file system, and with time, the files grow. If the cumulative size of audit files exceeds more than 1 GB, it is marked as a warning alert. You must configure the critical threshold if you want to define a critical alert.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 6 Hours	1024	Not Defined	%FILE_SIZE% MB of Audit Trail files collected (.aud: %AUD_FILE_SIZE% MB, .xml: %XML_FILE_SIZE% MB, .bin: %BIN_FILE_SIZE% MB)

Data Source

OS audit files of the target database

User Action

Load the audit files back to the database and find out why the audit trails are written to the OS file system.

Recovery Area

This metric category contains the Recovery Area metrics that enable you to monitor Fast Recovery Area usage. These metrics represent the respective space consumption as a percentage, and are database-level metrics that are evaluated by the database server every 15 minutes or during file creation, whichever occurs first. The metric data is also printed in the alert log. For cluster databases, these metrics are monitored at the cluster database target level and not by member instances.

Recovery Area Free Space (%)

This metric represents the recovery area free space as a percentage. The Critical Threshold is set for < 3% and the Warning Threshold is set for < 15%. You cannot customize these thresholds. An alert is returned the first time the alert occurs, and the alert is not cleared until the available space rises above 15%.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes or during file creation, whichever occurs first	15% (cannot be changed)	3% (cannot be changed)	db_recovery_file_dest_size of N bytes is N% used, and has N remaining bytes available

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Recovery Area Used Space (%)

This metric represents the recovery area used space as a percentage. The Critical threshold is set for < 97% and the Warning threshold is set for < 85% and these can be customized. Any changes will directly be applied on the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR2, 11gR202, 12c, 12cR102, 12cR2	Every 15 minutes or during file creation, whichever occurs first	None	None	–
18c and later	Every 15 minutes or during file creation, whichever occurs first	85%	97%	The value of Recovery Area Used Space (%) for RECOVERY AREA is XX.YYY.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Response

The metrics in this category represent the responsiveness of the Oracle Server, with respect to a client.

State

This metric represents the state of the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 15 Seconds	MOUNTED	DOWN|UNKNOWN.*	The database status is %value%.

Data Source

Not available.

User Action

The required actions are specific to your site. The required actions are specific to your site.

Status

This metric checks whether a new connection can be established to a database. If the maximum number of users is exceeded or the listener is down, this test is triggered.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 15 Seconds	Not Defined	0	Failed to connect to database instance %oraerr%.

Data Source

Perl returns 1 when a connection can be made to the database (using Management Agent monitoring connection details), 0 otherwise.

User Action

Check the status of the listener to make sure it is running on the node where the event was triggered. If the listener is running, check to see if the number of users is at the session limit.

> **Note:**
>
> The choice of user credentials for the Probe metric should be considered. If the preferred user has the RESTRICED SESSION privilege, the user will be able to connect to a database even if the LICENSE_MAX_SESSIONS limit is reached.

SCN Growth Statistics

This metric category provides information about the Systems Change Number (SCN) in the database environment and reports on the health of the SCN growth in the database.

Current SCN

This metric displays the value of the current SCN.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Hour	Not Defined	Not Defined	The current SCN is %current_scn%.

Current SCN Compatibility

This metric displays the current SCN compatibility for the database.

> **Note:**
>
> This is for internal customers only.

Target Version	Collection Frequency
All versions	Every 60 Minutes

Max Rate

This metric displays the rate at which the SCN growth is calculated, such as 16k/32k.

> **Note:**
>
> This metric is used by internal users.

Target Version	Collection Frequency
All versions	Every 60 Minutes

Maximum SCN Compatibility

This metric displays the maximum SCN compatibility for the database.

> **Note:**
>
> This is for internal customers only.

Target Version	Collection Frequency
All versions	Every 60 Minutes

SCN Health

This metric displays the SCN health of the database, that is, headroom or the number of days before the database runs out of SCN at the current SCN consumption rate.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Hour	62	10	The SCN health is %scn_health%.

SCN Total Growth Rate (per sec)

This metric displays the total SCN growth rate over the previous 24 hours.

SCNs occur in a monotonically increasing sequence (that is, each SCN is greater than or equal to the one before it), and there is a very large upper limit to how many SCNs Oracle Database can use. Because there is an upper limit, it is important that Oracle Database does not run out of available SCNs and therefore it is important to monitor the SCN growth rate.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Hour	Not Defined	Not Defined	The total SCN Growth rate per second (last 24 hours) is %scn_total_growth%.

SCN Instance Statistics

This metric category provides information about the SCN growth rate due to intrinsic activity.

SCN Intrinsic Growth Rate (per sec)

This metric displays the rate at which the SCN of the database increases only due to database transactions, and not due to database links. It is averaged per second over the last 24 hours. The rate is displayed in SCNs per second.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Hour	Not Defined	Not Defined	The intrinsic SCN Growth rate per second is %scn_intrinsic_growth_rate%.

SCN Max Statistics

This metric category provides information about the maximum value of the SCN.

Max SCN Jump in one second (last 24 hours)

This metric displays the maximum SCN jump in one second over the previous 24 hours.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Hour	Not Defined	Not Defined	The maximum SCN jump in one second (last 24 hours) is %scn_max_jump%.

Segment Advisor Recommendations

The metrics in this category provide segment advisor recommendations. Oracle uses the Automatic Segment Advisor job to detect segment issues regularly within maintenance windows. It determines whether the segments have unused space that can be released. The Number of recommendations is the number of segments that have Reclaimable Space. The

recommendations come from all runs of the automatic segment advisor job and any user scheduled segment advisor jobs.

Number of Recommendations

Oracle uses the Automatic Segment Advisor job to detect segment issues regularly within maintenance windows. It determines whether the segments have unused space that can be released. The Number of recommendations is the number of segments that have Reclaimable Space. The recommendations come from all runs of the automatic segment advisor job and any user scheduled segment advisor jobs.

Target Version	Collection Frequency
All versions	Every 60 Minutes

Data Source

Not available.

User Action

Oracle recommends shrinking or reorganizing these segments to release unused space.

Session Suspended

The metrics in this category represent the number of resumable sessions that are suspended due to some correctable error.

Session Suspended by Data Object Limitation

This metric represents the session suspended by data object limitation.

This metric is collected for all versions.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Session Suspended by Quota Limitation

This metric represents the session suspended by quota limitation.

This metric is collected for all versions.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Session Suspended by Rollback Segment Limitation

This metric represents the session suspended by rollback segment limitation.

This metric is collected for all versions.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Session Suspended by Tablespace Limitation

This metric represents the session suspended by a tablespace limitation.

This metric is collected for all versions.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

SGA Pool Wastage

The metrics in this category represent the percentage of the various pools in the SGA that are being wasted.

Java Pool Free (%)

This metric represents the percentage of the Java Pool that is currently marked as free.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	Not Defined	Not Defined	%value%% of the Java pool is free.

Data Source

The data is derived from the formula ((Free/Total)*100) where:

- Free: select sum(decode(name,'free memory',bytes)) from v$sgastat where pool = 'java pool'
- Total: select sum(bytes) from v$sgastat where pool = 'java pool'

User Action

If this pool size is too small, the database JVM (Java Virtual Machine) may not have sufficient memory to satisfy future calls, leading potentially to unexpected database request failures.

Large Pool Free (%)

This metric represents the percentage of the Large Pool that is currently marked as free.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	Not Defined	Not Defined	%value%% of the Java pool is free.

Data Source

The data is derived from the formula ((Free/Total)*100) where:

- Free: select sum(decode(name,'free memory',bytes)) from v$sgastat where pool = 'large pool'
- Total: select sum(bytes) from v$sgastat where pool = 'large pool'

User Action

Consider enlarging the large pool or utilizing it more sparingly. This reduces the possibility of large memory areas competing with the library cache and dictionary cache for available memory in the shared pool.

Shared Pool Free (%)

This metric represents the percentage of the Shared Pool that is currently marked as free.

This test checks the percentage of Shared Pool that is currently free. If the value is less than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	Not Defined	Not Defined	Generated By Database Server

Data Source

The data is derived from the formula ((Free/Total)*100) where:

- free: select sum(decode(name,'free memory',bytes)) from v$sgastat where pool = 'shared pool'
- total: select sum(bytes) from v$sgastat where pool = 'shared pool'

User Action

If the percentage of Free Memory in the Shared Pool rises above 50%, too much memory has been allocated to the shared pool. This extra memory could be better utilized by other

applications on the machine. In this case the size of the Shared Pool should be decreased. This can be accomplished by modifying the shared_pool_size initialization parameter.

Snapshot Too Old

The metrics in this category represent the snapshots that are too old due to rollback segment limit or tablespace limit.

Snapshot Too Old Due to Rollback Segment Limit

This metric represents the snapshot too old because of the rollback segment limit.

This metric is collected for all versions.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Snapshot Too Old Due to Tablespace Limit

This metric represents the snapshot too old because of the tablespace limit.

This metric is collected for all versions.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Space Usage by Buffered Queues

The metrics in this category monitor the space usage of buffered queues with respect to the streams pool size.

Queue Size (MB)

This metric display's the size of buffered queue, which is the total number of megabytes allocated for all messages and metadata.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Size of buffered queue %schema%.%queue_name% is %value% MB.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Schema Name and Queue Name objects.

If warning or critical threshold values are currently set for any unique combination of Schema Name and Queue Name objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Schema Name and Queue Name objects, use the Edit Thresholds page.

Data Source

The data is derived from the INSTANCE_NAME column in the GV$INSTANCE view.

User Action

When using queues for storing or propagating messages, monitor this metric to get the instance in which the buffered queue is available.

Space Usage of Buffered Queue With Respect to Streams Pool Size (%)

This metric gives the space usage percentage of buffered queue with respect to streams pool size per buffered queue.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR202 and later	Every 30 Minutes	Not Defined	Not Defined	Buffered queue %schema%.%queue_name% has consumed %value% percent of streams pool size.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Schema Name and Queue Name objects.

If warning or critical threshold values are currently set for any unique combination of Schema Name and Queue Name objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Schema Name and Queue Name objects, use the Edit Thresholds page.

Data Source

The data is derived from the QUEUE_SIZE AND CURRENT_SIZE columns from GV$BUFFERED_QUEUES and GV$SGA_DYNAMIC_COMPONENTS views.

User Action

When using buffered queues for storing or propagating messages, monitor this metric to get the space usage percentage of buffered queue with respect to the allocated streams pool size.

SQL Response Time

The metrics in this category approximate the responsiveness of SQL. The SQL Response Time metrics are related to user workloads. For multitenant databases, they should only be enabled at the PDB level. They should be disabled at the CDB level to avoid false alerting.

Baseline SQL Response Time

This metric contains the response time of the baseline.

Target Version	Collection Frequency
All Versions	Every 5 Minutes

Data Source

Not available.

User Action

No user action is required.

Current SQL Response Time

This metric contains the response time of the latest collection.

Target Version	Collection Frequency
All Versions	Every 5 Minutes

Data Source

Not available.

User Action

No user action is required.

SQL Response Time (%)

SQL Response Time is the average elapsed time per execution of a representative set of SQL statements, relative to a baseline. It is expressed as a percentage.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every10 Minutes	Not Defined	Not Defined	SQL response time is %value%% of baseline.

Data Source

The data is derived from the PL/SQL packaged procedure `mgmt_response.get_metric_curs`.

User Action

If the SQL Response Time is less than 100%, then SQL statements are taking less time to execute when compared to the baseline. Response Time greater than 100% indicates that the database is not performing well when compared to the baseline.

SQL Response Time is a percentage of the baseline, not a simple percentage. So, for example, 100% of baseline means the SQL Response Time is the same as the baseline. 200%

of baseline means the SQL Response Time is two times slower than the baseline. 50% of baseline means SQL Response Time is two times faster than baseline. A warning threshold of 200% indicates that the database is two times slower than the baseline, while a critical threshold of 500% indicates the database is 5 times slower than the baseline.

Representative statements are selected when two V$SQL snapshots are taken. All calculations are based on the deltas between these two snapshots. First, the median elapsed_time/execution for all statements that were executed in the time interval between the two snapshots are calculated. Then all statements that have an elapsed_time/execution > median elapsed_time/execution are taken, and the top 25 most frequently executed statements are displayed.

Pre-requisites for Monitoring SQL Response Time

Some tables and a PL/SQL package must be installed on the monitored database. This can be done by going to the database targets page and pressing the Configure button for your database. If a database has not been configured, the message Not Configured will be displayed for SQL Response Time.

Configuring the Baseline

The baseline is configured on demand, automatically. The first time the agent calls the stored procedure to get the value of the metric, a snapshot of V$SQL is taken. The second time, another snapshot is taken. Then the representative statements are picked and stored in a table. The next time the agent requests the value of the metric, the relative SQL response time is calculated and returned.

Because of baseline configuration, there will be a delay between the time the database is configured and the value of the metric is displayed. During this period, the message of the collection status will be displayed for SQL Response Time.

Enterprise Manager will automatically configure the baseline against which SQL Response Time will be compared. However, in order for the SQL Response Time metric to be truly representative, the DBA must reconfigure the baseline at a time when the load on the database is typical.

To reconfigure the baseline, click on the link titled Edit Reference Collection located next to the SQL Response Time value on the Database Home Page. The SQL statements used for tracking the SQL Response Time and baseline values are displayed. Click **Reset Reference Collection**. This clears the list of statements and the baseline values. Enterprise Manager will then automatically reconfigure the baseline within minutes.

If the database was lightly loaded at the time the baseline was taken, then the metric can indicate that the database is performing poorly under typical load when such is not the case. In this case, the DBA must reset the baseline. If the DBA has never manually reset the baseline, then the metric value will not be representative.

Streams Apply Aborted

The metrics in this category check for the Streams Apply processes.

> **Note:**
>
> This is a server-generated alert.

Streams Apply Process Aborted

This metric detects when a Streams Apply process configured on this database aborts. This metric indicates a critical error.

Data Source

The DBA_APPLY view STATUS column indicates ABORTED if the apply process has aborted.

User Action

Obtain the exact error message in dba_apply, take the appropriate action for this error, then restart the apply process using dbms_apply_adm.start_apply.

Using the DBA_APPLY_ERROR view, identify the specific change record which encountered an error(MESSAGE_NUMBER) within a failed transaction and the complete error message (ERROR_MESSAGE). Detailed information about the transaction can be found using Enterprise Manager or by using the scripts described in the documentation Displaying Detailed Information about Apply Errors.

If DBA_APPLY error message is ORA-26714, then consider setting the 'DISABLE_ON_ERROR' apply parameter to 'N' to avoid aborting on future user errors.

Streams Apply Process Error

This metric indicates that the apply process encountered an error when it was applying a transaction.

Data Source

Not available.

User Action

Look at the contents of the error queue as well as dba_apply_error to determine the cause of the error. After the errors are resolved, reexecute them using dbms_apply_adm.execute_error or dbms_apply_adm.execute_all_errors.

Streams Apply Coordinator Statistics

The metrics in this category show statistics about the transactions processed by the coordinator process of each apply process. The **Total Number of Transactions Received** field shows the total number of transactions received by a coordinator process. The **Number of Transactions Assigned** field shows the total number of transactions assigned by a coordinator process to apply servers. The **Total Number of Transactions Applied** field shows the total number of transactions successfully applied by the apply process.

The values for an apply process are reset to zero if the apply process is restarted.

Total Number of Transactions Assigned

This metric shows statistics about the total number of transactions assigned by the coordinator process to apply servers since the apply process last started.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data is derived from the TOTAL_ASSIGNED column in the following query shows this metric for an apply process:

```
SELECT APPLY_NAME, TOTAL_RECEIVED, TOTAL_ASSIGNED, TOTAL_APPLIED
  FROM V$STREAMS_APPLY_COORDINATOR;
```

User Action

When an apply process is enabled, monitor this metric to ensure that the apply process assigning transactions to apply servers.

Rate of Transactions Applied (per Sec)

This metric reports the rate (per second) at which transactions are applied by the apply process.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data is derived from the target database, gv%streams_apply_coordinator table.

User Action

No user action is required.

Rate of Transactions Assigned (per Sec)

This metric reports the rate (per second) at which transactions are assigned to the apply servers.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data is derived from the target database, gv%streams_apply_coordinator table.

User Action

No user action is required.

Rate of Transactions Received (per Sec)

This metric reports the rate (per second) at which apply coordinator is receiving the transactions.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data is derived from the target database, gv%streams_apply_coordinator table.

User Action

No user action is required.

Total Number of Transactions Applied

This metric shows statistics about the total number of transactions applied by the apply process since the apply process last started.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The TOTAL_APPLIED column in the following query shows this metric for an apply process:

```
SELECT APPLY_NAME, TOTAL_RECEIVED, TOTAL_ASSIGNED, TOTAL_APPLIED
  FROM V$STREAMS_APPLY_COORDINATOR;
```

User Action

When an apply process is enabled, monitor this metric to ensure that the apply process is applying transactions.

Total Number of Transactions Received

This metric shows statistics about the total number of transactions received by the coordinator process since the apply process last started.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The TOTAL_RECEIVED column in the following query shows this metric for an apply process:

```
SELECT APPLY_NAME, TOTAL_RECEIVED, TOTAL_ASSIGNED, TOTAL_APPLIED
  FROM V$STREAMS_APPLY_COORDINATOR;
```

User Action

When an apply process is enabled, monitor this metric to ensure that the apply process is receiving transactions.

Streams Apply Errors

The metrics in this category collect information about Apply Errors and Error transactions.

Error Message

This metric reports the error message of the error raised by the transaction.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data source for this metric is target database, dba_apply_error table.

User Action

No user action is required.

Error Number

This metric reports the error code of the error raised by the transaction.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data source for this metric is target database, dba_apply_error table.

User Action

No user action is required.

Local Transaction ID

This metric reports the local transaction ID for the error transaction.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

Data source for this metric is the target database, dba_apply_error table.

User Action

No user action is required.

Message Count

This metric reports the total number of events inside the error transaction.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data source for this metric is the target database, dba_apply_error table.

User Action

No user action is required.

Source Transaction ID

This metric reports the original transaction ID at the source database, for the error transaction.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data source for this metric is the target database, dba_apply_error table.

User Action

No user action is required.

Streams Apply Queue - Buffered

The metrics in this category show the current total number of messages in a buffered queue to be dequeued by each apply process and the total number of messages to be dequeued by each apply process that have spilled from memory into the persistent queue table.

Streams Apply - (%) Spilled Messages

This metric usually indicates that transactions are staying longer in memory.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	Spilled messages for Apply process [%APPLY_NAME%] queue is %value% percent.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Apply Name object.

If warning or critical threshold values are currently set for any Apply Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Apply Name object, use the Edit Thresholds page.

Data Source

The data source for this metric is the target database, gv$buffered_queues, gv$buffered_subscribers tables.

User Action

Either increase Streams Pool size and /or increase Apply Parallelism to speed up Apply processing.

Streams Apply Queue - Persistent

The metrics in this category show the number of messages in a persistent queue in READY state and WAITING state for each apply process.

Streams Apply - (%) Messages in Waiting State

This metric shows the percentage of messages in a wait state.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	Messages waiting for Apply process [%APPLY_NAME%] queue is %value% percent.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Apply Name and Messages Delivery Mode objects.

If warning or critical threshold values are currently set for any unique combination of Apply Name and Messages Delivery Mode objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Apply Name and Messages Delivery Mode objects, use the Edit Thresholds page.

Data Source

The data source for this metric is Target Database and Apply Queue.

User Action

No user action is required.

Streams Apply Reader Statistics

The reader server for an apply process dequeues messages from the queue. The reader server computes dependencies between LCRs and assembles messages into transactions. The reader server then returns the assembled transactions to the coordinator, which assigns them to idle apply servers.

The metrics in this category shows the total number of messages dequeued by the reader server for the apply process since the last time the apply process was started.

Rate at Which Messages Are Getting Spilled (per Sec)

The reader server for an apply process dequeues messages from the queue. The reader server computes dependencies between LCRs and assembles messages into transactions. The reader server then returns the assembled transactions to the coordinator, which assigns them to idle apply servers.

This metric shows the rate at which message are getting spilled (per second) by the reader server for the apply process since the last time the apply process was started.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	Total number of spilled messages for Apply Process [%APPLY_NAME%] is %value% .

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Apply Name object.

If warning or critical threshold values are currently set for any Apply Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Apply Name object, use the Edit Thresholds page.

Data Source

For this metric, the data source is Target database, gv$streams_apply_reader view.

User Action

No user action is required.

Total Number of Messages Dequeued

The reader server for an apply process dequeues messages from the queue. The reader server computes dependencies between LCRs and assembles messages into transactions. The reader server then returns the assembled transactions to the coordinator, which assigns them to idle apply servers.

This metric shows the total number of messages dequeued by the reader server for the apply process since the last time the apply process was started.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The TOTAL_MESSAGES_DEQUEUED column in the following query shows this metric for an apply process:

```
SELECT APPLY_NAME, TOTAL_MESSAGES_DEQUEUED FROM V$STREAMS_APPLY_READER;
```

User Action

When an apply process is enabled, monitor this metric to ensure that the apply process is dequeuing messages.

Total Number of Spilled Messages

The reader server for an apply process dequeues messages from the queue. The reader server computes dependencies between LCRs and assembles messages into transactions. The reader server then returns the assembled transactions to the coordinator, which assigns them to idle apply servers.

This metric shows the total number of messages spilled by the reader server for the apply process since the last time the apply process was started.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

For this metric, the data source is Target database, gv$streams_apply_reader view.

User Action

No user action is required.

Streams Capture Message Statistics

The metrics in this category show the number of messages captured and the number of messages enqueued by each capture process since the capture process last started.

The **Total Messages Captured** field shows the total number of redo entries passed by LogMiner to the capture process for detailed rule evaluation. A capture process converts a redo entry into a message and performs detailed rule evaluation on the message when capture process prefiltering cannot discard the redo entry. After detailed rule evaluation, the message is enqueued if it satisfies the capture process rule sets, or the message is discarded if it does not satisfy the capture process rule sets. The **Total Messages Enqueued** field shows the total number of messages enqueued. The number of captured messages captured can be higher than the number of enqueued messages.

The total messages enqueued includes enqueued logical change records (LCRs) that encapsulate data manipulation language (DML) and data definition language (DDL) changes. The total messages enqueued also includes messages that contain transaction control statements. These messages contain directives such as COMMIT and ROLLBACK. Therefore, the total messages enqueued is higher than the number of row changes and DDL changes enqueued by a capture process.

Message Capture Rate (per Sec)

This metric shows the number of messages captured by each capture process since the capture process last started.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

For this metric, the data source is Target database, gv$streams_capture view.

User Action

No user action is required.

Messages Enqueue Rate (per Sec)

This metric shows the number of messages enqueued by each capture process since the capture process last started.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

Not available.

User Action

The required actions are specific to your site.

Total Messages Captured

This metric shows information about the number of redo entries passed by LogMiner to the capture process for detailed rule evaluation. A capture process converts a redo entry into a message and performs detailed rule evaluation on the message when capture process prefiltering cannot discard the change.

After detailed rule evaluation, the message is enqueued if it satisfies the capture process rule sets, or the message is discarded if it does not satisfy the capture process rule sets.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The TOTAL_MESSAGES_CAPTURED column in the following query shows this metric for a capture process:

```
SELECT CAPTURE_NAME, TOTAL_MESSAGES_CAPTURED, TOTAL_MESSAGES_ENQUEUED   FROM
V$STREAMS_CAPTURE;
```

User Action

When a capture process is enabled, monitor this metric to ensure that the capture process is scanning redo entries.

Total Messages Enqueued

This metric shows information about the number of messages enqueued by a capture process. The number of messages enqueued includes logical change records (LCRs) that encapsulate data manipulation language (DML) and data definition language (DDL) changes. The number of messages enqueued also includes messages that contain transaction control statements. These messages contain directives such as COMMIT and ROLLBACK. Therefore, the number of messages enqueued is higher than the number of row changes and DDL changes enqueued by a capture process.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The TOTAL_MESSAGES_ENQUEUED column in the following query shows this metric for a capture process:

```
SELECT CAPTURE_NAME, TOTAL_MESSAGES_CAPTURED, TOTAL_MESSAGES_ENQUEUED   FROM
V$STREAMS_CAPTURE;
```

User Action

When a capture process is enabled, monitor this metric to ensure that the capture process is enqueuing messages. If you know that there were source database changes that should be captured by the capture process, and the capture process is not capturing these changes, then there might be a problem with the rules used by the capture process.

Streams Capture Queue Statistics

The metrics in this category show the current total number of messages in a buffered queue that were enqueued by each capture process and the total number of messages enqueued by each capture process that have spilled from memory into the queue spill table.

If queue publishers other than the capture process enqueue messages into a buffered queue, then the values shown can include messages from these other queue publishers.

Capture Queue - Cumulative Number of Messages

This metric shows information about the cumulative number of messages enqueued by a capture process in a buffered queue.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

Not available.

User Action

The required actions are specific to your site.

Capture Queue - Cumulative Number of Spilled Messages

This metric shows information about the cumulative number of spilled messages enqueued by a capture process in a buffered queue.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

Not available.

User Action

The required actions are specific to your site.

Capture Queue - Number of Messages

This metric shows information about the number of messages enqueued by a capture process in a buffered queue. This number includes both messages in memory and messages spilled from memory.

If queue publishers other than the capture process enqueue messages into a buffered queue, then the values shown can include messages from these other queue publishers.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The NUM_MSGS column in the following query shows this metric for a capture process:

```
SELECT CAPTURE_NAME, P.NUM_MSGS NUM_MSGS, Q.SPILL_MSGS SPILL_MSGS
  FROM V$BUFFERED_PUBLISHERS P, V$BUFFERED_QUEUES Q, DBA_CAPTURE C
  WHERE C.QUEUE_NAME = P.QUEUE_NAME
    AND C.QUEUE_OWNER = P.QUEUE_SCHEMA
    AND C.QUEUE_NAME = Q.QUEUE_NAME
    AND C.QUEUE_OWNER = Q.QUEUE_SCHEMA
    AND C.CAPTURE_NAME = P.SENDER_NAME
    AND P.SENDER_ADDRESS IS NULL
    AND P.SENDER_PROTOCOL = 1;
```

User Action

When a capture process is enabled, monitor this metric to ensure that the capture process enqueuing messages.

Capture Queue - Number of Spilled Messages

This metric shows information about the number of messages enqueued by a capture process that have spilled from memory to the queue spill table. Messages in a buffered queue can spill from memory into the queue spill table if they have been staged in the buffered queue for a period of time without being dequeued, or if there is not enough space in memory to hold all of the messages.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The SPILL_MSGS column in the following query shows this metric for a capture process:

```
SELECT CAPTURE_NAME, P.NUM_MSGS NUM_MSGS, Q.SPILL_MSGS SPILL_MSGS
  FROM V$BUFFERED_PUBLISHERS P, V$BUFFERED_QUEUES Q, DBA_CAPTURE C
  WHERE C.QUEUE_NAME = P.QUEUE_NAME
    AND C.QUEUE_OWNER = P.QUEUE_SCHEMA
    AND C.QUEUE_NAME = Q.QUEUE_NAME
```

```
AND C.QUEUE_OWNER = Q.QUEUE_SCHEMA
AND C.CAPTURE_NAME = P.SENDER_NAME
AND P.SENDER_ADDRESS IS NULL
AND P.SENDER_PROTOCOL = 1;
```

User Action

The number of spilled messages should be kept as low as possible for the best performance. A high number of spilled messages can result in the following cases:

- There might be a problem with a propagation that propagates the messages captured by the capture process, or there might be a problem with an apply process that applies messages captured by the capture process. When this happens, the number of messages can build in a queue because they are not being consumed. In this case, make sure the relevant propagations and apply processes are enabled, and correct any problems with these propagations and apply processes.
- The Streams pool might be too small to hold the captured messages. In this case, increase the size of the Streams pool. You can also configure Automatic Shared Memory Management to manage the size of the Streams pool automatically. Set the SGA_TARGET initialization parameter to use Automatic Shared Memory Management.

Streams Capture - (%) Cumulative Spilled Messages

The percentage of Cumulative spilled messages indicate the messages are staying in memory longer. It can also indicate that the Propagation or Apply Process is slow to consume the enqueued messages.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data source for this metric is the target database, gv$buffered_queues table.

User Action

No user action is required.

Streams Capture - (%) Spilled Messages

Queue spill indicates the messages are staying in memory longer. It can also indicate that the Propagation or Apply Process is slow to consume the enqueued messages.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	Spilled messages for Capture process %CAPTURE_NAME% queue is %value% percent.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Capture Name object. If warning or critical threshold values are currently set for any Capture Name object, those thresholds can be viewed on the Metric Detail page for this metric. To specify or

change warning or critical threshold values for each Capture Name object, use the Edit Thresholds page.

Data Source

The data source for this metric is the target database, gv$buffered_queues table.

User Action

Increase Streams Pool Size to avoid queue spills.

Streams Latency and Throughput

The metrics in this category collect information about latency and throughput for each capture, propagation and apply component in the database. Latency and throughput are important indicators for the overall performance of the streams path.

Latency

This metric reports latency. High Latency indicates that the components are slow.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	Latency for Streams %streams_process_type% Process %streams_process_name% is %value% seconds.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects.

If warning or critical threshold values are currently set for any unique combination of Streams Process Name and Streams Process Type objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects, use the Edit Thresholds page.

Data Source

The data source for this metric is the target database, gv$streams_capture, gv$propagation_sender, and gv$streams_apply_server views.

User Action

Identify and correct the least performing component in the streams configuration.

Throughput (per sec)

This metric collects information about throughput for each capture, propagation and apply component in the database

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	Throughput for Streams %streams_process_type% Process %streams_process_name% is %value% messages/sec.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects.

If warning or critical threshold values are currently set for any unique combination of Streams Process Name and Streams Process Type objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects, use the Edit Thresholds page.

Data Source

Not available.

User Action

The required actions are specific to your site.

Total Messages

This metric collects the total number of messages for each capture, propagation and apply component in the database.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

Not available.

User Action

The required actions are specific to your site.

Streams Pool Usage

The metrics in this category check for the memory usage of the Streams pool.

Streams Pool Full

This alert is generated when the memory usage of the Streams pool has exceeded the percentage specified by the STREAMS_POOL_USED_PCT metric. This alert can be raised only if the database is not using Automatic Memory Management or Automatic Shared Memory Management.

Data Source

Not available.

User Action

If the currently running workload is typical, consider increasing the size of the Streams pool.

Streams Processes Count

The metrics in this category show the total number of Streams capture processes, propagations, and apply processes at the local database. This metric also shows the number of capture processes, propagations, and apply processes that have encountered errors.

Number of Apply Processes Having Errors

This metric shows the number of apply processes that have encountered errors at the local database.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The information in this metric is in the DBA_APPLY data dictionary view.

User Action

If an apply process has encountered errors, then correct the conditions that caused the errors.

Number of Capture Processes Having Errors

This metric shows the number of capture processes that have encountered errors at the local database.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The information in this metric is in the DBA_CAPTURE data dictionary view.

User Action

If a capture process has encountered errors, then correct the conditions that caused the errors.

Number of Apply Processes

This metric shows the number of apply processes at the local database.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The information in this metric is in the DBA_APPLY data dictionary view.

User Action

Use this metric to determine the total number of apply processes at the local database.

Number of Capture Processes

This metric shows the number of capture processes at the local database.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The information in this metric is in the DBA_CAPTURE data dictionary view.

User Action

Use this metric to determine the total number of capture processes at the local database.

Number of Propagation Jobs

This metric shows the number of propagations at the local database.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The information in this metric is in the DBA_PROPAGATION data dictionary view.

User Action

Use this metric to determine the total number of propagations at the local database.

Number of Propagations Having Errors

This metric shows the number of propagations that have encountered errors at the local database.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The information in this metric is in the DBA_PROPAGATION data dictionary view.

User Action

If a propagation has encountered errors, then correct the conditions that caused the errors.

Total Number of Propagation Errors

This metric provides the total number of propagation errors.

Target Version	Collection Frequency
All versions	Every 5 Minutes

Data Source

The data source for this metric is the target database, DBA_Propagation view.

User Action

No user action is required.

Streams Processes Status

The metrics in this category collect the current status and number of errors for each capture, propagation and apply process in the database.

Streams Process Errors

This metric collects the number of errors for each capture, propagation and apply process in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	0	Not Defined	Stream component %streams_process_name% has %value% errors.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects.

If warning or critical threshold values are currently set for any unique combination of Streams Process Name and Streams Process Type objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects, use the Edit Thresholds page.

Data Source

Not available.

User Action

The required actions are specific to your site.

Streams Process Status

This metric collects the current status and number of errors for each capture, propagation, and apply process in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	DISABLED	ABORTED	Status for Streams process %streams_process_name% is %streams_process_status%.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects.

If warning or critical threshold values are currently set for any unique combination of Streams Process Name and Streams Process Type objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects, use the Edit Thresholds page.

Data Source

The data source for this metric is the target database, DBA_CAPTURE, dba_propagation, dba_apply views.

User Action

Analyze status change reason and enable the disabled/aborted component.

Streams Propagation - Messages State Stats

The metrics in this category collect the number of messages in Ready and Waiting state for each Propagation process.

Number of Ready Messages

This metric collects the number of messages in Ready state for each Propagation process.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data source for this metric is the target database, source and destination queues.

User Action

No user action is required.

Number of Waiting Messages

This metric collects the number of messages in Waiting state for each Propagation process.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data source for this metric is the target database, source and destination queues.

User Action

No user action is required.

Streams Prop - (%) Messages in Waiting State

This metric collects the percentage of messages in Waiting state for each Propagation process.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	Messages waiting for %PROPAGATION_NAME% queue is %value% percent.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Propagation Name and Messages Delivery Mode objects.

If warning or critical threshold values are currently set for any unique combination of Propagation Name and Messages Delivery Mode objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Propagation Name and Messages Delivery Mode objects, use the Edit Thresholds page.

Data Source

The data source for this metric is the target database, source and destination queues.

User Action

No user action is required.

Streams Propagation - Queue Propagation

The metrics in this category collect propagation statistics in terms of number of messages and number of Kbytes propagated by each propagation process.

Message Propagation Rate (per Sec)

This metric collects propagation statistics in terms of the rate of messages propagated by each propagation process.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data source for this metric is the target database - DBA_PROPAGATION.

User Action

No user action is required.

Rate of KBytes Propagated (per Sec)

This metric collects propagation statistics in terms of the rate of Kbytes propagated by each propagation process.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data source for this metric is the target database - DBA_PROPAGATION.

User Action

No user action is required.

Total Number of KBytes Propagated

This metric collects propagation statistics in terms of total number of Kbytes propagated by each propagation process.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data source for this metric is the target database - DBA_PROPAGATION.

User Action

No user action is required.

Total Number of Messages Propagated

This metric collects propagation statistics in terms of the total number of messages propagated by each propagation process.

Target Version	Collection Frequency
All versions	Every 30 Minutes

Data Source

The data source for this metric is the target database - DBA_PROPAGATION.

User Action

No user action is required.

Streams Propagation Aborted

The metrics in this category check for the Streams Propagation processes.

> **Note:**
>
> This is a server-generated alert.

Streams Propagation Process Aborted

This metric detected when a Streams Propagation process configured on this database aborts. This alert indicates a critical error.

Data Source

Not Available

User Action

Obtain the exact error message in dba_queue_schedules, take the appropriate action for this error, and restart the propagation process using dbms_propagation_adm.start_propagation.

System Response Time Per Call

The metrics in this category represent the system response time.

Response Time (centi-seconds per call)

This metric represents the average time taken for each call (both user calls and recursive calls) within the database. A change in this value indicates that either the workload has changed or that the database's ability to process the workload has changed because of either resource constraints or contention.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	Not Defined

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

System Time Model

This section provides information on the metrics in the System Time Model category.

Target Version	Evaluation and Collection Frequency
All versions	Every 15 minutes

Metric Name	Description
DB Background CPU Per Second	Time spent on CPU by database background processes.
DB CPU Per Second	CPU usage by the database processes, per second.
DB Time Per Second	Total foreground process time spent on database calls. This includes CPU time, I/O time, and non-idle wait time.
Failed Parsing (SQL) Time Per Second	Time spent performing SQL parses that ultimately failed with a parse error.
Hard Parsing (Bind Mismatch to Cursor) Time Per Second	Time spent performing SQL hard parses when the hard parse resulted from bind type or bind size mismatch with an existing cursor in the SQL cache.
Hard Parsing (Inability to Share Cursor in SQL Cache) Time Per Second	Time spent performing SQL hard parses when the hard parse resulted from not being able to share an existing cursor in the SQL cache.
Hard Parsing Time Per Second	Time spent hard parsing SQL statements.
Inbound PL/SQL RPC Time Per Second	Time spent on executing inbound PL/SQL remote procedure calls. This includes the time spent recursively executing SQL and Java.
Java Execution Time Per Second	Time spent running the Java VM. This does not include the time spent recursively executing or parsing SQL statements or the time spent recursively executing PL/SQL.
Loading next Sequence Number Time Per Second	Time spent getting the next sequence number from the data dictionary. If a sequence is cached, then this is the time spent replenishing the cache when it runs out. No time is charged when a sequence number is found in the cache. For non-cached sequences, some time will be charged for every `nextval` call.
PL/SQL Compiling Time Per Second	Time spent running the PL/SQL compiler.
PL/SQL Running Time Per Second	Time spent running the PL/SQL interpreter. This does not include the time spent recursively executing or parsing SQL statements or the time spent recursively executing the Java VM.
Parsing Time Per Second	Time spent parsing SQL statements. This includes both soft and hard parse time.
RMAN Backup/Restore Time Per Second	CPU time spent in RMAN backup and restore operations.
Rebinding Time Per Second	Time spent giving new values to bind variables.
SQL Execution Time Per Second	Time spent on SQL execution. For `select` statements, this also includes the time spent performing fetches of query results.
Session Connect/Disconnect Time Per Second	Time spent performing session connect and disconnect calls.
Time Spent in SQL parsing where Parsing fails (ORA-04031) Per Second	Time spent performing SQL parses that ultimately fail with error `ORA-04031`.

Tablespace Allocation

The metrics in this category check the amount of space used and the amount of space allocated to each tablespace. The used space can then be compared to the allocated space to determine how much space is unused in the tablespace. This metric is not intended for alerts. Rather it is intended for reporting. Historical views of unused allocated free space can help DBAs to correctly size their tablespaces, eliminating wasted space.

Tablespace Allocated Space (MB)

The allocated space of a tablespace is the sum of the current size of its datafiles. A portion of this allocated space is used to store data while some may be free space. If segments are added to a tablespace, or if existing segments grow, they will use the allocated free space. The allocated free space is only available to segments within the tablespace. If, over time, the segments within a tablespace are not using this free space, then the allocated free space is being unused.

This metric calculates the space allocated for each tablespace. It is not intended to generate alerts. Rather it should be used in conjunction with the Allocated Space Used (MB) metric to produce an historical view of the amount of space being used and unused by each tablespace.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

Tablespace Allocated Space (MB) is calculated by querying the DBA_TABLESPACES, DBA_UNDO_EXTENTS, DBA_DATA_FILES, DBA_FREE_SPACE and DBA_TEMP_FILES data dictionary views.

User Action

Specific to your site.

Tablespace Used Space (MB)

The allocated space of a tablespace is the sum of the current size of its datafiles. Some of this allocated space is used to store data and some of it may be free space. If segments are added to a tablespace, or if existing segments grow, they will use the allocated free space. The allocated free space is only available to segments within the tablespace. If, over time, the segments within a tablespace are not using this free space, then the allocated free space is being wasted.

This metric calculates the space used for each tablespace. It is not intended to generate alerts. Rather it should be used in conjunction with the Tablespace Allocated Space (MB) metric to produce an historical view of the amount of space being used and unused by each tablespace.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

Tablespace Used Space (MB) is Tablespace Allocated Space (MB) - Tablespace Allocated Free Space (MB) where:

- Tablespace Allocated Space (MB) is calculated by looping through the tablespace's data files and totaling the size of the data files.
- Tablespace Allocated Free Space (MB) is calculated by looping through the tablespace's data files and totaling the size of the free space in each data file.

User Action

Specific to you site.

Tablespaces Full

The metrics in this category check for the amount of space used by each tablespace. The used space is then compared to the available free space to determine tablespace fullness. The available free space takes into account the maximum data file size as well as available disk space. This means that a tablespace will not be flagged as full if datafiles can extend and there is enough disk space available for them to extend.

Tablespace Free Space (MB)

As segments within a tablespace grow, the available free space decreases. If there is no longer any available free space, meaning datafiles have hit their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

This metric checks for the total available free space in each tablespace. This metric is intended for larger tablespaces, where the Available Space Used (%) metric is less meaningful. If the available free space falls below the size specified in the threshold arguments, then a warning or critical alert is generated.

> **Note:**
>
> This metric collects data for locally managed permanent tablespaces only. The Tablespace Free Space (MB) (Temp) metric collects data for temporary tablespaces. The Tablespace Free Space (MB) (Undo) metric collects data for Undo tablespaces.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	Tablespace [%name%] only has [%value% megabytes] free space

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Tablespace Name object.

You can also set default warning and critical thresholds that will be used for all tablespaces that do not have their own defined thresholds.

If warning or critical threshold values are currently set for any Tablespace Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Tablespace Name object:

1. From the target's home page menu, select **Monitoring**, then **Metric and Collection Settings**. The Metric and Collection Settings page appears.
2. Click the pencil icon for the Tablespaces Full metric to access the Edit Advanced Settings.

Data Source

The data for this metric is derived from the MaximumSize - Total Used Space formula where:

- TotalUsedSpace: total used space in MB of tablespace.
- MaximumSize: Maximum size (in MB) of the tablespace. The maximum size is determined by looping through the tablespaces data files, as well as additional free space on the disk that would be available for the tablespace should a data file autoextend.

User Action

Perform one of the following:

- Increase the size of the tablespace by: Enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.
- If the tablespace is suffering from tablespace free space fragmentation problems, consider reorganizing the entire tablespace.
- Relocate segments to another tablespace, thus increasing the free space in this tablespace.
- Run the Segment Advisor on the tablespace.

Tablespace Space Used (%)

As segments within a tablespace grow, the available free space decreases. If there is no longer any available free space, meaning datafiles have hit their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

> **Note:**
>
> This metric collects data for locally managed permanent tablespaces only. The Tablespace Space Used (%) (Temp) metric collects data for temporary tablespaces. The Tablespace Space Used (%) (Undo) metric collects data for Undo tablespaces.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Every 30 Minutes	85	97	Management Agent generates alert message.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Tablespace Name object. You can also set default warning and critical thresholds that will be used for all tablespaces that do not have their own defined thresholds.

If warning or critical threshold values are currently set for any Tablespace Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Tablespace Name object:

1. From the target's home page menu, select **Monitoring**, then **Metric and Collection Settings**. The Metric and Collection Settings page appears.
2. Click the pencil icon for the required metric to access the Edit Advanced Settings.

Data Source

The data for this metric is derived from the (TotalUsedSpace / MaximumSize) * 100 formula where:

- TotalUsedSpace: total used space in MB of tablespace.
- MaximumSize: Maximum size (in MB) of the tablespace. The maximum size is determined by looping through the tablespace's data files.

User Action

Perform one of the following:

- Increase the size of the tablespace by: Enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.
- If the tablespace is suffering from tablespace free space fragmentation problems, consider reorganizing the entire tablespace.
- Relocate segments to another tablespace, thus increasing the free space in this tablespace.
- Run the Segment Advisor on the tablespace.

Tablespaces Full (Temp)

The metrics in this category check for the amount of space used by each locally managed temporary tablespace. The used space is then compared to the available free space to determine tablespace fullness. The available free space takes into account the maximum data file size as well as available disk space. This means that a tablespace will not be flagged as full if data files can extend and there is enough disk space available for them to extend.

> **Note:**
>
> These metrics collect data for locally managed temporary tablespaces.

> **Note:**
>
> Temporary tablespaces do not typically grow in a steady manner but are subject to spikes of high usage. For this reason, thresholds for both Tablespace Free Space (MB) (Temp) and Tablespace Space Used (%) (Temp) are not defined. Take care when you are setting thresholds for these metrics to avoid unwanted alerts.

Tablespace Free Space (MB) (Temp)

As segments within a tablespace grow, the available free space decreases. If there is no more free space available, that is, the data files have hit their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

This metric checks for the total available free space in each temporary tablespace. This metric is intended for larger temporary tablespaces, where the Available Space Used (%) metric is less meaningful. If the available free space falls below the size specified in the threshold arguments, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	Tablespace [%name%] has [%value% mbytes] free

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each locally managed temporary Tablespace Name object.

You can also set default warning and critical thresholds that will be used for all locally managed temporary tablespaces that do not have their own defined thresholds.

If warning or critical threshold values are currently set for any Tablespace Name object, you can view those thresholds from the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Tablespace Name object:

1. From the target's home page menu, select **Monitoring**, then **Metric and Collection Settings**. The Metric and Collection Settings page appears.
2. Click the pencil icon for the required metric to access the Edit Advanced Settings.

Data Source

The data for this metric is retrieved from the DBA_TABLESPACE_USAGE_METRICS data dictionary view.

MB Free: TABLESPACE_SIZE (Total size of the tablespace) - USED_SPACE (Total space consumed by the tablespace)

User Action

Perform one of the following:

- Increase the size of the tablespace by either enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.

- If the tablespace is suffering from tablespace free space fragmentation problems, consider reorganizing the entire tablespace.
- Create additional temporary tablespaces.

Tablespace Space Used (%) (Temp)

As segments within a tablespace grow, the available free space decreases. If there is no more free space available, that is, the data files have hit their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	[%name%] is [%value% percent] full

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each locally managed temporary Tablespace Name object. You can also set default warning and critical thresholds that will be used for all locally managed temporary tablespaces that do not have their own defined thresholds.

If warning or critical threshold values are currently set for any Tablespace Name object, you can view those thresholds from the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Tablespace Name object:

1. From the target's home page menu, select **Monitoring**, then **Metric and Collection Settings**. The **Metric and Collection Settings** page appears.
2. Click the pencil icon for the required metric to access the Edit Advanced Settings.

Data Source

The data for this metric is retrieved from the DBA_TABLESPACE_USAGE_METRICS data dictionary view.

Used Percent: USED_PERCENT, Percentage of used space, as a function of the maximum possible tablespace size

User Action

Perform one of the following:

- Increase the size of the tablespace by either enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.
- If the tablespace is suffering from tablespace free space fragmentation problems, consider reorganizing the entire tablespace.
- Create additional temporary tablespaces.

Tablespaces Full (Undo)

The metrics in this category check for the amount of space used by each Undo tablespace. The used space is then compared to the available free space to determine tablespace fullness. The available free space takes into account the maximum data file size as well as available

disk space. This means that a tablespace will not be flagged as full if data files can extend and there is enough disk space available for them to extend.

> **Note:**
>
> Undo tablespaces do not typically grow in a steady manner but are subject to spikes of high usage. For this reason, thresholds for both Tablespace Free Space (MB) (Undo) and Tablespace Space Used (%) (Undo) are not defined. Take care when setting thresholds for these metrics to avoid unwanted alerts.

Tablespace Free Space (MB) (Undo)

As segments within a tablespace grow, the available free space decreases. If there is no more free space available, that is, the data files have hit their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

This metric checks for the total available free space in each Undo tablespace. This metric is intended for larger Undo tablespaces, where the Available Space Used (%) metric is less meaningful. If the available free space falls below the size specified in the threshold arguments, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	Tablespace [%name%] has [%value% mbytes] free

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Undo Tablespace Name object.

You can also set default warning and critical thresholds that will be used for all Undo tablespaces that do not have their own defined thresholds.

If warning or critical threshold values are currently set for any Tablespace Name object, you can view those thresholds from the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Tablespace Name object:

1. From the target's home page menu, select **Monitoring**, then **Metric and Collection Settings**. The **Metric and Collection Settings** page appears.
2. Click the pencil icon for the required metric to access the Edit Advanced Settings.

Data Source

The data for this metric is retrieved from the DBA_TABLESPACE_USAGE_METRICS data dictionary view.

MB Free: TABLESPACE_SIZE (Total size of the tablespace) - USED_SPACE (Total space consumed by the tablespace)

User Action

Perform one of the following:

- Increase the size of the tablespace by either enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.
- If the tablespace is suffering from tablespace free space fragmentation problems, then consider reorganizing the entire tablespace.
- Use Undo Advisor (Automatic Undo Management) to obtain sizing advice and manage the Undo tablespaces.

Tablespace Space Used (%) (Undo)

As segments within a tablespace grow, the available free space decreases. If there is no longer any available free space, meaning data files have hit their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	Not Defined	Not Defined	Tablespace [%name%] is [%value% percent] full

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Undo Tablespace Name object. You can also set default warning and critical thresholds that will be used for all Undo tablespaces that do not have their own defined thresholds.

If warning or critical threshold values are currently set for any Tablespace Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Tablespace Name object:

1. From the target's home page menu, select **Monitoring**, then **Metric and Collection Settings**. The **Metric and Collection Settings** page appears.
2. Click the pencil icon for the required metric to access the Edit Advanced Settings.

Data Source

The data for this metric is retrieved from the DBA_TABLESPACE_USAGE_METRICS data dictionary view.

Used Percent: USED_PERCENT, Percentage of used space, as a function of the maximum possible tablespace size

User Action

Perform one of the following:

- Increase the size of the tablespace by either enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.
- If the tablespace is suffering from tablespace free space fragmentation problems, then consider reorganizing the entire tablespace.
- Use Undo Advisor (Automatic Undo Management) to obtain sizing advice and manage the Undo tablespaces.

Temporary File Status

The metrics in this category provide the name and status of temporary files.

Status

This metric displays the status of a temporary file.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	OFFLINE	Not Defined	The temporary file %NAME% is %STATUS%.

User Action

Temporary files that are offline can be placed online by selecting the **Place Online** action from the **Datafiles** page in the Enterprise Manager console.

To access the **Datafiles** page, from the database target's home page, select **Administration**, then **Storage**, and then **Datafiles**.

Throughput

The metrics in this category represent rates of resource consumption or throughput.

Average Active Sessions

This metric represents the average active sessions at a point in time. It is the number of sessions that are either working or waiting.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	Not Defined

Data Source

Not available.

User Action

No user action is required.

Average Synchronous Single-Block Read Latency (ms)

The average latency in milliseconds of a synchronous single-block read. Synchronous single-block reads are a reasonably accurate way of assessing the performance of the storage subsystem. High latencies are typically caused by a high I/O request load. Excessively high CPU load can also cause the latencies to increase.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Not Defined	Not Defined	Not Defined	Not Defined

Data Source

The source of the data is the v$sysmetric view.

User Action

First, verify that your storage subsystem is not operating with component failures, for example, disk, network, or HBA failures. If no issues are found, consider upgrading your storage subsystem.

BG Checkpoints (per second)

This metric represents the BG checkpoints per second.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

Not available.

User Action

The required actions are specific to your site.

Branch Node Splits (per second)

Number of times per second an index branch block was split because of the insertion of an additional value.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

the branch node splits / time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Branch Node Splits (per transaction)

Number of times per transaction an index branch block was split because of the insertion of an additional value.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

branch node splits / transaction

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Consistent Read Blocks Created (per second)

This metric represents the number of current blocks per second cloned to create consistent read (CR) blocks.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 5 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

CR blocks created / time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Consistent Read Blocks Created (per transaction)

This metric represents the number of current blocks per transaction cloned to create consistent read (CR) blocks.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

CR blocks created / transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Consistent Read Changes (per second)

This metric represents the number of times per second a user process has applied rollback entries to perform a consistent read on the block.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

consistent changes / time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Consistent Read Changes (per transaction)

This metric represents the number of times per transaction a user process has applied rollback entries to perform a consistent read on the block.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

consistent changes / transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Consistent Read Gets (per second)

This metric represents the number of times per second a consistent read was requested for a block.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

consistent gets/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Consistent Read Gets (per transaction)

This metric represents the number of times per transaction a consistent read was requested for a block.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

consistent gets/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Consistent Read Undo Records Applied (per second)

This metric represents the number of undo records applied for consistent read per second.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

current blocks converted for CR/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Consistent Read Undo Records Applied (per transaction)

This metric represents the consistent read undo records applied per transaction.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

Not available.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Cumulative Logons (per second)

This metric represents the number of logons per second during the sample period.

This test checks the number of logons that occurred per second during the sample period. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Not Defined	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

DeltaLogons / Seconds where:

- DeltaLogons: difference in 'select value from v$sysstat where name='logons cumulative'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

A high logon rate may indicate that an application is inefficiently accessing the database. Database logon's are a costly operation. If an application is performing a logon for every SQL access, that application will experience poor performance as well as affect the performance of other applications on the database. If there is a high logon rate, try to identify the application that is performing the logons to determine if it could be redesigned such that session connections could be pooled, reused, or shared.

Cumulative Logons (per transaction)

This metric represents the number of logons per transaction during the sample period.

The value of this statistic will be zero if there have not been any write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name will be a better indicator of current performance.

This test checks the number of logons that occurred per transaction. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

DeltaLogons/Transactions where:

- DeltaLogons: difference in 'select value from v$sysstat where name='logons cumulative'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

A high logon rate may indicate that an application is inefficiently accessing the database. Database logon's are a costly operation. If an application is performing a logon for every SQL access, that application will experience poor performance as well as affect the performance of other applications on the database. If there is a high logon rate try to identify the application that is performing the logons to determine if it could be redesigned such that session connections could be pooled, reused or shared.

Database Block Changes (per second)

This metric represents the total number of changes per second that were part of an update or delete operation that were made to all blocks in the SGA.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

db block changes/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Database Block Changes (per transaction)

This metric represents the total number of changes per transaction that were part of an update or delete operation that were made to all blocks in the SGA.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

db block changes/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Database Block Gets (per second)

This metric represents the number of times per second a current block was requested.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

db block gets/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Database Block Gets (per transaction)

This metric represents the number of times per transaction a current block was requested.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

db block gets/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Database Time (centiseconds per second)

This metric denotes the database time.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	Not Defined

Data Source

Not available.

User Action

The required actions are specific to your site.

DBWR Checkpoints (per second)

This metric represents the number of times, per second, during this sample period DBWn was asked to scan the cache and write all blocks marked for a checkpoint.

The database writer process (DBWn) writes the contents of buffers to datafiles. The DBWn processes are responsible for writing modified (dirty) buffers in the database buffer cache to disk.

When a buffer in the database buffer cache is modified, it is marked dirty. The primary job of the DBWn process is to keep the buffer cache clean by writing dirty buffers to disk. As user processes dirty buffers, the number of free buffers diminishes. If the number of free buffers drops too low, user processes that must read blocks from disk into the cache are not able to find free buffers. DBWn manages the buffer cache so that user processes can always find free buffers.

When the Oracle Server process cannot find a clean reusable buffer after scanning a threshold of buffers, it signals DBWn to write. When this request to make free buffers is received, DBWn writes the least recently used (LRU) buffers to disk. By writing the least recently used dirty buffers to disk, DBWn improves the performance of finding free buffers while keeping recently used buffers resident in memory. For example, blocks that are part of frequently accessed small tables or indexes are kept in the cache so that they do not need to be read in again from disk. The LRU algorithm keeps more frequently accessed blocks in the buffer cache so that when a buffer is written to disk, it is unlikely to contain data that may be useful soon.

Additionally, DBWn periodically writes buffers to advance the checkpoint that is the position in the redo log from which crash or instance recovery must begin.

This test checks the number of times DBWR was asked to advance the checkpoint. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

DeltaCheckpoints/Seconds where:

- DeltaCheckpoints: difference in 'select value from v$sysstat where name='DBWR checkpoints'' between sample end and start
- Seconds: number of seconds in sample period

User Action

A checkpoint tells the DBWR to write out modified buffers to disk. This write operation is different from the make free request in that the modified buffers are not marked as free by the DBWR process. Dirty buffers may also be written to disk at this time and freed.

The write size is dictated by the _db_block_checkpoint_batch parameter. If writing, and subsequently waiting for checkpoints to complete is a problem, the checkpoint completed event displays in the Top Waits page sorted by Time Waited or the Sessions Waiting for this Event page.

If the database is often waiting for checkpoints to complete you may want to increase the time between checkpoints by checking the init.ora parameter db_block_checkpoint_batch: select name, value, is default from v$parameter where name = db_block_checkpoint_batch. The value should be large enough to take advantage of parallel writes. The DBWR uses a write batch that is calculated like this: (db_files * db_file_simultaneous_writes)/2 The write_batch is also limited by two other factors:

- A port specific limit on the numbers of I/Os (compile time constant).
- 1/4 of the number of buffers in the SGA.

The db_block_checkpoint_batch is always smaller or equal to the _db_block_write_batch. You can also consider enabling the check point process.

Enqueue Deadlocks (per second)

This metric represents the number of times per second that a process detected a potential deadlock when exchanging two buffers and raised an internal, restartable error.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

enqueue deadlocks/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Enqueue Deadlocks (per transaction)

This metric represents the number of times per transaction that a process detected a potential deadlock when exchanging two buffers and raised an internal, restartable error.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

enqueue deadlocks/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Enqueue Requests (per second)

This metric represents the total number of table or row locks acquired per second.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

enqueue requests/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Enqueue Requests (per transaction)

This metric represents the total number of table or row locks acquired per transaction.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

enqueue requests/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Enqueue Timeout (per second)

This metric represents the total number of table and row locks (acquired and converted) per second that time out before they could complete.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

enqueue timeouts/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Enqueue Timeout (per transaction)

This metric represents the total number of table and row locks (acquired and converted) per transaction that timed out before they could complete.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

enqueue timeouts/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Enqueue Waits (per second)

This metric represents the total number of waits per second that occurred during an enqueue convert or get because the enqueue get was deferred.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

enqueue waits/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Enqueue Waits (per transaction)

This metric represents the total number of waits per transaction that occurred during an enqueue convert or get because the enqueue get was deferred.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

enqueue waits / transaction

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Executes (per second)

This metric represents the rate of SQL command executions over the sampling interval.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

DeltaExecutions / Seconds where:

- DeltaExecutions: difference in 'select value from v$sysstat where name='execute count'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

No user action is necessary.

Executes Performed without Parses (%)

This metric represents the percentage of statement executions that do not require a corresponding parse. A perfect system would parse all statements once and then execute the parsed statement over and over without reparsing. This ratio provides an indication as to how often the application is parsing statements as compared to their overall execution rate. A higher number is better.

This test checks the percentage of executes that do not require parses. If the value is less than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

((DeltaExecuteCount - (DeltaParseCountTotal)) / DeltaExecuteCount) * 100 where:

- DeltaParseCountTotal: difference in 'select value from v$sysstat where name='parse count (total)'' between sample end and start
- DeltaExecuteCount: difference in 'select value from v$sysstat where name='execute count'' between sample end and start

User Action

An execute to parse ratio of less than 70% indicates that the application may be parsing statements more often than it should. Reparsing the statement, even if it is a soft parse, requires a network round trip from the application to the database, as well as requiring the processing time to locate the previously compiled statement in the cache. Reducing network round trips and unnecessary processing improves application performance.

Use the Top Sessions page sorted by Parses to identify the sessions responsible for the bulk of the parse activity within the database. Start with these sessions to determine whether the application could be modified to make more efficient use of its cursors.

Full Index Scans (per second)

This metric represents the number of fast full index scans per second.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 5 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

index fast full scans (full)/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Full Index Scans (per transaction)

This metric represents the number of fast full index scans per transaction.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

index fast full scans (full)/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Hard Parses (per second)

This metric represents the number of hard parses per second during this sample period. A hard parse occurs when a SQL statement has to be loaded into the shared pool. In this case, the Oracle Server has to allocate memory in the shared pool and parse the statement.

Each time a particular SQL cursor is parsed, this count will increase by one. There are certain operations that will cause a SQL cursor to be parsed. Parsing a SQL statement breaks it down into atomic steps, which the optimizer will evaluate when generating an execution plan for the cursor.

This test checks the number of parses of statements that were not already in the cache. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

DeltaParses / Seconds where:

- DeltaParses: difference in 'select value from v$sysstat where name='parse count (hard)'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

If there appears to be excessive time spent parsing, evaluate SQL statements to determine those that can be modified to optimize shared SQL pool memory use and avoid unnecessary statement reparsing. This type of problem is commonly caused when similar SQL statements are written which differ in space, case, or some combination of the two. You may also consider using bind variables rather than explicitly specified constants in your statements whenever possible.

The Top Sessions page sorted by Hard Parses will show you which sessions are incurring the most hard parses. Hard parses happen when the server parses a query and cannot find an exact match for the query in the library cache. Hard parses can be avoided by sharing SQL statements efficiently. The use of bind variables instead of literals in queries is one method to increase sharing.

By showing you which sessions are incurring the most hard parses, this page may lead you to the application or programs that are the best candidates for SQL rewrites.

Also, examine SQL statements which can be modified to optimize shared SQL pool memory use and avoid unnecessary statement reparsing. This type of problem is commonly caused when similar SQL statements are written which differ in space, case, or some combination of the two. You may also consider using bind variables rather than explicitly specified constants in your statements whenever possible.

The SHARED_POOL_SIZE initialization parameter controls the total size of the shared pool. Consider increasing the SHARED_POOL_SIZE to decrease the frequency in which SQL requests are being flushed from the shared pool to make room for new requests.

To take advantage of the additional memory available for shared SQL areas, you may also need to increase the number of cursors permitted per session. You can increase this limit by increasing the value of the initialization parameter OPEN_CURSORS.

Hard Parses (per transaction)

This metric represents the number of hard parses per second during this sample period. A hard parse occurs when a SQL statement has to be loaded into the shared pool. In this case, the Oracle Server has to allocate memory in the shared pool and parse the statement.

Each time a particular SQL cursor is parsed, this count will increase by one. There are certain operations which will cause a SQL cursor to be parsed. Parsing a SQL statement breaks it down into atomic steps which the optimizer will evaluate when generating an execution plan for the cursor. The value of this statistic will be zero if there have not been any write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name will be a better indicator of current performance.

This test checks the number of hard parses per second during this sample period. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	>	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data for this metric is derived by the following formula:

DeltaParses/Transactions where:

- DeltaParses: difference in 'select value from v$sysstat where name='parse count (hard)'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

If there appears to be excessive time spent parsing, evaluate SQL statements to determine which can be modified to optimize shared SQL pool memory use and avoid unnecessary

statement reparsing. This type of problem is commonly caused when similar SQL statements are written which differ in space, case, or some combination of the two. You may also consider using bind variables rather than explicitly specified constants in your statements whenever possible.

The Top Sessions page sorted by Hard Parses will show you which sessions are incurring the most hard parses. Hard parses happen when the server parses a query and cannot find an exact match for the query in the library cache. Hard parses can be avoided by sharing SQL statements efficiently. The use of bind variables instead of literals in queries is one method to increase sharing.

By showing you which sessions are incurring the most hard parses, this page may lead you to the application or programs that are the best candidates for SQL rewrites.

Also, examine SQL statements which can be modified to optimize shared SQL pool memory use and avoid unnecessary statement reparsing. This type of problem is commonly caused when similar SQL statements are written which differ in space, case, or some combination of the two. You may also consider using bind variables rather than explicitly specified constants in your statements whenever possible.

The SHARED_POOL_SIZE initialization parameter controls the total size of the shared pool. Consider increasing the SHARED_POOL_SIZE to decrease the frequency in which SQL requests are being flushed from the shared pool to make room for new requests.

To take advantage of the additional memory available for shared SQL areas, you may also need to increase the number of cursors permitted per session. You can increase this limit by increasing the value of the initialization parameter OPEN_CURSORS.

I/O Megabytes (per second)

The total I/O throughput of the database for both reads and writes in megabytes per second. A very high value indicates that the database is generating a significant volume of I/O data.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	Not Defined

Data Source

The source of the data is the v$sysmetric view.

User Action

A high I/O throughput value is not in itself problematic. However, if high I/O latencies (for example, Synchronous Single-Block Read Latencies are causing a performance problem, then reducing the total I/O throughput may help. The source of the I/O throughput can be investigated by viewing a breakdown by either Component or Resource Consumer Group.

I/O Requests (per second)

This metric represents the total rate of I/O read and write requests for the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	Not Defined

Data Source

The source of the data is the v$sysmetric view.

User Action

A high I/O request rate is not in itself problematic. However, if high I/O latencies (for example, Synchronous Single-Block Read Latencies are causing a performance problem, then reducing the total I/O request rate may help. The source of the I/O requests can be investigated by viewing a breakdown by either Component or Resource Consumer Group.

Leaf Node Splits (per second)

Number of times per second an index leaf node was split because of the insertion of an additional value.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

leaf node splits / time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Leaf Node Splits (per transaction)

This metric reports the number of times per transaction an index leaf node was split because of the insertion of an additional value.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

leaf node splits / transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Network Bytes (per second)

This metric represents the total number of bytes sent and received through the SQL Net layer to and from the database.

This test checks the network read/write per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

(DeltaBytesFromClient+DeltaBytesFromDblink+DeltaBytesToClient+DeltaBytesToDblink) / Seconds where:

- Delta Bytes From Client: difference in 'select s.value from v$sysstat s, visitation n where n.name='bytes received via SQL*Net from client' and n.statistic#=s.statistic#' between end and start of sample period
- DeltaBytesFromClient: difference in 'select s.value from v$sysstat s, v$statname n where n.name='bytes received via SQL*Net from dblink' and n.statistic#=s.statistic#' between end and start of sample period
- DeltaBytesFromClient: difference in 'select s.value from v$sysstat s, v$statname n where n.name='bytes sent via SQL*Net to client' and n.statistic#=s.statistic#' between end and start of sample period
- DeltaBytesFromClient: difference in 'select s.value from v$sysstat s, v$statname n where n.name='bytes sent via SQL*Net to dblink' and n.statistic#=s.statistic#' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

This metric represents the amount of network traffic in and out of the database. This number may only be useful when compared to historical levels to understand network traffic usage related to a specific database.

Number of Transactions (per second)

This metric represents the total number of commits and rollbacks performed during this sample period.

This test checks the number of commits and rollbacks performed during sample period. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaCommits + DeltaRollbacks where:

- DeltaCommits: difference of 'select value from v$sysstat where name='user commits'' between sample end and start
- DeltaRollbacks: difference of 'select value from v$sysstat where name='user rollbacks'' between sample end and start

User Action

This statistic is an indication of how much work is being accomplished within the database. A spike in the transaction rate may not necessarily be bad. If response times stay close to normal, it means your system can handle the added load. Actually, a drop in transaction rates and an increase in response time may be indicators of problems. Depending upon the application, transaction loads may vary widely across different times of the day.

Open Cursors (per second)

This metric represents the total number of cursors opened per second.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

opened cursors cumulative/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Open Cursors (per transaction)

This metric represents the total number of cursors opened per transaction.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

opened cursors cumulative/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Parse Failure Count (per second)

This metric represents the total number of parse failures per second.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

parse count (failures)/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Parse Failure Count (per transaction)

This metric represents the total number of parse failures per transaction.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

parse count (failures)/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Physical Reads (per second)

This metric represents the number of data blocks read from disk per second during this sample period. When a user performs a SQL query, Oracle tries to retrieve the data from the database buffer cache (memory) first, then searches the disk if it is not already in memory. Reading data blocks from disk is much more inefficient than reading the data blocks from memory. The goal with Oracle should always be to maximize memory utilization.

This test checks the data blocks read from disk per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaPhysicalReads/Seconds where:

- DeltaPhysicalReads: difference in 'select s.value from v$sysstat s, v$statname n where n.name='physical reads' and n.statistic#=s.statistic#' between sample end and start
- Seconds: number of seconds in sample period

User Action

Block reads are inevitable so the aim should be to minimize unnecessary IO. This is best achieved by good application design and efficient execution plans. Changes to execution plans can yield profound changes in performance. Tweaking at system level usually only achieves percentage gains.

To view I/O on a per session basis to determine which sessions are responsible for your physical reads, you should visit the Top Sessions page sorted by Physical Reads. This approach allows you to identify problematic sessions and then drill down to their current SQL statement and perform tuning from there.

To identify the SQL that is responsible for the largest portion of physical reads, visit the Top SQL page sorted by Physical Reads. This page allows you to quickly determine which SQL statements are the causing your I/O activity. From this display you can view the full text of the SQL statement.

The difference between the two methods for identifying problematic SQL is that the Top Sessions view displays sessions that are performing the most physical reads at the moment. The Top SQL view displays the SQL statements that are still in the SQL cache that have performed the most I/O over their lifetime. A SQL statement could show up in the Top SQL view that is not currently being executed.

If the SQL statements are properly tuned and optimized, consider the following suggestions. A larger buffer cache may help - test this by actually increasing DB_BLOCK_BUFFERS. Do not use DB_BLOCK_LRU_EXTENDED_STATISTICS, as this may introduce other performance issues. Never increase the SGA size if it may induce additional paging or swapping on the system.

A less obvious issue which can affect the I/Orates is how well data is clustered physically. For example, assume that you frequently fetch rows from a table where a column is between two values via an index scan. If there are 100 rows in each index block then the two extremes are: 1.Each of the table rows is in a different physical block (100 blocks must be read for each index block). 2.The table rows are all located in the few adjacent blocks (a handful of blocks must be read for each index block).

Pre-sorting or reorganizing data can improve this situation in severe situations as well.

Physical Reads (per transaction)

This metric represents the number of disk reads per transaction during the sample period. When a user performs a SQL query, Oracle tries to retrieve the data from the database buffer cache (memory) first, then goes to disk if it is not in memory already. Reading data blocks from disk is much more expensive than reading the data blocks from memory. The goal with Oracle should always be to maximize memory utilization.

The value of this statistic will be zero if there have not been any write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name will be a better indicator of current performance.

This test checks the data blocks read from disk per transaction. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaReads/Transactions where:

- DeltaReads: difference in 'select value from v$sysstat where name='physical reads'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

Block reads are inevitable so the aim should be to minimize unnecessary IO. This is best achieved by good application design and efficient execution plans. Changes to execution plans can yield orders of magnitude changes in performance. Tweaking at system level usually only achieves percentage gains.

To identify the SQL that is responsible for the largest portion of physical reads, visit the Top SQL page sorted by Physical Reads. This view will allow you to quickly determine which SQL statements are causing the I/O activity. From this display you can view the full text of the SQL statement.

To view I/O on a per session basis to determine which sessions are responsible for your physical reads, you can visit the Top Sessions page sorted by Physical Reads. This approach allows you to identify problematic sessions and then drill down to their current SQL statement to perform tuning.

If the SQL statements are properly tuned and optimized the following suggestions may help. A larger buffer cache may help - test this by actually increasing DB_BLOCK_BUFFERS and not by using DB_BLOCK_LRU_EXTENDED_STATISTICS. Never increase the SGA size if it will induce additional paging or swapping on the system.

A less obvious issue which can affect the I/Orates is how well data is clustered physically. For example, assume that you frequently fetch rows from a table where a column is between two values via an index scan. If there are 100 rows in each index block then the two extremes are: 1. Each of the table rows is in a different physical block (100 blocks must be read for each index block). 2. The table rows are all located in the few adjacent blocks (a handful of blocks must be read for each index block).

Pre-sorting or reorganizing data can help to tackle this in severe situations as well.

Physical Reads Direct (per second)

This metric represents the number of direct physical reads per second.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

physical reads direct/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Physical Reads Direct (per transaction)

This metric represents the number of direct physical reads per transaction.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

physical reads direct/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Physical Reads Direct Lobs (per second)

This metric represents the number of direct large object (LOB) physical reads per second.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

physical reads direct (lob)/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Physical Reads Direct Lobs (per transaction)

This metric represents the number of direct large object (LOB) physical reads per transaction.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived using the following formula:

physical reads direct (lob)/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Physical Writes (per second)

This metric represents the number of disk writes per second during the sample period. This statistic represents the rate of database blocks written from the SGA buffer cached to disk by the DBWR background process, and from the PGA by processes performing direct writes.

This test checks the data blocks written disk per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaWrites/Seconds where:

- DeltaWrites: difference in 'select value from v$sysstat where name='physical writes'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

Because this statistic shows both DBWR writes as well as direct writes by sessions, you should view the physical writes directly to determine where the write activity is actually occurring. If the physical writes direct value comprises a large portion of the writes, then there are probably many sorts or writes to temporary tablespaces occurring.

If the majority of the writes are not direct, they are being performed by the DBWR writes process. This is only be a problem if log writer or redo waits are showing up in the Sessions Waiting for this Event page or the Top Waits page sorted by Time Waited.

Physical Writes (per transaction)

This metric represents the number of disk writes per transaction during the sample period.

The value of this statistic is zero if there have not been any write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name is a better indicator of current performance.

This test checks the data blocks written disk per transaction. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived using the following formula:

DeltaWrites/Transactions where:

- DeltaWrites: difference in 'select value from v$sysstat where name='physical writes'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

Because this statistic shows both DBWR writes as well as direct writes by sessions, you should view the physical writes directly to determine where the write activity is really occurring. If the physical writes direct value comprises a large portion of the writes, then there are likely many sorts or writes to temporary tablespaces that are occurring.

If the majority of the writes are not direct, they are being performed by the DBWR writes process. This will typically only be a problem if log writer or redo waits are showing up in the Sessions Waiting for this Event page or the Top Waits page sorted by Time Waited.

Physical Writes Direct (per second)

This metric represents the number of direct physical writes per second.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

physical writes direct/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central on the Database Home page.

Physical Writes Direct (per transaction)

This metric represents the number of direct physical writes per transaction.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

physical writes direct/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Physical Writes Direct Lobs (per second)

This metric represents the number of direct large object (LOB) physical writes per second.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

physical writes direct (lob)/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Physical Writes Direct Lobs (per transaction)

This metric represents the number of direct large object (LOB) physical writes per transaction.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

physical writes direct (lob)/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Recursive Calls (per second)

This metric represents the number of recursive calls, per second during the sample period.

Sometimes, to execute a SQL statement issued by a user, the Oracle Server must issue additional statements. Such statements are called recursive calls or recursive SQL statements. For example, if you insert a row into a table that does not have enough space to hold that row, the Oracle Server makes recursive calls to allocate the space dynamically if dictionary managed tablespaces are being used. Recursive calls are also generated:

- When data dictionary information is not available in the data dictionary cache and must be retrieved from disk
- In the firing of database triggers
- In the execution of DDL statements
- In the execution of SQL statements within stored procedures, functions, packages and anonymous PL/SQL blocks
- In the enforcement of referential integrity constraints

This test checks the number of recursive SQL calls per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaRecursiveCalls/Seconds where:

- DeltaRecursiveCalls: difference in 'select value from v$sysstat where name='recursive calls'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

If the Oracle Server appears to be making excessive recursive calls while your application is running, determine what activity is causing these recursive calls. If you determine that the recursive calls are caused by dynamic extension, reduce the frequency of extension by allocating larger extents.

Recursive Calls (per transaction)

This metric represents the number of recursive calls, per second during the sample period.

Sometimes, to execute a SQL statement issued by a user, the Oracle Server must issue additional statements. Such statements are called recursive calls or recursive SQL statements. For example, if you insert a row into a table that does not have enough space to hold that row, the Oracle Server makes recursive calls to allocate the space dynamically if dictionary managed tablespaces are being used. Recursive calls are also generated:

- When data dictionary information is not available in the data dictionary cache and must be retrieved from disk
- In the firing of database triggers
- In the execution of DDL statements
- In the execution of SQL statements within stored procedures, functions, packages and anonymous PL/SQL blocks
- In the enforcement of referential integrity constraints

The value of this statistic will be zero if there have not been any write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name will be a better indicator of current performance.

This test checks the number of calls that result in changes to internal tables. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaRecursiveCalls/Transactions where:

- DeltaRecursiveCalls: difference in 'select value from v$sysstat where name='recursive calls'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

If the Oracle Server appears to be making excessive recursive calls while your application is running, determine what activity is causing these recursive calls. If you determine that the recursive calls are caused by dynamic extension, reduce the frequency of extension by allocating larger extents.

Redo Generated (per second)

This metric represents the amount of redo, in bytes, generated per second during this sample period.

The redo log buffer is a circular buffer in the SGA that holds information about changes made to the database. This information is stored in redo entries. Redo entries contain the information necessary to reconstruct, or redo, changes made to the database by INSERT, UPDATE, DELETE, CREATE, ALTER or DROP operations. Redo entries can be used for database recovery if necessary.

This test checks the amount of redo in bytes generated per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaRedoSize/Seconds where:

- DeltaRedoSize: difference in 'select value from v$sysstat where name='redo size'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

The LOG_BUFFER initialization parameter determines the amount of memory that is used when redo entries are buffered to the redo log file.

Consider increasing the LOG_BUFFER initialization parameter to increase the size of the redo log buffer should waiting be a problem. Redo log entries contain a record of the changes that have been made to the database block buffers. The log writer process (LGWR) writes redo log entries from the log buffer to a redo log. The redo log buffer should be sized so space is available in the log buffer for new entries, even when access to the redo log is heavy.

Redo Generated (per transaction)

This metric represents the amount of redo, in bytes, generated per transaction during this sample period.

The redo log buffer is a circular buffer in the SGA that holds information about changes made to the database. This information is stored in redo entries. Redo entries contain the information necessary to reconstruct, or redo, changes made to the database by INSERT, UPDATE, DELETE, CREATE, ALTER or DROP operations. Redo entries are used for database recovery, if necessary.

The value of this statistic is zero if there have been no write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name will be a better indicator of current performance.

This test checks the amount of redo in bytes generated per transaction. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaRedoSize/DeltaTransactions where:

- DeltaRedoSize: difference in 'select value from v$sysstat where name='redo size'' between end and start of sample period
- Transactions: difference in 'select value from v$sysstat where name = 'user commits'' between end and start of sample period

User Action

The LOG_BUFFER initialization parameter determines the amount of memory that is used when buffering redo entries to the redo log file.

Consider increasing the LOG_BUFFER initialization parameter to increase the size of the redo log buffer should waiting be a problem. Redo log entries contain a record of the changes that have been made to the database block buffers. The log writer process (LGWR) writes redo log entries from the log buffer to a redo log. The redo log buffer should be sized so space is available in the log buffer for new entries, even when access to the redo log is heavy.

Redo Writes (per second)

This metric represents the number redo write operations per second during this sample period.

The redo log buffer is a circular buffer in the SGA that holds information about changes made to the database. This information is stored in redo entries. Redo entries contain the information necessary to reconstruct, or redo, changes made to the database by INSERT, UPDATE, DELETE, CREATE, ALTER or DROP operations. Redo entries can be used for database recovery if necessary.

The log writer processes (LGWR) is responsible for redo log buffer management, that is, writing the redo log buffer to a redo log file on disk.

This test checks the number of writes by LGWR to the redo log files per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaRedoWrites/Seconds where:

- DeltaRedoWrites: difference in 'select value from v$sysstat where name='redo writes'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

The LOG_BUFFER initialization parameter determines the amount of memory that is used when redo entries are buffered to the redo log file.

Should waiting be a problem, consider increasing the LOG_BUFFER initialization parameter to increase the size of the redo log buffer. Redo log entries contain a record of the changes that have been made to the database block buffers. The log writer process (LGWR) writes redo log entries from the log buffer to a redo log. The redo log buffer should be sized so space is available in the log buffer for new entries, even when access to the redo log is heavy.

Redo Writes (per transaction)

This metric represents the number of redo write operations per second during this sample period.

The redo log buffer is a circular buffer in the SGA that holds information about changes made to the database. This information is stored in redo entries. Redo entries contain the information necessary to reconstruct, or redo, changes made to the database by INSERT, UPDATE, DELETE, CREATE, ALTER or DROP operations. Redo entries are used for database recovery, if necessary.

The log writer process (LGWR) is responsible for redo log buffer management, that is writing the redo log buffer to a redo log file on disk.

This test checks the number of writes by LGWR to the redo log files per transaction. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaRedoWrites/(DeltaCommits+DeltaRollbacks) where:

- DeltaRedoWrites: difference in 'select s.value from v$sysstat s, v$statname n where n.name='redo writes' and n.statistic#=s.statistic#' between sample end and start
- DeltaCommits: difference in 'select s.value from v$sysstat s, v$statname n where n.name='user commits' and n.statistic#=s.statistic#' between sample end and sample start
- DeltaRollbacks: difference in 'select s.value from v$sysstat s, v$statname n where n.name='user commits' and n.statistic#=s.statistic#' between sample end and sample start

User Action

The LOG_BUFFER initialization parameter determines the amount of memory that is used when buffering redo entries to the redo log file.

Consider increasing the LOG_BUFFER initialization parameter to increase the size of the redo log buffer should waiting be a problem. Redo log entries contain a record of the changes that have been made to the database block buffers. The log writer process (LGWR) writes redo log entries from the log buffer to a redo log. The redo log buffer should be sized so space is available in the log buffer for new entries, even when access to the redo log is heavy.

Rows Processed (per sort)

This metric represents the average number of rows per sort during this sample period.

This test checks the average number of rows per sort during sample period. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

(DeltaSortRows/(DeltaDiskSorts + DeltaMemorySorts)) * 100 where:

- DeltaSortRows: difference in 'select value from v$sysstat where name='sorts (rows)'' between sample end and start
- DeltaMemorySorts: difference in 'select value from v$sysstat where name='sorts (memory)'' between sample end and start
- DeltaDiskSorts: difference in 'select value from v$sysstat where name='sorts (disk)'' between sample end and start

User Action

This statistic displays the average number of rows that are being processed per sort. The size provides information about the sort size of the database. This can help you to determine the SORT_AREA_SIZE appropriately. If the rows per sort are high, you should investigate the sessions and SQL performing the most sorts to see if those SQL statements can be tuned to reduce the size of the sort sample set.

The sessions that are performing the most sorts should be identified, such that the SQL they are executing can be further identified. The sort area sizes for the database may be sized correctly and the application SQL may be performing unwanted or excessive sorts. The sessions performing the most sorts are available through the Top Sessions page sorted by Disk Sorts.

Further drilldown into the session performing the most disk sorts with the Current SQL page displays the SQL statement responsible for the disk sorts.

The Top SQL page sorted by Sorts provides a mechanism to quickly display the SQL statements in the cache presented in sorted order by their number of sort operations. This is an alternative to viewing the sort of current sessions. It allows you to view sort activity via SQL statements and contains cumulative statistics for all executions of that statement.

If excessive sorts are taking place on disk and the queries are correct, consider increasing the SORT_AREA_SIZE initialization parameter to increase the size of the sort area. A larger sort area allows the Oracle Server to keep sorts in memory, reducing the number of I/O operations required to do an equivalent amount of work using the current sort area size.

Scans on Long Tables (per second)

This metric represents the number of long table scans per second during sample period. A table is considered 'long' if the table is not cached and if its high-water mark is greater than 5 blocks.

This test checks the long table scans per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaScans/Seconds where:

- DeltaScans: difference in 'select value from v$sysstat where name='table scans (long tables)'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

A table scan means that the entire table is being scanned record by record in order to satisfy the query. For small tables that can easily be read into and kept in the buffer cache this may be advantageous. But for larger tables this will force a lot of physical reads and potentially push other needed buffers out of the cache. SQL statements with large physical read and logical read counts are candidates for table scans. They can be identified either through the Top SQL page sorted by Physical Reads, or through the Top Sessions page sorted by Physical Reads, with a drilldown to the current SQL for a session.

Scans on Long Tables (per transaction)

This metric represents the number of long table scans per transaction during sample period. A table is considered 'long' if the table is not cached and if its high-water mark is greater than 5 blocks.

The value of this statistic will be zero if there have not been any write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name will be a better indicator of current performance.

This test checks the number of long table scans per transaction. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaScans/Transactions where:

- DeltaScans: difference in 'select value from v$sysstat where name='table scans (long tables)'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

A table scan means that the entire table is being scanned record by record in order to satisfy the query. For small tables that can easily be read into and kept in the buffer cache this may be

advantageous. But for larger tables this will force a lot of physical reads and potentially push other needed buffers out of the cache. SQL statements with large physical read and logical read counts are candidates for table scans. They can be identified either through the Top SQL page sorted by Physical Reads, or through the Top Sessions page sorted by Physical Reads, with a drilldown to the current SQL for a session.

Session Logical Reads (per second)

This metric represents the number of logical reads per second during the sample period. A logical read is a read request for a data block from the SGA. Logical reads may result in a physical read if the requested block does not reside with the buffer cache.

This test checks the logical(db block gets + consistent gets) reads per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

LogicalReads/Seconds where:

- LogicalReads: difference in 'select value from v$sysstat where name='session logical reads'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

Excessive logical reads, even if they do not result in physical reads, can still represent an area that should be considered for performance tuning. Typically large values for this statistic indicate that full table scans are being performed. To identify the SQL that is performing the most logical reads (buffer gets), use the Top SQL page sorted by Buffer Gets. This quickly identifies the SQL responsible for the bulk of the logical reads. You can further investigate these SQL statements via drilldowns. Tuning these SQL statements will reduce your buffer cache access.

Session Logical Reads (per transaction)

This metric represents the number of logical reads per transaction during the sample period.

The value of this statistic is zero if there have not been any write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name will be a better indicator of current performance.

This test checks the logical (db block gets + consistent gets) reads per transaction. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the

number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaReads/Transactions where:

- DeltaReads: difference in 'select value from v$sysstat where name='session logical reads'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

Excessive logical reads, even if they do not result in physical reads, can still represent an area that should be considered for performance tuning. Typically large values for this statistic indicate that full table scans are being performed. To identify the SQL that is performing the most logical reads (buffer gets) use the Top SQL page sorted by Buffer Gets. This quickly identifies the SQL responsible for the bulk of the logical reads.

Soft Parse (%)

A soft parse is recorded when the Oracle Server checks the shared pool for a SQL statement and finds a version of the statement that it can reuse.

This metric represents the percentage of parse requests where the cursor was already in the cursor cache compared to the number of total parses. This ratio provides an indication as to how often the application is parsing statements that already reside in the cache as compared to hard parses of statements that are not in the cache.

This test checks the percentage of soft parse requests to total parse requests. If the value is less than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

((DeltaParseCountTotal - DeltaParseCountHard) / DeltaParseCountTotal) * 100 where:

- DeltaParseCountTotal: difference in 'select value from v$sysstat where name='parse count (total)'' between sample end and start
- DeltaParseCountHard: difference in 'select value from v$sysstat where name='parse count (hard)'' between sample end and start

User Action

Soft parses consume less resources than hard parses, so the larger the value for this item, the better. But many soft parses indicate the application is using SQL inefficiently. Reparsing the statement, even if it is a soft parse, requires a network round trip from the application to the database, as well as requiring the processing time to locate the previously compiled statement in the cache. Reducing network round trips and unnecessary processing will improve application performance.

If this metric value is below 80% you should look at the Top Sessions page sorted by Hard Parses. This page lists the sessions that are currently performing the most hard parses. Starting with these sessions and the SQL statements they are executing will indicate which applications and corresponding SQL statements are being used inefficiently.

If the metric is currently showing a high value, the expensive hard parses are not occurring but the application can still be tuned by reducing the amount of soft parses. Visit the Top SQL page sorted by Parses to identify the SQL statements that have been most parsed. This will allow you to quickly identify SQL that is being re-parsed unnecessarily. You should investigate these statements first for possible application logic changes such that cursors are opened once, and executed or fetched from many times.

Sorts to Disk (per second)

This metric represents the number of sorts going to disk per second for this sample period. For best performance, most sorts should occur in memory, because sorts to disks are expensive to perform. If the sort area is too small, extra sort runs will be required during the sort operation. This increases CPU and I/O resource consumption.

This test checks the number of sorts performed to disk per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaDiskSorts/Seconds where:

- DeltaDiskSorts: difference in 'select value from v$sysstat where name='sorts (disk)'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

The sessions that are performing the most sorts should be identified, such that the SQL they are executing can be further identified. The sort area sizes for the database may be sized correctly, the application SQL may be performing unwanted or excessive sorts. The sessions performing the most sorts are available through the Top Sessions sorted by Disk Sorts page.

Further drilldown into the session performing the most disk sorts with the Current SQL page will show you the SQL statement responsible for the disk sorts.

The Top SQL page sorted by Sorts provides a mechanism to quickly display the SQL statements in the cache, presented in sorted order by their number sort operations. This is an alternative to viewing sort of current sessions, it allows you to view sort activity via SQL statements, and will contain cumulative statistics for all executions of that statement.

If excessive sorts are taking place on disk, and the query's are correct, consider increasing the SORT_AREA_SIZE initialization parameter to increase the size of the sort area. A larger sort area will allow the Oracle Server to keep sorts in memory, reducing the number of I/O operations required to do an equivalent amount of work using the current sort area size.

Sorts to Disk (per transaction)

This metric represents the number of sorts going to disk per transactions for this sample period. For best performance, most sorts should occur in memory, because sorts to disks are expensive to perform. If the sort area is too small, extra sort runs will be required during the sort operation. This increases CPU and I/O resource consumption.

The value of this statistic will be zero if there have not been any write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name will be a better indicator of current performance.

This test checks the number of sorts performed to disk per transaction. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaDiskSorts/Transactions where:

- DeltaDiskSorts: difference in 'select value from v$sysstat where name='sorts (disk)'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

The sessions that are performing the most sorts should be identified, such that the SQL they are executing can be further identified. The sort area sizes for the database may be sized

correctly, the application SQL may be performing unwanted or excessive sorts. The sessions performing the most sorts are available through the Top Sessions page sorted by Disk Sorts.

Further drilldown into the session performing the most disk sorts with the Current SQL page will show you the SQL statement responsible for the disk sorts.

The Top SQL page sorted by Sorts provides a mechanism to quickly display the SQL statements in the cache, presented in sorted order by their number sort operations. This is an alternative to viewing sort of current sessions, it allows you to view sort activity via SQL statements, and will contain cumulative statistics for all executions of that statement.

If excessive sorts are taking place on disk, and the query's are correct, consider increasing the SORT_AREA_SIZE initialization parameter to increase the size of the sort area. A larger sort area will allow the Oracle Server to keep sorts in memory, reducing the number of I/O operations required to do an equivalent amount of work using the current sort area size.

Total Index Scans (per second)

This metric represents the total number of index scans per second.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

index scans kdiixs1/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Total Index Scans (per transaction)

This metric represents the total number of index scans per transaction.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

index scans kdiixsl/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Total Parses (per second)

This number reflects the total number of parses per second, both hard and soft. A hard parse occurs when a SQL statement has to be loaded into the shared pool. In this case, the Oracle Server has to allocate memory in the shared pool and parse the statement. A soft parse is recorded when the Oracle Server checks the shared pool for a SQL statement and finds a version of the statement that it can reuse.

Each time a particular SQL cursor is parsed, this count will increase by one. There are certain operations which will cause a SQL cursor to be parsed. Parsing a SQL statement breaks it down into atomic steps which the optimizer will evaluate when generating an execution plan for the cursor.

This test checks the number of parse calls per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaParses/Seconds where:

- DeltaParses: difference in 'select value from v$sysstat where name='parse count (total)" between end and start of sample period
- Seconds: number of seconds in sample period

User Action

If there appears to be excessive time spent parsing, evaluate SQL statements to determine which can be modified to optimize shared SQL pool memory use and avoid unnecessary statement reparsing. This type of problem is commonly caused when similar SQL statements are written which differ in space, case, or some combination of the two. You may also consider using bind variables rather than explicitly specified constants in your statements whenever possible.

The Top Sessions page sorted by Hard Parses will show you which sessions are incurring the most hard parses. Hard parses happen when the server parses a query and cannot find an exact match for the query in the library cache. Hard parses can be avoided by sharing SQL statements efficiently. The use of bind variables instead of literals in queries is one method to increase sharing.

By showing you which sessions are incurring the most hard parses, this page may lead you to the application or programs that are the best candidates for SQL rewrites.

Also, examine SQL statements which can be modified to optimize shared SQL pool memory use and avoid unnecessary statement reparsing. This type of problem is commonly caused when similar SQL statements are written which differ in space, case, or some combination of the two. You may also consider using bind variables rather than explicitly specified constants in your statements whenever possible.

The SHARED_POOL_SIZE initialization parameter controls the total size of the shared pool. Consider increasing the SHARED_POOL_SIZE to decrease the frequency in which SQL requests are being flushed from the shared pool to make room for new requests.

To take advantage of the additional memory available for shared SQL areas, you may also need to increase the number of cursors permitted per session. You can increase this limit by increasing the value of the initialization parameter OPEN_CURSORS.

Total Parses (per transaction)

This number reflects the total number of parses per transaction, both hard and soft. A hard parse occurs when a SQL statement has to be loaded into the shared pool. In this case, the Oracle Server has to allocate memory in the shared pool and parse the statement. A soft parse is recorded when the Oracle Server checks the shared pool for a SQL statement and finds a version of the statement that it can reuse.

Each time a particular SQL cursor is parsed, this count will increase by one. There are certain operations which will cause a SQL cursor to be parsed. Parsing a SQL statement breaks it down into atomic steps which the optimizer will evaluate when generating an execution plan for the cursor.

This test checks the number of parse calls per transaction. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaParses/Transactions where:

- DeltaParses: difference in 'select value from v$sysstat where name='parse count (total)" between end and start of sample period
- Transactions: number of transactions in sample period

User Action

If there appears to be excessive time spent parsing, evaluate SQL statements to determine which can be modified to optimize shared SQL pool memory use and avoid unnecessary statement reparsing. This type of problem is commonly caused when similar SQL statements are written which differ in space, case, or some combination of the two. You may also consider using bind variables rather than explicitly specified constants in your statements whenever possible.

The Top Sessions page sorted by Hard Parses will show you which sessions are incurring the most hard parses. Hard parses happen when the server parses a query and cannot find an exact match for the query in the library cache. Hard parses can be avoided by sharing SQL statements efficiently. The use of bind variables instead of literals in queries is one method to increase sharing.

By showing you which sessions are incurring the most hard parses, this page may lead you to the application or programs that are the best candidates for SQL rewrites.

Also, examine SQL statements which can be modified to optimize shared SQL pool memory use and avoid unnecessary statement reparsing. This type of problem is commonly caused when similar SQL statements are written which differ in space, case, or some combination of the two. You may also consider using bind variables rather than explicitly specified constants in your statements whenever possible.

The SHARED_POOL_SIZE initialization parameter controls the total size of the shared pool. Consider increasing the SHARED_POOL_SIZE to decrease the frequency in which SQL requests are being flushed from the shared pool to make room for new requests.

To take advantage of the additional memory available for shared SQL areas, you may also need to increase the number of cursors permitted per session. You can increase this limit by increasing the value of the initialization parameter OPEN_CURSORS.

Total Table Scans (per second)

This metric represents the number of long and short table scans per second during the sample period. A table is considered 'long' if the table is not cached and if its high-water mark is greater than 5 blocks.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every10 Minutes	Not Defined	Not Defined	Not Defined

Data Source

The data is derived from the following formula:

(DeltaLongScans + DeltaShortScans)/Seconds:

- DeltaLongScans: difference in 'select value from v$sysstat where name='table scans (long tables)'' between end and start of sample period
- DeltaShortScans: difference in 'select value from v$sysstat where name='table scans (short tables)'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

A table scan indicates that the entire table is being scanned record-by-record in order to satisfy the query. For small tables that can easily be read into and kept in the buffer cache, this may be advantageous. But larger tables will force many physical reads and potentially push other required buffers out of the cache. SQL statements with large physical read and logical read counts are candidates for table scans. They can be identified through two different methods. The Top Sessions page sorted by Physical Reads displays sessions that are responsible for the current I/O activity. The Top SQL page sorted by Physical Reads lists the SQL statements

in the cache by the amount of I/O they have performed. Some of these SQL statements may have high I/O numbers but they may not be attributing to the current I/O load.

Total Table Scans (per transaction)

This metric represents the number of long and short table scans per transaction during the sample period. A table is considered 'long' if the table is not cached and if its high-water mark is greater than 5 blocks.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	Not Defined

Data Source

The data is derived from the following formula:

(DeltaLongScans + DeltaShortScans)/Transactions:

- DeltaLongScans: difference in 'select value from v$sysstat where name='table scans (long tables)'' between end and start of sample period
- DeltaShortScans: difference in 'select value from v$sysstat where name='table scans (short tables)'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

A table scan indicates that the entire table is being scanned record-by-record in order to satisfy the query. For small tables that can easily be read into and kept in the buffer cache, this may be advantageous. But larger tables will force many physical reads and potentially push other required buffers out of the cache. SQL statements with large physical read and logical read counts are candidates for table scans. They can be identified through two different methods. The Top Sessions page sorted by Physical Reads displays sessions that are responsible for the current I/O activity. The Top SQL page sorted by Physical Reads lists the SQL statements in the cache by the amount of I/O they have performed. Some of these SQL statements may have high I/O numbers but they may not be attributing to the current I/O load.

User Calls (%)

This metric represents the percentage of user calls to recursive calls.

Occasionally, to execute a SQL statement issued by a user, the Oracle Server must issue additional statements. Such statements are called recursive calls or recursive SQL statements. For example, if you insert a row into a table that does not have enough space to hold that row, the Oracle Server makes recursive calls to allocate the space dynamically if dictionary managed tablespaces are being used. Recursive calls are also generated:

When data dictionary information is not available in the data dictionary cache and must be retrieved from disk.

- In the firing of database triggers
- In the execution of DDL statements

- In the execution of SQL statements within stored procedures, functions, packages and anonymous PL/SQL blocks
- In the enforcement of referential integrity constraints

This test checks the percentage of user calls to recursive calls. If the value is less than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

(DeltaUserCalls/(DeltaRecursiveCalls + DeltaUserCalls)) * 100 where:

- DeltaRecursiveCalls: difference in 'select value from v$sysstat where name='recursive calls'' between sample end and start
- DeltaUserCalls: difference in 'select value from v$sysstat where name='user calls'' between sample end and start

User Action

A low value for this metric means that the Oracle Server is making a large number of recursive calls. If the Oracle Server appears to be making excessive recursive calls while your application is running, determine what activity is causing these recursive calls. If you determine that the recursive calls are caused by dynamic extension, reduce the frequency of extension by allocating larger extents.

User Calls (per second)

This metric represents the number of logins, parses, or execute calls per second during the sample period.

This test checks the number of logins, parses, or execute calls. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaUserCalls/Seconds where:

- DeltaUserCalls: difference in 'select value from v$sysstat where name='user calls'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

This statistic is a reflection of how much activity is going on within the database. Spikes in the total user call rate should be investigated to determine which of the underlying calls is actually increasing. Parse, execute and logon calls each signify different types of user or application actions and should be addressed individually. User Calls is an overall activity level monitor.

User Calls (per transaction)

This metric represents the number of logins, parses, or execute calls per transaction during the sample period.

The value of this statistic will be zero if there have not been any write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name will be a better indicator of current performance.

This test checks the number of logins, parses, or execute calls per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaUserCalls/Transactions where:

- DeltaUserCalls: difference in 'select value from v$sysstat where name='user calls'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

This statistic is a reflection of how much activity is going on within the database. Spikes in the total user call rate should be investigated to determine which of the underlying calls is actually increasing. Parse, execute and logon calls each signify different types of user or application actions and should be addressed individually. User Calls is an overall activity level monitor.

User Commits (per second)

This metric represents the number of user commits performed, per second during the sample period. When a user commits a transaction, the redo generated that reflects the changes made to database blocks must be written to disk. Commits often represent the closest thing to a user transaction rate.

This test checks the number of user commits per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaCommits/Seconds where:

- DeltaCommits: difference in 'select value from v$sysstat where name='user commits'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

This statistic is an indication of how much work is being accomplished within the database. A spike in the transaction rate may not necessarily be bad. If response times stay close to normal, it means your system can handle the added load. Actually, a drop in transaction rates and an increase in response time may be indicators of problems. Depending upon the application, transaction loads may vary widely across different times of the day.

User Commits (per transaction)

This metric represents the number of user commits performed, per transaction during the sample period. When a user commits a transaction, the redo generated that reflects the changes made to database blocks must be written to disk. Commits often represent the closest thing to a user transaction rate.

The value of this statistic will be zero if there have not been any write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name will be a better indicator of current performance.

This test checks the number of user commits per transaction. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaCommits/Transactions where:

- DeltaCommits: difference in 'select value from v$sysstat where name='user commits'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

This statistic is an indication of how much work is being accomplished within the database. A spike in the transaction rate may not necessarily be bad. If response times stay close to normal, it means your system can handle the added load. Actually, a drop in transaction rates and an increase in response time may be indicators of problems. Depending upon the application, transaction loads may vary widely across different times of the day.

User Rollback Undo Records Applied (per second)

This metric represents the number of undo records applied to user-requested rollback changes per second.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

(rollback changes - undo records applied)/time

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

User Rollback Undo Records Applied (per transaction)

This metric represents the number of undo records applied to user-requested rollback changes per transaction.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

(rollback changes - undo records applied)/transactions

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

User Rollbacks (per second)

This metric represents the number of times, per second during the sample period, that users manually issue the ROLLBACK statement or an error occurred during a user's transactions.

This test checks the number of rollbacks per second. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaRollbacks/Seconds where:

- DeltaRollbacks: difference in 'select value from v$sysstat where name='user rollbacks'' between end and start of sample period
- Seconds: number of seconds in sample period

User Action

This value shows how often users are issuing the ROLLBACK statement or encountering errors in their transactions. Further investigation should be made to determine if the rollbacks are part of some faulty application logic or due to errors occurring through database access.

User Rollbacks (per transaction)

This metric represents the number of times, per transaction during the sample period, that users manually issue the ROLLBACK statement or an error occurred during a user's transactions.

The value of this statistic will be zero if there have not been any write or update transactions committed or rolled back during the last sample period. If the bulk of the activity to the database is read only, the corresponding per second metric of the same name will be a better indicator of current performance.

This test checks the Number of rollbacks per transaction. If the value is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived from the following formula:

DeltaRollbacks/Transactions where:

- DeltaRollbacks: difference in 'select value from v$sysstat where name='user rollbacks'' between end and start of sample period
- Transactions: number of transactions in sample period

User Action

This value shows how often users are issuing the ROLLBACK statement or encountering errors in their transactions. Further investigation should be made to determine if the rollbacks are part of some faulty application logic or due to errors occurring through database access.

Top Wait Events

This section provides information on the metrics in the Top Wait Events category.

Target Version	Evaluation and Collection Frequency
All versions	Every 15 minutes

Metric Name	Description
Average Foreground Wait Time (millisecond)	The average foreground wait time, in milliseconds.
Average Wait Time (millisecond)	The average wait time, in milliseconds.

Metric Name	Description
Total Foreground Wait Time (second)	The total foreground wait time, in seconds.
Total Number of Foreground Waits	The total number of foreground waits.
Total Number of Waits	The total number of waits.
Total Wait Time (second)	The total wait time, in seconds.
Wait Class Name	The name of the wait class.

Total Objects by Schema

The metrics in this category contain the metric that provides the number of database objects in a schema.

Total Object Count

This metric displays the total number of database objects in a schema.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	%value% object(s) exist in the %owner% schema.

Total Tables by Schema

The metrics in this category contain the metric that provides the number of tables in a schema.

Total Table Count

This metric displays the total number of tables in a schema.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	%value% table(s) exist in the %owner% schema.

Unusable Indexes

This metric category represents the number of unusable indexes in the database.

Unusable Index Count

This metric represents the total unusable index count in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 hours	Not Defined	Not Defined	Unusable Index Count in the database is %value%

Data Source

The data is derived from the dba_indexes, dba_ind_partitions, and dba_ind_subpartitions views.

User Action

The “Rebuild Unusable Indexes” corrective action could be setup against the incident to automatically attempt to rebuild the unusable indexes in the database. This lets the user to specify various rebuild options and the schemas in which the indexes should be rebuilt. In addition, the incident also enables the user to rebuild the unusable indexes from the Enterprise Manager console.

Unusable Indexes by Schema

This metric category represents the number of unusable indexes in each schema.

Unusable Index Count by Schema

This metric represents the total number of unusable indexes per schema.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 hours	Not Defined	Not Defined	Unusable Index Count in %Unusable_Index_owner% schema is %value%

Multiple Thresholds

Different warning and critical threshold values could be set for each Unusable Index Owner (schema) object.

If warning or critical threshold values are currently set for any Unusable Index Owner object, those thresholds could be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Unusable Index Owner object, use the Edit Thresholds page.

Data Source

The data is derived from the dba_indexes, dba_ind_partitions, and dba_ind_subpartitions views.

User Action

The “Rebuild Unusable Indexes” corrective action could be setup against the incident to automatically attempt to rebuild the unusable indexes in each Unusable Index Owner (schema) object. This lets the user to specify various rebuild options that should be used for the

operation. In addition, the incident also enables the user to rebuild the unusable indexes from the Enterprise Manager console.

User Audit

The metrics in this category contain the metrics used to represent logons to the database by audited users (such as SYS).

Audited User

This metric monitors specified database user connections. For example, an alert is displayed when a particular database user connection, specified by the User name filter argument, has been detected.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	SYS	-	User %value% logged on from %machine%.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Username_Machine object.

If warning or critical threshold values are currently set for any Username_Machine object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Username_Machine object, use the Edit Thresholds page.

Data Source

The following command is the data source for each metric index:

```
SELECT uname, mname, TO_CHAR(count(uname)) , concat(concat(uname,'_'), mname)
username_machine FROM
            (SELECT TRIM(TRANSLATE(username,CHR(0),' ')) uname,
TRIM(TRANSLATE(machine,CHR(0),' ')) mname
            FROM v$session where type != 'BACKGROUND' and lower(program) not like
'rman%' and username is not null )
            GROUP by uname, mname
```

User Action

User actions may vary depending on the user connection that is detected.

Audited User - Host

This metric represents the host system from which the audited user's login originated.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The following command is the data source for each metric index:

```
SELECT uname, mname, TO_CHAR(count(uname)) , concat(concat(uname,'_'), mname)
username_machine FROM
       (SELECT TRIM(TRANSLATE(username,CHR(0),' ')) uname,
TRIM(TRANSLATE(machine,CHR(0),' ')) mname
       FROM v$session where type != 'BACKGROUND' and lower(program) not like 'rman%' and
username is not null )
       GROUP by uname, mname
```

User Action

Review the access to the database from this client machine.

Audited User Session Count

This metric represents the number of logons the audited user has from a given machine.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The following command is the data source for each metric index:

```
SELECT uname, mname, TO_CHAR(count(uname)) , concat(concat(uname,'_'), mname)
username_machine from
         (SELECT TRIM(TRANSLATE(username,CHR(0),' ')) uname,
TRIM(TRANSLATE(machine,CHR(0),' ')) mname
          FROM v$session where type != 'BACKGROUND' and lower(program) not like 'rman%'
and username is not null )
          GROUP by uname, mname
```

User Action

No user action is necessary.

User Block

The metrics in this category contain the metrics that tell to what extent, and how consistently, a given session is blocking multiple other sessions.

Blocking Session Count

This metric signifies that a database user is blocking at least one other user from performing an action, such as updating a table. An alert is generated if the number of consecutive blocking occurrences reaches the specified value.

> **Note:**
>
> - The catblock.sql script needs to be run on the managed database prior to using the User Blocks test. This script creates some additional tables, view, and public synonyms that are required by the User Blocks test.
> - Unlike most metrics, which accept thresholds as real numbers, this metric can only accept an integer as a threshold.

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	Not Defined	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Blocking Session ID object.

If warning or critical threshold values are currently set for any Blocking Session ID object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Blocking Session ID object, use the Edit Thresholds page.

Data Source

The data is derived from the following formula:

```
SELECT SUM(num_blocked)
  FROM (SELECT id1, id2, MAX(DECODE(block, 1, sid, 0)) blocking_sid,
    SUM(DECODE(request, 0, 0, 1)) num_blocked
    FROM v$lock
      WHERE block = 1 OR request>0
      GROUP BY id1, id2)
  GROUP BY blocking SID
```

User Action

Either have user who is blocking other users rollback the transaction, or wait until the blocking transaction has been committed.

User Block Chain

The metrics in this category collect information on lock chains, including DB time currently accumulated per chain and the blocked sessions for each chain.

Blocking Session Count

This metric represents the total number of sessions blocked in this chain.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The data is derived from the v$lock and v$session views.

User Action

No user action is required.

Blocking Session DB Time

This metric represents the total DB time currently accumulated in this chain.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	Total db time %value% seconds is consumed by %count% sessions blocked by session (%blocker_session_info%).

Data Source

The data is derived from the v$lock and v$session views.

User Action

No user action is required.

User Locks

The metrics in this category provide information regarding user locks.

Enterprise Manager will issue the alert when the Maximum Blocked Session Count or maximum blocked DB time (seconds) of transactional locks: TM, TX, UL reach the threshold.

Maximum Blocked DB Time (seconds)

This metric represents the maximum time wasted in any given lock chain, not for the total time wasted by everyone in any lock chain.

Target Version	Key	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	lockType: "TM"	Every 10 Minutes	Not Defined	Not Defined	%value% seconds in DB Time is spent waiting for %lockType% lock.
All versions	lockType: "TX"	Every 10 Minutes	Not Defined	Not Defined	%value% seconds in DB Time is spent waiting for %lockType% lock.
All versions	lockType: "UL"	Every 10 Minutes	Not Defined	Not Defined	%value% seconds in DB Time is spent waiting for %lockType% lock.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each User Lock Type object.

If warning or critical threshold values are currently set for any User Lock Type object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each User Lock Type object, use the Edit Thresholds page.

Data Source

The data for the metric is retrieved from database view gv$session.

User Action

User can set the threshold for warning alert or critical alert for maximum Blocked DB Time (seconds). When maximum time wasted in any given lock chain reaches the threshold, Enterprise Manager will issue the alert.

Maximum Blocked Session Count

This metric represents the maximum length of any lock chain, not for the total number of people stuck in lock chains.

Target Version	Key	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	lockType: "TM"	Every 10 Minutes	Not Defined	Not Defined	%value% sessions are blocked by %lockType% lock.
All versions	lockType: "TX"	Every 10 Minutes	Not Defined	Not Defined	%value% sessions are blocked by %lockType% lock.
All versions	lockType: "UL"	Every 10 Minutes	Not Defined	Not Defined	%value% sessions are blocked by %lockType% lock.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each User Lock Type object.

If warning or critical threshold values are currently set for any User Lock Type object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each User Lock Type object, use the Edit Thresholds page.

Data Source

The data for the metric is retrieved from database view gv$session.

User Action

User can set the threshold for warning alert or critical alert for Maximum Blocked Session Count. When maximum length of any lock chain reaches the threshold, Enterprise Manager will issue the alert.

User-Defined SQL

The metrics in this category enable you to execute your own SQL statements. The data returned by these SQL statements can be compared against thresholds and generate severity alerts similar to alerts in predefined metrics.

User-Defined Numeric Metric

This metric contains a value if the value type is NUMBER. Otherwise, the value is "", if the value is STRING.

Data Source

The data source is a SQL statement which can be either a Select statement or function that returns a single scalar value (numeric or string).

User-Defined String Metric

Contains a value if the value type is STRING. Otherwise, the value is "", if the value is NUMBER.

Data Source

The data source is a SQL statement which can be either a Select statement or function that returns a single scalar value (numeric or string).

Wait Bottlenecks

This metric category contains the metrics that approximate the percentage of time spent waiting by user sessions. This approximation takes system-wide totals and discounts the effects of sessions belonging to background processes.

Active Sessions Using CPU

This metric represents the active sessions using CPU.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Active Sessions Waiting: I/O

This metric represents the active sessions waiting for I/O.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Active Sessions Waiting: Other

This metric represents all the waits that are neither idle nor user I/O.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Average Instance CPU (%)

This metric represents the percentage of average CPU used.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Host CPU Utilization (%)

This metric represents the percentage of CPU being used across hosts.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Wait Time (%)

This metric represents the percentage of time spent waiting, instance-wide, for resources or objects during this sample period.

This test checks the percentage time spent waiting, instance-wide, for resources or objects during this sample period. If the % Wait Time is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Table 1-8 Metric Summary Table

Target Version	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Data Source

The data is derived using the following formula:

DeltaTotalWait / (DeltaTotalWait + DeltaCpuTime) where:

- DeltaTotalWait: difference of 'sum of time waited for all wait events in v$system_event' between sample end and start
- DeltaCpuTime: difference of 'select value from v$sysstat where name='CPU used by this session' between sample end and start

User Action

Investigate further into which specific wait events are responsible for the bulk of the wait time. Individual wait events may identify unique problems within the database. Diagnosis will be tailored where appropriate through drilldowns specific to individual wait events.

Waits by Wait Class

This metric category contains the waits by wait class metrics.

Average Users Waiting Count

This metric represents the average number of users that have made a call to the database and that are waiting for an event, such as an I/O or a lock request, to complete. If the number of users waiting on events increases, it indicates that either more users are running, increasing workload, or that waits are taking longer, for example when maximum I/O capacity is reached and I/O times increase.

Table 1-9 Metric Summary Table

Target Version	Key	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	class: "Administrative"	Every 10 Minutes	Not Defined	Not Defined	Not Defined
All versions	class: "Application"	Every 10 Minutes	Not Defined	Not Defined	Not Defined
All versions	class: "Cluster"	Every 10 Minutes	Not Defined	Not Defined	Not Defined
All versions	class: "Commit"	Every 10 Minutes	Not Defined	Not Defined	Not Defined
All versions	class: "Concurrency"	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "Configuration"	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "Network"	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "Other"	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "Scheduler"	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

Table 1-9 (Cont.) Metric Summary Table

Target Version	Key	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	class: "System I/O"	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "User I/O"	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

[1] For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Wait Class object.

If warning or critical threshold values are currently set for any Wait Class object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Wait Class object, use the Edit Thresholds page.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Database Time Spent Waiting (%)

This metric represents the percentage of time that database calls spent waiting for an event. Although there is no correct value for this metric, it can be used to detect a change in the operation of a system, for example, an increase in Database Time Spent Waiting from 50% to 75%. ('No correct value' means that there is no single value that can be applied to any database. The value is a characteristic of the system and the applications running on the system.)

Table 1-10 Metric Summary Table

Target Version	Key	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	class: "Administrative"	Every Minute	Every 10 Minutes	30	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "Application"	Every Minute	Every 10 Minutes	30	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "Cluster"	Every Minute	Every 10 Minutes	50	Not Defined	The Management Agent generates the alert text.[1]

Table 1-10 (Cont.) Metric Summary Table

Target Version	Key	Server Evaluation Frequency	Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	class: "Commit"	Every Minute	Every 10 Minutes	50	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "Concurrency"	Every Minute	Every 10 Minutes	30	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "Configuration"	Every Minute	Every 10 Minutes	30	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "Network"	Every Minute	Every 10 Minutes	30	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "Other"	Every Minute	Every 10 Minutes	30	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "Scheduler"	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "System I/O"	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]
All versions	class: "User I/O"	Every Minute	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Wait Class object.

If warning or critical threshold values are currently set for any Wait Class object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Wait Class object, use the Edit Thresholds page.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page. ADDM will highlight the source of increased time spent in wait events.

2
Cluster Database

This chapter provides information about the Cluster Database (Oracle Real Application Clusters (RAC) database) metrics.
The metric information includes some or all of the following: metric name, description, target version, default collection frequency, default warning threshold, default critical threshold, and alert text.

Archive Area

This metric category contains metrics that track the utilization of the database archived redo log destinations.

If the database is running in ARCHIVELOG mode, these metrics examine the space utilization in the database archived redo log destinations (as specified in the LOG_ARCHIVED_DEST_n initialization parameters). If the database is not running in ARCHIVELOG mode, these metrics are not applicable and the collections do not run. For each archived redo log destination, this metric category returns the total, used, and free space. The methodology used to collect this information is different depending on whether the destinations are configured to use a conventional filesystem or an ASM diskgroup.

> **Note:**
>
> For databases that are configured to archive to the Fast Recovery Area, the Archive Area metrics (Archive Area Used (%), Archive Area Used (KB), Free Archive Area (KB), and Total Archive Area (KB)) are not applicable. Instead, use the Recovery Area Free Space (%) metric to monitor Fast Recovery Area usage.

The formulas used to calculate all the metrics in this metric group depend on the following conditions:

- Whether there is a quota configured in the LOG_ARCHIVE_DEST_n parameter setting.
- Whether the archived redo log destination is configured to use an ASM diskgroup or a regular filesystem location.

Applying these conditions yields three possible archive area scenarios that must be accommodated by these metrics:

- Quota is set
- No quota set
  - Archive area on regular filesystem
  - Archive area on ASM diskgroup

Archive Area Used (%)

The Archive Area Used (%) metric returns the percentage of space used in the archived redo log destination. If the space used is more than the threshold value given in the threshold arguments, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 15 Minutes	80	Not Defined	%value%%% of archive area %archDir% is used.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Archive Area Destination object.

If warning or critical threshold values are currently set for any Archive Area Destination object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Archive Area Destination object, use the Edit Thresholds page.

Data Source

The formula used for each of the three archive area scenarios is as follows:

- Quota is set: Regardless of whether the destination is using ASM or a conventional filesystem location, if the QUOTA_SIZE attribute is specified in the associated LOG_ARCHIVE_DEST_n parameter (meaning there is a quota specified for the destination), the percentage is calculated using the following formula:

```
Archive Area Used (%) = (QUOTA_USED/QUOTA_SIZE) * 100
```

- No quota is set:
  - Archive area on regular filesystem: Free and used space in the archive area is determined by running the UNIX `df -k` command against the filesystem on which the archive area resides.
  - Archive area on ASM diskgroup: The space used in the archive area is calculated by first determining the total diskgroup size (minus space required for redundancy management) and dividing that by the diskgroup redundancy factor (1 for External, 2 for Normal, 3 for High) to arrive at the "Total Safely Usable" space. Then, the "Safely Usable Free" space is determined, which is the free space that can be safely utilized taking mirroring and redundancy needs into account. The SQL used to determine these values is as follows:

```
select (((NVL(dg.total_mb,0) -
NVL(dg.required_mirror_free_mb,0))*1024)/
decode(dg.type,'EXTERN',1,'NORMAL',2,'HIGH',3,1))
TotalSafelyUsable,NVL(dg.usable_file_mb,0)*1024 SafelyUsableFree from
V$ASM_DISKGROUP_STAT dg
where state in ('CONNECTED', 'MOUNTED') and name='$diskGroup'";
```

Using the values from this query, the Archive Area Used (%) is calculated as follows:

```
Archive Area Used (%) = [(TotalSafelyUsable - SafelyUsableFree)/
TotalSafelyUsable] * 100
```

User Action

Verify that the database archived redo log destination parameters are configured properly. For more information, see Specifying Archive Destinations in *Oracle Database Administrator's Guide*.

Archive Area Used (KB)

This metric returns the total space used (in KB) on the device containing the archived redo log destination directory.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The formula used for each of the three archive area scenarios is as follows:

- Quota is set: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula:

  ```
  Archive Area Used (KB) = QUOTA_USED
  ```

- No quota is set:
  - Archive area on regular filesystem: Same underlying methodology as described above for the Archive Area Used (%) metric.
  - Archive area on ASM diskgroup: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula (referencing the values from the SQL query in the Archive Area Used (%) metric):

    ```
    Archive Area Used (KB) = TotalSafelyUsable - SafelyUsableFree
    ```

User Action

Verify that the database archived redo log destination parameters are configured properly. For more information, see Specifying Archive Destinations in *Oracle Database Administrator's Guide*.

Free Archive Area (KB)

This metric returns the free space (in KB) on the device containing the archived redo log destination directory.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 Minutes	Not Defined	Not Defined	Archive area %archDir% has %value% free KB remaining.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Archive Area Destination object.

If warning or critical threshold values are currently set for any Archive Area Destination object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Archive Area Destination object, use the Edit Thresholds page.

Data Source

The formula used for each of the three archive area scenarios is as follows:

- Quota is set: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula:

```
Free Archive Area (KB) = QUOTA_SIZE - QUOTA_USED
```

- No quota is set:
  - Archive area on regular filesystem: Same underlying methodology as described above for the Archive Area Used (%) metric.
  - Archive area on ASM diskgroup: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula (referencing the values from the SQL query in the Archive Area Used (%) metric):

```
Free Archive Area (KB) = SafelyUsableFree
```

User Action

Verify that the database archived redo log destination parameters are configured properly. For more information, see Specifying Archive Destinations in *Oracle Database Administrator's Guide*.

Total Archive Area (KB)

This metric returns the total space (in KB) on the device containing the archived redo log destination directory.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Data Source

The formula used for each of the three archive area scenarios is as follows:

- Quota is set: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula:

```
Total Archive Area (KB) = QUOTA_SIZE
```

- No quota is set:
  - Archive area on regular filesystem: Same underlying methodology as described above for the Archive Area Used (%) metric.
  - Archive area on ASM diskgroup: Same underlying methodology as described above for the Archive Area Used (%) metric, but using the following formula (referencing the values from the SQL query in the Archive Area Used (%) metric):

```
Total Archive Area (KB) = TotalSafelyUsable
```

User Action

Verify that the database archived redo log destination parameters are configured properly. For more information, see Specifying Archive Destinations in *Oracle Database Administrator's Guide*.

Availability Notifications (Server Generated Alert)

This section provides information on the metrics in the Availability Notifications (Server Generated Alert) category.

Target Version	Evaluation and Collection Frequency
All versions	N/A

Metric Name	Description
Database Down	Notifies when a database is down.

Data Guard Fast-Start Failover

This section provides information on the metrics in the Data Guard Fast-Start Failover category, which generates an alert to notify of a new primary database after a fast-start failover occurred.

Target Version	Evaluation and Collection Frequency
All versions	Every 5 minutes

Metric Name	Description
Fast-Start Failover Occurred	Indicates whether a fast-start failover occurred in the last 15 minutes.
Last Fast-Start Failover Reason	The reason why the fast-start failover occurred.
Last Fast-Start Failover Time	The timestamp of the last fast-start failover.

Data Source

The data source for this metric is the v$fs_failover_stats view.

Data Guard Fast-Start Failover Observer – Oracle Database 11gR2 to 18c

The metrics in this category monitor the status of a fast-start failover observer in the Data Guard configuration.

Observer Status

This metric generates a critical alert on the primary database if the fast-start failover (FSFO) configuration is in an unobserved condition, indicating that FSFO is not currently possible.

Table 2-1 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR2, 11gR202, 12c, 12cR102, 12cR2, 18c	Every 1 minute	Not Defined	Error	The Data Guard fast-start failover observer status is %value%.

User Action

If the Data Guard configuration was configured in Enterprise Manager to use the automatic Observer restart feature, the alert will clear after a new observer process is restarted. Otherwise, determine the cause of the unobserved condition, and restart the Observer process if necessary.

Data Guard Fast-Start Failover Observers – Oracle Database 19c and later

This section provides information on the metrics in the Data Guard Fast-Start Failover Observers category for Oracle Database 19c and later. These metrics monitor the status of all the fast-start failover observers in the Data Guard configuration.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
19c and later	Every 5 minutes	Not Defined	Error	The Data Guard fast-start failover observer status is %value%.

Metric Name	Description	Data Source
Is Master Observer	Indicates whether the observer is the master observer (YES or NO).	v$fs_failover_observers

Metric Name	Description	Data Source
Is Observer Registered	Indicates whether the observer is registered (YES or NO).	v$fs_failover_observers
Observer Host	The server on which the observer is running.	v$fs_failover_observers
Observer Log File	The observer log file. **Note**: This metric is only available for Oracle Database 21c. For Oracle Database 19c, this metric column is empty.	v$fs_failover_observers
Observer Name	The name of the observer.	v$fs_failover_observers
Observer State File	The observer state file. **Note**: This metric is only available for Oracle Database 21c. For Oracle Database 19c, this metric column is empty.	v$fs_failover_observers
Observer Status	The status of the observer. **Note**: This metric is obtained from the **Observer Host**, **Pinging Active Target**, and **Pinging Primary** metrics.	v$fs_failover_observers
Pinging Active Target	Indicates whether the observer is currently connected to the active target (YES or NO).	v$fs_failover_observers
Pinging Primary	Indicates whether the observer is currently connected to the primary database (YES or NO).	v$fs_failover_observers

Data Guard Performance

This metric category provides the Data Guard Performance metric.

Apply Lag (seconds)

This metric displays (in seconds) how far the standby is behind the primary.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	The standby database is approximately %value% seconds behind the primary database.

Data Source

The data source for this metric is the following command:

v$dataguard_stats('apply lag')

Estimated Failover Time (seconds)

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

This metric shows the approximate number of seconds required to failover to this standby database. This accounts for the startup time, if necessary, plus the remaining time required to

apply all the available redo on the standby. If a bounce is not required, it is only the remaining apply time.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	The estimated time to failover is approximately %value% seconds.

Data Source

The data source for this metric is the following command:

v$dataguard_stats ('estimated startup time','apply finish time','standby has been open')

Redo Apply Rate (KB/second)

Displays the Redo Apply Rate in KB/second on this standby.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	The estimated time to failover is approximately %value% seconds.

Redo Generation Rate (KB/second)

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	The redo generation rate is %value% KB/sec.

Transport Lag (seconds)

The approximate number of seconds of redo not yet available on this standby database. This may be because the redo has not yet been shipped or there may be a gap.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Not Defined	Not Defined	There are approximately %value% seconds of redo not yet available on this standby database.

Data Source

The data source for this metric is the following command:

v$dataguard_stats('transport lag')

Transport Lag Data Refresh Time

Transport Lag metrics are computed based on data that is periodically received from the primary database. An unchanging value in this column across multiple queries indicates that the standby database is not receiving data from the primary database.

Data Source

DATUM_TIME in v$dataguard_stats

Data Guard Status

The metrics in the Data Guard metrics category check the status, data not received, and data not applied for the databases in the Data Guard configuration.

Data Guard Status

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Use the Data Guard Status metric to check the status of each database in the Data Guard configuration.

By default, a critical and warning threshold value was set for this metric column. Alerts will be generated when threshold values are reached. You can edit the value for a threshold as required.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	Warning	Error	The Data Guard status of %dg_name% is %value%.

Data Source

1. Check the Edit Properties General page for the primary and standby databases for detailed information.
2. Examine the database alert logs and the Data Guard broker logs for additional information.

Database Cardinality

This metric category contains the metrics that monitor the number of active instances of a cluster database.

Open Instance Count

This metric monitors how many instances are in an open state.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every 5 Minutes	Not Defined	Not Defined	%value% instance(s) out of %total_count% are up.

Database Job Status

This metric category contains the metrics that represent the health of database jobs registered through the DBMS_SCHEDULER interface.

Broken Job Count

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

The Oracle Server job queue is a database table that stores information about local jobs such as the PL/SQL call to execute for a job such as when to run a job. Database replication is also managed by using the Oracle job queue mechanism using jobs to push deferred transactions to remote master sites, to purge applied transactions from the deferred transaction queue, or to refresh snapshot refresh groups.

A job can be broken in two ways:

Oracle has failed to successfully execute the job after sixteen attempts. The job has been explicitly marked as broken by using the procedure DBMS_ JOB.BROKEN.

This metric checks for broken DBMS jobs. A critical alert is generated if the number of broken jobs exceeds the value specified by the threshold argument.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every 30 Minutes	0	Not Defined	%value% job(s) are broken.

Data Source

The data source for this metric is the following command:

```
SELECT COUNT(*)
  FROM dba_jobs
  WHERE broken < > 'N'
```

User Action

Check the ALERT log and trace files for error information. Correct the problem that is preventing the job from running. Force immediate re-execution of the job by calling DBMS_SCHEDULER.RUN.

Failed Job Count

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

The Oracle Server job queue is a database table that stores information about local jobs such as the PL/SQL call to execute for a job such as when to run a job. Database replication is also

managed by using the Oracle job queue mechanism using jobs to push deferred transactions to remote master sites, to purge applied transactions from the deferred transaction queue or to refresh snapshot refresh groups.

If a job returns an error while Oracle is attempting to execute it, the job fails. Oracle repeatedly tries to execute the job doubling the interval of each attempt. If the job fails sixteen times, Oracle automatically marks the job as broken and no longer tries to execute it.

This metric checks for failed DBMS jobs. An alert is generated if the number of failed job exceeds the value specified by the threshold argument.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every 30 Minutes	0	Not Defined	%value% job(s) have failed.

Data Source

The data source for this metric is the following command:

```
SELECT COUNT(*)
  FROM dba_jobs
  WHERE NVL(failures, 0) < >  0"
```

User Action

Check the ALERT log and trace files for error information. Correct the problem that is preventing the job from running.

Database Scheduler Jobs

This section provides information on the metrics in the Database Scheduler Jobs category, which report the current status of DBMS jobs registered through the DBMS_SCHEDULER interface. Using these metrics, you can monitor long running jobs and obtain alerts on individual jobs.

Elapsed Running Time (in Minutes)

The duration of time the current DBMS job has been running, in minutes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	> 5 Minutes	> 30 Minutes	DBMS job %job_name% for %owner% has been running for %value% minutes.

Failure Count

The number of times the DBMS job has failed during the last collection period.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	> 0	Not Defined	DBMS job %job_name% for %owner% has failed %value% time(s) during last collection period.

State

The current state of the DBMS job.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	DISABLED	BROKEN	DBMS job %job_name% for %owner% is in %state% state.

Database Service Performance

This section provides information on the metrics in the Database Service Performance category, which report the performance statistics for all database services.

Target Version	Evaluation and Collection Frequency
All versions	Every 5 minutes

Metric Name	Description
Average Response Time (msec/call)	The average response time of the service during the last metric collection interval.
CPU Usage (%)	The percentage of the CPU time consumed by the service during the last metric collection interval.
Instance with Maximum Response Time	The database instance on which the service experienced the maximum response time during the last metric collection interval.
Instance with Minimum Response Time	The database instance on which the service experienced the minimum response time during the last metric collection interval.
Maximum Response Time (msec/call)	The maximum service response time during the last metric collection interval.
Minimum Response Time (msec/call)	The minimum service response time during the last metric collection interval.
Running Instances	The list of database instances the service is currently running on.
Status	The status of the service. Possible values are: • Up: The service is running on at least one database instance. • Down: The service is not running on any database instance.

Data Source

The data for these metrics is collected with service-related information from the following views:

- gv$servicemetric_history
- gv$active_services
- gv$osstat

Database Services

This section provides information on the metrics in the Database Services category, which report the configuration and status of clusterware-managed database services for the database.

Target Version	Evaluation and Collection Frequency
All versions	Every 15 minutes

Metric Name	Description
Cardinality	The configured cardinality setting for the service. Possible values are: • Uniform: The service is offered on all instances in the server pool. • Singleton: The service is running on only one instance at a time.
Pluggable Database Name	The name of the pluggable database for which the service is configured. If the database is not a container database or if the service is only configured for the root container, the value returned for this metric is Null.
Preferred Instances	The list of configured preferred database instances for the service.
Running Instances	The list of database instances the service is currently running on.
Status	The status of the service. Possible values are: • Up: Service is running on all preferred instances. • Up with Warning: Service is running on a subset of preferred instances. • Down: Service is not running on any instance.

Data Source

The data for these metrics are collected from the PL/SQL defined in the Enterprise Manager repository database, `EMD_RAC.service_status_buld_eval_proc`.

Database Wait Bottlenecks

This metric category contains the metrics that approximate the percentage of time spent waiting by user sessions across instances for the cluster database. This approximation takes system-wide totals and discounts the effects of sessions belonging to background processes.

Active Sessions Using CPU

This metric represents the active sessions using CPU.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 15 Minutes

Active Sessions Waiting: I/O

This database-level metric represents the active sessions waiting for I/O. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 15 Minutes

Active Sessions Waiting: Other

This database-level metric represents all the waits that are neither idle nor user I/O. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 15 Minutes

Average Database CPU (%)

This metric represents the average database CPU across instances as a percentage.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Host CPU Utilization (%)

This metric represents the percentage of CPU being used across hosts.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 15 Minutes

Load Average

This metric reports the sum of the current CPU load for all cluster database hosts.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Maximum CPU

This metric represents the total CPU count across all the cluster database hosts.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 15 Minutes

Wait Time (%)

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

This metric represents the percentage of time spent waiting, database-wide, for resources or objects during this sample period.

This test checks the percentage time spent waiting, database-wide, for resources or objects during this sample period. If the % Wait Time is greater than or equal to the threshold values specified by the threshold arguments, and the number of occurrences exceeds the value specified in the Number of Occurrences parameter, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
10*g*, 11*g*, 12*c*	Every 15 Minutes	Not Defined	Not Defined	Generated By Database Server

Data Source

The data source for this metric is the following formula where:

- DeltaTotalWait: Difference of 'sum of time waited for all wait events in v$system_event' between sample end and start.
- DeltaCpuTime: Difference of 'select value from v$sysstat where name='CPU used by this session' between sample end and start.

DeltaTotalWait / (DeltaTotalWait + DeltaCpuTime)

User Action

Investigate further into which specific wait events are responsible for the bulk of the wait time. Individual wait events may identify unique problems within the database. Diagnosis will be tailored where appropriate through drilldowns specific to individual wait events.

Database Vault Attempted Violations - Command Rules

The metrics in the Database Vault Attempted Violations - Command Rules metric category provides information about the attempted Database Vault command rule violations.

Database Vault Attempted Violations Count - Command Rules

This metric raises an alert, which helps the Oracle Database Vault security analyst to monitor violation attempts on the Database Vault database. This user can select the command rules to be affected by the alert and filter these command rules based on the different types of attempts by using error codes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
9*i*R2,10*g*, 11*g*, 12*c*	Every Hour	Not Defined	Not Defined	%ACTION_OBJECT_NAME% got violated at %VIOLATIONTIMESTAMP%

Database Vault Attempted Violations - Realms

The metrics in the Database Vault Attempted Violations - Realms metric category provide information about realm violations (for example, when an unauthorized user tries to modify an object that is protected by the realm).

Database Vault Attempted Violations Count - Realms

This metric raises an alert, which helps the Oracle Database Vault security analyst to monitor violation attempts on the Database Vault database. This user can select the realms to be affected by the alert and filter these realms based on the different types of attempts by using error codes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
9*i*R2,10*g*, 11*g*, 12*c*	Every Hour	Not Defined	Not Defined	%ACTION_OBJECT_NAME% got violated at %VIOLATIONTIMESTAMP%.

Database Vault Configuration Issues - Realms

The metrics in the Database Vault Configuration Issues - Realms metric category provide information about configuration issues in the realm. The Oracle Database Vault realm protects configuration information in the Oracle Database Vault.

Database Vault Configuration Issues Count - Realms

This metric tracks and raises an alert if users misconfigure realms.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
9*i*R2,10*g*, 11*g*, 12*c*	Every Hour	Not Defined	0	%ACTION_OBJECT_NAME% has configuration issues.

Database Vault Configuration Issues - Command Rules

The metrics in the Database Vault Configuration Issues - Command Rules metric category provide information about configuration issues in the command rules. A command rule is a rule that you create to protect SELECT, ALTER SYSTEM, database definition language (DDL), and data manipulation language (DML) statements that affect one or more database objects

DV (Command Rule) - Configuration Issue Count

This metric tracks and raises an alert if users misconfigure command rules.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
9*i*R2,10*g*, 11*g*, 12*c*	Every Hour	Not Defined	0	%ACTION_OBJECT_NAME% has configuration issues.

Database Vault Policy Changes

The metrics in the Database Vault Policy Changes metric category provide information about any changes to a Database Vault Policy.

Database Vault Policy Changes Count

This metric raises an alert on any change to any Database Vault policy, such as policies for realms and command rules.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
9*i*R2,10*g*, 11*g*, 12*c*	Every Hour	Not Defined	0	%ACTION_OBJECT_NAME% has configuration issues.

Datafile Allocation

This section provides information on the metrics in the Datafile Allocation category.

The allocated space is the current size of the datafile. A portion of this allocated space is used to store data while some may be free space. The metrics in this category check the amount of space used and the amount of space allocated to each datafile. The used space can then be compared to the allocated space to determine how much space is unused in the datafile. This metric is not intended for alerts. Rather it is intended for reporting. Historical views of unused allocated free space can help DBAs to correctly size their datafiles, eliminating wasted space.

Target Version	Evaluation and Collection Frequency
All versions	Every 24 hours

Metric Name	Description	Data Source
Allocated File Size (MB)	The space allocated to the datafile. This metric should be used in conjunction with the Used File Size (MB) metric to produce a historical view of the amount of space being used and unused in each datafile.	• Datafile: dba_data_files • Tempfile: dba_temp_files

Metric Name	Description	Data Source
Used File Size (MB)	The space used in the datafile. This metric should be used in conjunction with the Allocated File Size (MB) metric to produce a historical view of the amount of space being used and unused in each datafile.	• Datafile: Subtract allocated free space (dba_lmt_free_space and dba_dmt_free_space) from allocated file size (dba_data_files) • Tempfile: gv$temp_extent_pool.bytes_used

Deferred Transactions

This metric category contains the metrics associated with this distributed database's deferred transactions.

Deferred Transaction Count

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Oracle uses deferred transactions to propagate data-level changes asynchronously among master sites in an advanced replication system as well as from an updatable snapshot to its master table.

This metric checks for the number of deferred transactions. An alert is generated if the number of deferred transactions exceeds the value specified by the threshold argument.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every 5 Minutes	100	Not Defined	Number of deferred transactions is %value%.

Data Source

The data source for this metric is the following command:

```
SELECT count(*)
  FROM sys.deftran
```

User Action

When the advanced replication facility pushes a deferred transaction to a remote site, it uses a distributed transaction to ensure that the transaction has been properly committed at the remote site before the transaction is removed for the queue at the local site. If transactions are not being pushed to a given remote site, verify that the destination for the transaction was correctly specified. If you specify a destination database when calling DBMS_DEFER_SYS.SCHEDULE_EXECUTION using the DBLINK parameter, or

DBMS_DEFER_SYS.EXECUTE using the DESTINATION parameter, make sure the full database link is provided.

Wrong view destinations can lead to erroneous deferred transaction behavior. Verify that the DEFCALLEST and DEFTRANDEST views are the definitions from the CATREPC.SQL and not those from CATDEFER.SQL.

Deferred Transaction Error Count

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Oracle uses deferred transactions to propagate data-level changes asynchronously among master sites in an advanced replication system as well as from an updatable snapshot to its master table. If a transaction is not successfully propagated to the remote site, Oracle rolls back the transaction, logs the transaction in the SYS.DEFERROR view in the remote destination database.

This metric checks for the number of transactions in SYS.DEFERROR view and raises an alert if it exceeds the value specified by the threshold argument.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every 5 Minutes	0	Not Defined	Number of deferred transactions with errors is %value%.

Data Source

The data source for this metric is the following command:

```
SELECT count(*)
  FROM sys.deferror
```

User Action

An error in applying a deferred transaction may result from a database problem, such as a lack of available space in the table to be updated, or may be the result of an unresolved insert, update, or delete conflict. The SYS.DEFERROR view provides the ID of the transaction that could not be applied. Use this ID to locate the queued calls associated with the transaction. These calls are stored in the SYS.DEFCALL view. You can use the procedures in the DBMS_DEFER_QUERY package to determine the arguments to the procedures listed in the SYS.DEFCALL view.

Exadata Module Version Failure

This metric category provides information about any Exadata module version errors.

Error Count

This metric tracks and raises an alert when a defined number of Exadata module version errors occur.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 24 Hours	0	Not Defined	%errorCode% occurrences of %errorCount%.

Failed Logins

The metric in this metric category checks for the number of failed logins on the target database. This check is performed every ten minutes and returns the number of failed logins for that ten-minute interval. This metric will only work for databases where the audit_trail initialization parameter is set to DB or XML and the session is being audited.

Failed Login Count

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

This metric checks for the number of failed logins on the target database. This check is performed every ten minutes and returns the number of failed logins for that ten-minute interval. This metric will only work for databases where the audit_trail initialization parameter is set to DB or XML and the session is being audited.

If the failed login count crosses the values specified in the threshold arguments, then a warning or critical alert is generated. Because it is important to know every time a significant number of failed logins occurs on a system, this metric will generate a new alert for any ten-minute interval where the thresholds are crossed. You can manually clear these alerts. They will not automatically clear after the next collection.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every 30 Minutes	150	300	Number of failed login attempts exceeds threshold value.

Data Source

The database stores login information in different views based on the audit_trail setting. The database views used are:

- DB or DB_EXTENDED: DBA_AUDIT_SESSION
- XML (10*g* Release 2 only): DBA_COMMON_AUDIT_TRAIL

Fast Recovery

The metrics in the Fast Recovery metrics category relate to the fast recovery area.

Fast Recovery Area

Formerly referred to as flash recovery area, this metric returns an optional disk location that you can use to store recovery-related files such as control file and online redo log copies, archived redo log files, flashback logs, and RMAN backups.

Oracle Database and RMAN manage the files in the fast recovery area automatically. You can specify the disk quota, which is the maximum size of the fast recovery area.

Target Version	Collection Frequency
10*g*, 11*g*	Every 15 Minutes

Data Source

The data source for this metric is the following command:

```
SELECT value
  FROM v$parameter
  WHERE name='db_recovery_file_dest';
```

User Action

No user action is required.

Fast Recovery Area Size

This metric returns the Fast Recovery Area Size.

Target Version	Collection Frequency
10*g*, 11*g*	Every 15 Minutes

Data Source

The data source for this metric is the following command:

```
SELECT value
  INTO 1_fast_recovery_size
  FROM v$parameter
  WHERE name='db_recovery_file_dest_size';
```

User Action

No user action is required.

Flashback On

This metric returns whether or not flashback logging is enabled - YES, NO, or RESTORE POINT ONLY. For the RESTORE POINT ONLY option, flashback is ON but you can only flashback to guaranteed restore points.

Target Version	Collection Frequency
10*g*, 11*g*	Every 15 Minutes

Data Source

The data source for this metric is the following command:

```
SELECT flashback_on
  FROM v$database;
```

User Action

No user action is required.

Log Mode

This metric returns the log mode of the database - ARCHIVELOG or NOARCHIVELOG.

Target Version	Collection Frequency
10*g*, 11*g*	Every 15 Minutes

Data Source

The data source for this metric is the following command:

```
SELECT log_mode
  FROM v$database;
```

User Action

No user action is required.

Non-Reclaimable Fast Recovery Area (%)

This metric represents the percentage of space non-reclaimable (spaced used minus space reclaimable) in the fast recovery area.

Target Version	Collection Frequency
10*g*, 11*g*	Every 15 Minutes

Data Source

The data source for this metric is one of the following commands:

```
Non-reclaimable = space used - space reclaimable

Space Used:
  SELECT SUM(PERCENT_SPACE_USED
     FROM v$fast_recovery_area_usage;

Space Reclaimable:
  SELECT SUM(PERCENT_SPACE_RECLAIMABLE)
     FROM v$fast_recovery_area_usage;
```

User Action

No user action is required.

Oldest Flashback Time

This metric returns the oldest point-in-time to which you can flashback your database.

Target Version	Collection Frequency
10*g*, 11*g*	Every 15 Minutes

Data Source

The data source for this metric is the following command:

```
SELECT to_char(oldest_flashback_time, 'YYYY-MM-DD HH24:MI:SS')
  FROM v$flashback_database_log;
```

User Action

No user action is required.

Reclaimable Fast Recovery Area (%)

This metric represents the percentage of space reclaimable in the fast recovery area.

Target Version	Collection Frequency
10*g*, 11*g*	Every 15 Minutes

Data Source

The data source for this metric is the following command:

```
Space Reclaimable:
  SELECT SUM(PERCENT_SPACE_RECLAIMABLE)
     FROM v$fast_recovery_area_usage;
```

User Action

No user action is required.

Usable Fast Recovery Area (%)

This metric represents the percentage of space usable in the fast recovery area. The space usable is composed of the space that is free in addition to the space that is reclaimable.

Target Version	Collection Frequency
10*g*, 11*g*	Every 15 Minutes

Data Source

The data source for this metric is the following command:

```
SELECT (CASE WHEN PERCENT_USED > 100 THEN 0 ELSE (100-PERCENT_USED) END)
PERCENT_FREE
   FROM (SELECT (SUM(PERCENT_SPACE_USED)-SUM(PERCENT_SPACE_RECLAIMABLE))
PERCENT_USED
   FROM V$FAST_RECOVERY_AREA_USAGE);
```

User Action

No user action is required.

Fragmented Text Indexes

This metric category represents the number of text indexes in the database fragmented beyond the warning and critical percentage thresholds specified by the user. The collection is disabled by default. Before enabling this metric and specifying a metric threshold for the number of text indexes, the "Evaluate and Fix Text Index Fragmentation" job should be

submitted against the database target. The following details could be specified as part of the job parameters:

- Warning/Critical percentage threshold against which the text indexes are to be evaluated.
- List of text indexes to be evaluated (all indexes, specific schemas, or list of fully qualified names).
- List of text indexes to be fixed (all indexes, specific schemas, or list of fully qualified names). The scheduled DBMS job would attempt to fix the fragmented text indexes by optimizing (if warning threshold exceeded) or rebuilding them (if critical threshold exceeded, using shadow creation).
- The DBMS job schedule.

Fragmented Text Index count

This metric collects the total number of text indexes that have crossed the fragmentation percentage threshold specified by the user.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 hours	NA	NA	NA

Fragmented Text Index count crossing critical threshold

This metric collects the number of text indexes that have crossed the critical fragmentation percentage threshold specified by the user.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 hours	Not Defined	Not Defined	Fragmented Text Index count crossing critical threshold is %value%

Data Source

The fragmentation percentage for each index or index partition is derived by computing the data from DBA_IND_PARTITIONS, CTXSYS.CTX_INDEX_PARTITIONS, and its relevant text index metadata tables. The list of text indexes and the critical percentage threshold against which their fragmentation is to be evaluated are specified by the user as part of the "Evaluate and Fix Text Index Fragmentation" job.

User Action

A metric threshold could be set to generate incidents on the number of text indexes that have crossed the critical fragmentation threshold specified in "Evaluate and Fix Text Index Fragmentation" job. The scheduled DBMS job would automatically attempt to fix such text indexes (if they were specified in the fix list) by rebuilding them (using shadow creation). In addition, the incident also enables the user to fix the fragmented text indexes from the Enterprise Manager console.

Fragmented Text Index count crossing warning threshold

This metric collects the total number of text indexes that have crossed the warning fragmentation percentage threshold, but not the critical percentage threshold, specified by the user.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 hours	Not Defined	Not Defined	Fragmented Text Index count crossing warning threshold is %value%

Data Source

The fragmentation percentage for each index or index partition is derived by computing the data from DBA_IND_PARTITIONS, CTXSYS.CTX_INDEX_PARTITIONS, and its relevant text index metadata tables. The list of text indexes and the warning percentage threshold against which their fragmentation is to be evaluated are specified by the user as part of the "Evaluate and Fix Text Index Fragmentation" job.

User Action

A metric threshold could be set to generate incidents on the number of text indexes that have crossed the warning fragmentation threshold, but not the critical threshold, specified in "Evaluate and Fix Text Index Fragmentation" job. The scheduled DBMS job would automatically attempt to fix such text indexes (if they were specified in the fix list) by optimizing them. In addition, the incident also enables the user to fix the fragmented text indexes from the Enterprise Manager console.

High Availability (RMAN Configuration)

This section provides information on the metrics in the High Availability (RMAN Configuration) category, which lists RMAN persistent configuration settings.

Target Version	Evaluation and Collection Frequency
All versions	Every 24 hours

Metric Name	Description
Conf Number	A unique key identifying the RMAN configuration record within the target database that owns it.
Name	The name or type of configuration.
Value	The `CONFIGURE` command setting. Example, `RETENTION POLICY TO RECOVERY WINDOW OF 10 DAYS`.
Container ID	The ID of the container to which the data (the Conf Number, Name, and Value) pertains. Possible values include: • `0`: This value is used for rows containing data that pertains to the entire Container Database (CDB). This value is also used for rows in non-CDBs. • `1`: This value is used for rows containing data that pertains to only the root. • `n`: This value is used where `n` is the applicable container ID for the rows containing data.

High Availability Backup

This section provides information on the metrics in the High Availability Backup category.

Target Version	Evaluation and Collection Frequency
All versions	Every 12 hours

Metric Name	Description	Data Source
Command ID	The unique command ID of the backup command corresponding to this backup job.	v$rman_backup_job_details
End Time	The end time of the last backup command in the job.	v$rman_backup_job_details
Size of Input Files	The sum of all input file sizes backed up by this job, expressed as a string.	v$rman_backup_job_details
Input Type	The input type, which contains one of the following values: DB FULL, RECVR AREA, DB INCR, DATAFILE FULL, DATAFILE INCR, ARCHIVELOG, CONTROLFILE, SPFILE.	v$rman_backup_job_details
Size of Output Files	The output size of all pieces generated by this job, expressed as a string.	v$rman_backup_job_details
Output Bytes Per Sec	The generation rate of the output pieces for this backup, expressed as a string.	v$rman_backup_job_details
Output Device Type	The media device type for this backup job.	v$rman_backup_job_details
Session Key	The session key for this backup job.	v$rman_backup_job_details
Session RECID	The session RECID for this backup job.	v$rman_backup_job_details
Session Stamp	The session stamp for this backup job.	v$rman_backup_job_details
Start Time	The start time of the first backup command in the job.	v$rman_backup_job_details
Status	The status of the backup job.	v$rman_backup_job_details
Time Taken	The time taken for this backup job.	v$rman_backup_job_details

High Availability Backup History

This section provides information on the metrics in the High Availability Backup History category.

> **Note:**
>
> These metrics are not enabled out of the box, and must be enabled on the **Metric and Collection Settings** page. By default, these metrics are collected every 12 hours, but you can change the collection frequency.

Target Version	Evaluation and Collection Frequency
12c and later	Every 12 hours

Metric Name	Description	Data Source
Oracle Storage Container	If the backup is on Oracle Cloud storage, this specifies the Cloud storage container where the backup is located.	v$backup_piece_details
Compressed	Indicates whether the backup piece is compressed.	v$backup_piece_details
Compression Ratio	If the backup is compressed, this specifies the compression ratio of the backup.	v$rman_backup_job_details
Container ID	If the backup is on Oracle Cloud storage, this specifies the ID of the Cloud storage container where the backup is located.	v$rman_backup_job_details
Elapsed Seconds	The number of seconds that the backup job took to complete.	v$rman_backup_job_details
Encrypted	Indicates whether the backup was encrypted (YES or NO).	v$backup_piece_details
End Time	The end time of the last backup command in the job.	v$rman_backup_job_details
Incremental Level	Indicates the incremental level of this backup set.	v$backup_set_details
Size of Input Files (bytes)	The sum of all input file sizes backed up by this job, in bytes.	v$rman_backup_job_details
Size of Input Files	The sum of all input file sizes backed up by this job, expressed as a string.	v$rman_backup_job_details
Input Type	The input type, which contains one of the following values: DB FULL, RECVR AREA, DB INCR, DATAFILE FULL, DATAFILE INCR, ARCHIVELOG, CONTROLFILE, SPFILE.	v$rman_backup_job_details
Keep	Indicates whether or not this backup set has a retention policy.	v$backup_set_details
Keep Options	The additional retention options for this backup set.	v$backup_set_details
Keep Until	Indicates the date after which the backup becomes obsolete.	v$backup_set_details
Media	The name of the media on which the backup piece resides.	v$backup_piece_details
Name	The unique command ID for the backup command corresponding to this backup job.	v$rman_backup_job_details
Size of Output Files (bytes)	The output size of all pieces generated by this job, in bytes.	v$rman_backup_job_details
Size of Output Files	The output size of all pieces generated by this job, expressed as a string.	v$rman_backup_job_details
Output Bytes Per Sec (bytes/sec)	The generation rate of the output pieces for this backup, in bytes/second.	v$rman_backup_job_details
Output Bytes Per Sec	The generation rate of the output pieces for this backup, expressed as a string.	v$rman_backup_job_details
Output Device Type	The media device type for this backup job.	v$rman_backup_job_details
Session Key	The session key for this backup job.	v$rman_backup_job_details
Session RECID	The session RECID for this backup job.	v$rman_backup_job_details

Metric Name	Description	Data Source
Session Stamp	The session stamp for this backup job.	v$rman_backup_job_details
Start Time	The start time of the first backup command in the job.	v$rman_backup_job_details
Status	The status of the backup job.	v$rman_backup_job_details
Tag	The tag for this backup job.	v$backup_piece_details
Time Taken	The time taken for this backup job.	v$rman_backup_job_details

High Availability Client Recovery Window

This section provides information on the metrics in the High Availability Client Recovery Window category.

Note that the data source for these metrics is the database control file, and the collection for these metrics is disabled by default. If the database is backing up to a Recovery Appliance, these metrics are not applicable and the collection should remain disabled. If the database is not backing up to a Recovery Appliance and you want to monitor the database recovery window, you can enable the collection. In this case, the data in these metrics is used as the source data for the related High Availability Recovery Window metric category. See High Availability Recovery Window.

Target Version	Evaluation and Collection Frequency
12c and later	Every 15 minutes

Metric Name	Description	Data Source
Disk Recovery Window (seconds)	The recovery window for disk backups.	v$disk_restore_range
Disk Unprotected Data Window (seconds)	The current amount of potential data loss for disk backups.	v$disk_restore_range
Last Complete Disk Backup Date	The latest point in time for which a complete disk backup is available for all datafiles in this database.	v$disk_restore_range
Last Complete Media Backup Date	The latest point in time for which a complete media backup is available for all datafiles in this database.	v$sbt_restore_range
Media Recovery Window (seconds)	The recovery window for media backups.	v$sbt_restore_range
Media Unprotected Data Window (seconds)	The current amount of potential data loss for media backups.	v$sbt_restore_range

High Availability Data Guard Target Summary

This section provides information on the metrics in the High Availability Data Guard Target Summary category.

Target Version	Evaluation and Collection Frequency
All versions	Every 24 hours

Metric Name	Description
Source Type	The role (Primary or Standby) of the database that was the source of the data row.
Row Type	The role (Primary or Standby) of the database to which the data in the row pertains.
Using Broker	Whether the Data Guard configuration for the database specified by the row is using Data Guard broker.
Active Standby	Whether the database specified by the row is an Active Data Guard standby database.
Database Unique Name	The value of the DB_UNIQUE_NAME initialization parameter for the database specified by the row.
Database ID	The value of DBID (as found in v$database) of the database specified by the row.
Primary Database Unique Name	The Database Unique Name of the primary database associated with the database specified by the row (if the database is a standby database).
Primary Database ID	The Database ID of the primary database associated with the database specified by the row.
Role	The Data Guard role of the database specified by the row.
Standby Database List	The list of standby databases associated with the database specified by the row (if the database is a primary database).
Protection Mode	The protection mode for the database specified by the row.
Fast-Start Failover Status	The fast-start failover status for the database specified by the row.
Status	The Data Guard status for the database specified by the row.
Redo Source	The Database Unique Name of the database that is shipping redo to the database specified by the row.
Data Guard Connect Identifier	The net connect identifier used to reach the database.

High Availability Disk Backup

This section provides information on the metrics in the High Availability Disk Backup category.

Target Version	Evaluation and Collection Frequency
All versions	Every 30 minutes

Metric Name	Description	Data Source
Time Since Last Successful Full Backup (hours)	The time since the last successful full disk backup, in hours. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful full database disk backup was %value% hours ago.	v$rman_backup_job_details
Time Since Last Successful Incremental Backup (hours)	The time since the last successful incremental disk backup, in hours. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful incremental database disk backup was %value% hours ago.	v$rman_backup_job_details
Time Since Last Successful Archived Log Backup (minutes)	The time since the last successful archived log disk backup, in minutes. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful archived log disk backup was %value% minutes ago.	v$rman_backup_job_details
Last Executed Full Backup Status	The status of the last executed full disk backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed full database disk backup status was %value%.	v$rman_backup_job_details
Last Executed Incremental Backup Status	The status of the last executed incremental disk backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed incremental database disk backup status was %value%.	v$rman_backup_job_details
Last Executed Archived Log Backup Status	The status of the last executed archived log disk backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed archived log disk backup status was %value%.	v$rman_backup_job_details
Recovery Window (seconds)	This column is obsolete and is not populated. This data is now available in the High Availability Client Recovery Window metric category.	v$disk_restore_range
Last Successful Archived Log Backup Date	The date of the last successful archived log disk backup.	v$rman_backup_job_details
Last Successful Archived Log Backup Size (bytes)	The size of the last successful archived log disk backup, in bytes.	v$rman_backup_job_details
Last Complete Backup Date	The latest point in time for which a complete disk backup is available for all datafiles.	v$disk_restore_range
Last Successful Full Backup Date	The date of the last successful full disk backup.	v$rman_backup_job_details
Last Successful Full Backup Size (bytes)	The size of the last successful full disk backup, in bytes.	v$rman_backup_job_details

Metric Name	Description	Data Source
Last Executed Archived Log Backup Date	The date of the last executed archived log disk backup.	v$rman_backup_job_details
Last Executed Full Backup Date	The date of the last executed full disk backup.	v$rman_backup_job_details
Last Executed Incremental Backup Date	The date of the last executed incremental disk backup.	v$rman_backup_job_details
Last Executed Incremental Level 0 Backup Status	The status of the last executed incremental level 0 disk backup.	v$rman_backup_job_details
Last Executed Incremental Level 1 Backup Status	The status of the last executed incremental level 1 disk backup.	v$rman_backup_job_details
Last Successful Incremental Backup Date	The date of the last successful incremental disk backup.	v$rman_backup_job_details
Last Successful Incremental Backup Size (bytes)	The size of the last successful incremental disk backup, in bytes.	v$rman_backup_job_details
Time Since Last Successful Incremental Level 0 Backup (hours)	The time since the last successful incremental level 0 disk backup, in hours.	v$rman_backup_job_details
Last Successful Incremental Level 0 Backup Size (bytes)	The size of the last successful incremental level 0 disk backup, in bytes.	v$rman_backup_job_details
Time Since Last Successful Incremental Level 1 Backup (hours)	The time since the last successful incremental level 1 disk backup, in hours.	v$rman_backup_job_details
Last Successful Incremental Level 1 Backup Size (bytes)	The size of the last successful incremental level 1 disk backup, in bytes.	v$rman_backup_job_details
Unprotected Data Window (seconds)	This column is obsolete and is not populated. This data is now available in the High Availability Client Recovery Window metric category.	v$disk_restore_range

High Availability Media Backup

This section provides information on the metrics in the High Availability Media Backup category.

Target Version	Evaluation and Collection Frequency
All versions	Every 30 minutes

Metric Name	Description	Data Source
Time Since Last Successful Full Backup (hours)	The time since the last successful full media backup, in hours. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful full database media backup was %value% hours ago.	v$rman_backup_job_details
Time Since Last Successful Archived Log Backup (minutes)	The time since the last successful archived log media backup, in minutes. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful archived log media backup was %value% minutes ago.	v$rman_backup_job_details
Time Since Last Successful Incremental Backup (hours)	The time since the last successful incremental media backup, in hours. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last successful incremental database media backup was %value% hours ago.	v$rman_backup_job_details
Last Executed Archived Log Backup Status	The status of the last executed archived log media backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed archived log media backup status was %value%.	v$rman_backup_job_details
Last Executed Full Backup Status	The status of the last executed full media backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed full database media backup status was %value%.	v$rman_backup_job_details
Last Executed Incremental Backup Status	The status of the last executed incremental media backup. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The last executed incremental database media backup status was %value%.	v$rman_backup_job_details
Recovery Window (seconds)	This column is obsolete and is not populated. This data is now available in the High Availability Client Recovery Window metric category.	v$sbt_restore_range
Last Successful Archived Log Backup Date	The date of the last successful archived log media backup.	v$rman_backup_job_details
Last Successful Archived Log Backup Media	The name of the media on which the last successful archived log backup resides.	v$backup_piece_details
Last Successful Archived Log Backup Size (bytes)	The size of the last successful archived log media backup, in bytes.	v$rman_backup_job_details
Last Complete Backup Date	The latest point in time for which a complete media backup is available for all datafiles.	v$sbt_restore_range
Last Successful Full Backup Date	The date of the last successful full media backup.	v$rman_backup_job_details

Metric Name	Description	Data Source
Last Successful Full Backup Media	The name of the media on which the last successful full backup resides.	v$backup_piece_details
Last Successful Full Backup Size (bytes)	The size of the last successful full media backup, in bytes.	v$rman_backup_job_details
Last Executed Archived Log Backup Date	The date of the last executed archived log media backup.	v$rman_backup_job_details
Last Executed Full Backup Date	The date of the last executed full media backup.	v$rman_backup_job_details
Last Executed Incremental Backup Date	The date of the last executed incremental media backup.	v$rman_backup_job_details
Last Executed Incremental Level 0 Backup Status	The status of the last executed incremental level 0 media backup.	v$rman_backup_job_details
Last Executed Incremental Level 1 Backup Status	The status of the last executed incremental level 1 media backup.	v$rman_backup_job_details
Last Successful Incremental Backup Date	The date of the last successful incremental media backup.	v$rman_backup_job_details
Last Successful Incremental Backup Media	The name of the media on which the last successful incremental backup resides.	v$backup_piece_details
Last Successful Incremental Backup Size (bytes)	The size of the last successful incremental media backup, in bytes.	v$rman_backup_job_details
Time Since Last Successful Incremental Level 0 Backup (hours)	The time since the last successful incremental level 0 media backup, in hours.	v$rman_backup_job_details
Last Successful Incremental Level 0 Backup Media	The name of the media on which the last successful incremental level 0 backup resides.	v$rman_backup_job_details
Last Successful Incremental Level 0 Backup Size (bytes)	The size of the last successful incremental level 0 media backup, in bytes.	v$rman_backup_job_details
Time Since Last Successful Incremental Level 1 Backup (hours)	The time since the last successful incremental level 1 media backup, in hours.	v$rman_backup_job_details
Last Successful Incremental Level 1 Backup Media	The name of the media on which the last successful incremental level 1 backup resides.	v$rman_backup_job_details
Last Successful Incremental Level 1 Backup Size (bytes)	The size of the last successful incremental level 1 media backup, in bytes.	v$rman_backup_job_details
Unprotected Data Window (seconds)	This column is obsolete and is not populated. This data is now available in the High Availability Client Recovery Window metric category.	v$sbt_restore_range

High Availability Recovery Window

This section provides information on the metrics in the High Availability Recovery Window category.

Note that the data source for these metrics depends on the database backup destination. If the database is backing up to a Recovery Appliance, all of the data is sourced from the Recovery Appliance, and metrics that are applicable only in the Recovery Appliance case are noted. If the database is not backing up to a Recovery Appliance, all data is sourced from the High Availability Client Recovery Window metric category, which in turn gets its data from the local database control file. In this case, if there is no data for these metrics, it may be because the High Availability Client Recovery Window collection is disabled. See High Availability Client Recovery Window.

Target Version	Evaluation and Collection Frequency
All versions	Every 15 minutes

Metric Name	Description	Data Source
Recovery Appliance Downstream One	The name of the first downstream Recovery Appliance that is receiving replicated backups for this database. (Note that this is applicable only to databases backing up to a Recovery Appliance.)	–
Recovery Appliance Downstream Two	The name of the second downstream Recovery Appliance that is receiving replicated backups for this database. (Note that this applicable only to databases backing up to a Recovery Appliance.)	–
Final Change Number	The highest SCN to which this database can be recovered when using backups and redo logs available on the Recovery Appliance. (Note that this is applicable only to databases backing up to a Recovery Appliance.)	rc_database on Recovery Appliance
Last Complete Disk Backup	The latest point in time for which a complete disk backup is available for all datafiles in this database. If the database is backing up to a Recovery Appliance, this is based on backups contained in Recovery Appliance disk storage.	v$disk_restore_range, if the database is configured to backup to a disk. ra_database, if the database is configured to backup to a Recovery Appliance.
Last Complete Media Backup	The latest point in time for which a complete media backup is available for all datafiles in this database. If the database is backing up to a Recovery Appliance, this is based on backups copied by the Recovery Appliance to tape or Cloud storage.	v$sbt_restore_range
Recovery Appliance	The Recovery Appliance that this database is currently backing up to, if any.	–
Recovery Appliance Replication Server List	The list of replication servers configured for this database on the Recovery Appliance. (Note that this is applicable only to databases backing up to a Recovery Appliance.)	ra_protection_policy on Recovery Appliance
Disk Recovery Window Goal (seconds)	The recovery window goal specified within the Recovery Appliance protection policy applicable to this database. (Note that this is applicable only to databases backing up to a Recovery Appliance.)	ra_protection_policy on Recovery Appliance

Metric Name	Description	Data Source
Disk Unprotected Data Window Threshold (seconds)	The unprotected data window threshold specified within the Recovery Appliance protection policy applicable to this database. (Note that this is applicable only to databases backing up to a Recovery Appliance.)	ra_database on Recovery Appliance
Disk Unprotected Data Window (seconds)	The current amount of potential data loss for disk backups. If the database is backing up to a Recovery Appliance, this is the amount of data not present in backups contained in Recovery Appliance disk storage. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The disk unprotected data window is %value% seconds.	v$disk_restore_range, if the database is configured to backup to a disk. ra_database, if the database is configured to backup to a Recovery Appliance.
Disk Recovery Window (seconds)	The current database recovery window for disk backups. If the database is backing up to a Recovery Appliance, this is the current disk recovery window reported for this database by the Recovery Appliance, based on backups contained in Recovery Appliance disk storage. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The disk recovery window is %value% seconds.	v$disk_restore_range, if the database is configured to backup to a disk. ra_disk_restore_range, if the database is configured to backup to a Recovery Appliance.
Media Recovery Window (seconds)	The current database recovery window for media backups. If the database is backing up to a Recovery Appliance, this is the current media recovery window reported for this database by the Recovery Appliance, based on backups copied by the Recovery Appliance to attached tape or Cloud storage. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The media recovery window is %value% seconds.	v$sbt_restore_range
Media Unprotected Data Window (seconds)	The current amount of potential data loss for media backups. If the database is backing up to a Recovery Appliance, this is the amount of data not present in backups copied by the Recovery Appliance to tape or Cloud storage. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: The media unprotected data window is %value% seconds.	v$sbt_restore_range

Invalid Objects

The metrics in this category represent the number of invalid objects in the database.

Invalid Object Count

This metric represents the total number of invalid objects in the database.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Invalid Object Count in the database is %value%

Data Source

The data is derived from the SYS.OBJ$ and SYS.USER$ tables.

User Action

The “Recompile Invalid Objects” corrective action could be setup against the incident to automatically attempt to recompile the invalid objects in the database. Some objects might need specific corrective steps to be performed manually before re-compilation.

Invalid Objects by Schema

The metrics in this category represent the number of invalid objects in each schema.

Invalid Object Count by Schema

This metric represents the total number of invalid objects per schema.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Invalid Object Count in %owner% schema is %value%

Multiple Thresholds

Different warning and critical threshold values could be set for each Invalid Object Owner (schema) object.

If warning or critical threshold values are currently set for any Invalid Object Owner object, those thresholds could be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Invalid Object Owner object, use the Edit Thresholds page.

Data Source

The data is derived from the SYS.OBJ$ and SYS.USER$ tables.

User Action

The “Recompile Invalid Objects” corrective action could be setup against the incident to automatically attempt to recompile the invalid objects in a schema. Some objects might need specific corrective steps to be performed manually before recompilation.

Messages Per Buffered Queue

The metrics in the Messages Per Buffered Queue metrics category monitor the age and state of the first (top of the queue) message for each buffered queue in the database except for the system queues. Queues that are in the schema of SYS, SYSTEM, DBSNMP, and SYSMAN are defined as system level queues.

Average age of messages per buffered queue (seconds)

This metric provides the average age (in seconds) of the messages in the buffered queue for all nonsystem queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Average age of messages in %schema%.%queue_name% queue is %value% seconds.

First Message Age in Buffered Queue Per Queue (Seconds)

This metric gives the age (in seconds) of the first message in the buffered queue for all non-system queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Age of first message in %schema%.%queue_name% buffered queue is %value% seconds.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Schema Name and Queue Name objects.

If warning or critical threshold values are currently set for any unique combination of Schema Name and Queue Name objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Schema Name and Queue Name objects, use the Edit Thresholds page.

Data Source

This metric is calculated by finding the age of the first message in all the subscribers of the queue and then the oldest amongst all is taken.

The following views and tables are used for the calculation:

1. <SCHEMA>.AQ$<QUEUE_TABLE>
2. v$buffered_queues

User Action

When using buffered queues for storing and propagating messages, monitor this metric to get the age of first message in the queue.

Messages processed per buffered queue (%)

This metric gives the messages processed percentage per minute per buffered queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Messages processed for queue %schema%.%queue_name% is %value% percent.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Schema Name and Queue Name objects.

If warning or critical threshold values are currently set for any unique combination of Schema Name and Queue Name objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Schema Name and Queue Name objects, use the Edit Thresholds page.

Data Source

This is calculated as the percent of total number of messages processed per minute and total number of messages received per minute in the last collection interval per buffered queue.

User Action

When using queues for storing/propagating messages, monitor this metric to get the messages processed percent (or throughput) per minute in the last collection interval for the queue.

Messages processed per buffered queue (%) per minute

This metric gives the messages processed percentage per minute in the last interval per buffered queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Messages processed per minute in the last interval for queue %schema%.%queue_name% is %value% .

Spilled Messages

This metric displays the current number of overflow messages spilled to disk from the buffered queue.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Current number of overflow messages spilled to disk from the buffered queue %schema%.%queue_name% is %value%

Total Messages Processed per Buffered Queue per Minute

This metric gives the total number of messages processed per minute per buffered queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Total messages processed per minute in the last interval for queue %schema%.%queue_name% is %value% .

Total Messages Received per Buffered Queue per Minute

This metric gives the total number of messages received or enqueued into the buffered queue per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Total messages received per minute in the last interval for queue %schema%.%queue_name% is %value% .

Messages Per Buffered Queue Per Subscriber

This metric category monitors the messages for buffered queues per subscriber in the database.

Average Age of Messages Per Buffered Queue Per Subscriber (Seconds)

This metric display's the average age of messages in the buffered queue per queue in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Average age of messages for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

First Message Age in Buffered Queue per Subscriber (Seconds)

This metric displays the age of the first message in the buffered queue per queue per subscriber in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Age of first message for subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

Messages Processed Per Buffered Queue (%) Per Subscriber Per Minute

This metric gives the total number of messages processed per minute per buffered queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value%

Messages Processed Per Buffered Queue Per Subscriber (%)

This metric gives the messages processed percentage for the buffered queue per subscriber. Messages processed percent is calculated as the percent of the total number messages processed or dequeued to the total number of messages received or enqueued.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Messages processed for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% percent.

Total Messages Processed Per Buffered Queue Per Subscriber Per Minute

This metric gives the total number of messages processed per minute per buffered queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Total messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% .

Total Messages Received Per Buffered Queue Per Subscriber Per Minute

This metric gives the total number of messages received or enqueued into the queue per subscriber per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Total messages received per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% .

Messages Per Persistent Queue

The metrics in the Messages Per Persistent Queue metrics category monitor the age and state of the first (top of the queue) message for each persistent queue in the database except for the system queues. Queues that are in the schema of SYS, SYSTEM, DBSNMP, and SYSMAN are defined as system level queues.

Average Age of Messages Per Persistent Queue (Seconds)

This metric displays the average age of messages in the persistent queue per queue in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Average age of messages in %schema%.%queue_name% queue is %value% seconds.

Age of The First Message in Persistent Queue Per Queue (Seconds)

This metric gives the age (in seconds) of the first message in the persistent queue for all non-system queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Age of first message in %schema%.%queue_name% queue is %value% seconds.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Schema Name and Queue Name objects.

If warning or critical threshold values are currently set for any unique combination of Schema Name and Queue Name objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Schema Name and Queue Name objects, use the Edit Thresholds page.

Data Source

This metric is calculated by finding the age of the first message in all the subscribers of the queue and then the oldest amongst all is taken.

The following views/tables are used for the calculation:

1. <SCHEMA>.AQ$_<QUEUE_TABLE>_S
2. <SCHEMA>.AQ$_<QUEUE_TABLE>_I
3. <SCHEMA>.AQ$<QUEUE_TABLE>

User Action

When using persistent queues for storing and propagating messages, monitor this metric to get the age of first message in the queue.

Messages Processed Per Persistent Queue (%)

This metric gives the messages processed percentage for the persistent queue. Messages processed percent is calculated as the percent of the total number messages processed or dequeued to the total number of messages received or enqueued.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Messages processed for queue %schema%.%queue_name% is %value% percent.

Messages Processed Per Persistent Queue (%) Per Minute

This metric gives the messages processed percentage per minute per persistent queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Messages processed per minute in the last interval for queue %schema%.%queue_name% is %value%

Total Messages Processed per Persistent Queue per Minute

This metric gives the total number of messages processed per minute per persistent queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Total messages processed per minute in the last interval for queue %schema%.%queue_name% is %value% .

Total Messages Received per Persistent Queue per Minute

This metric gives the total number of messages received or enqueued into the queue per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Total messages received per minute in the last interval for queue %schema%.%queue_name% is %value% .

Messages Per Persistent Queue Per Subscriber

The metrics in the Messages Per Persistent Queue Per Subscriber metrics category monitor the age and state of the first (top of the queue) message for each persistent queue per queue subscriber in the database except for the system queues. Queues that are in the schema of SYS, SYSTEM, DBSNMP, and SYSMAN are defined as system level queues.

Average Age of Messages Per Persistent Queue Per Subscriber (Seconds)

This metric display's the average age of messages in the persistent queue per queue in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Average age of messages for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

First Message Age in Persistent Queue per Subscriber (Seconds)

This metric gives the age (in seconds) of the first message in the persistent queue per subscriber for all non-system queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Age of first message for subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

Messages Processed Per Persistent Queue (%) Per Subscriber Per Minute

This metric gives the messages processed percentage per minute per persistent queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value%.

Messages Processed Per Persistent Queue Per Subscriber (%)

This metric gives the messages processed percentage for the persistent queue per subscriber. Messages processed percent is calculated as the percent of the total number messages processed or dequeued to the total number of messages received or enqueued.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Messages processed for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% percent.

Total Messages Processed Per Persistent Queue Per Subscriber Per Minute

This metric gives the messages processed percentage per minute per persistent queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value%.

Total Messages Received Per Persistent Queue Per Subscriber Per Minute

This metric gives the total number of messages received or enqueued into the queue per subscriber per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Total messages received per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% .

PDB Mode (All Pluggable Databases)

This is a CDB-level metric category that provides the current mode of the PDB targets in a RAC instance. It covers the READ ONLY, READ WRITE, MOUNTED, READ ONLY RESTRICTED, and READ WRITE RESTRICTED modes.

Target Version	Collection Frequency
All versions	Every 5 minutes

Metric Name	Description
Any Restricted	Indicates if the PDB is open in restricted modes (Read Only Restricted or Read Write Restricted). Possible values are: • YES • NO
Mode	Open mode of the PDB. Possible values are: • READ ONLY • READ WRITE • MOUNTED
Read Only Restricted	Indicates if the PDB is in Read Only Restricted mode. Possible values are: • YES • NO
Read Write Restricted	Indicates if the PDB is in Read Write Restricted mode. Possible values are: • YES • NO

Monitoring User Account

The metrics in this category provide visibility into potential problems with the Monitoring User account (for example DBSNMP) to prevent a lapse in monitoring.

Monitoring User Connectivity Issue

This metric monitors the expiry of the Monitoring User account password, and raises an alert when the password is not updated in Oracle Enterprise Manager's target configuration.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 30 Minutes	ORA-	ORA-	Connection for monitoring user %USER_NAME% failed with error %PASSWORD_INVALID%.

Monitoring User Expiry

This metric monitors the potential expiry of the Monitoring User account.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 24 Hours	72	Not Defined	Monitoring user %USER_NAME% will expire in %ACCOUNT_EXPIRY_IN_HOURS% hours.

Database Monitoring User Privileges Check

This metric checks whether the Monitoring User account has monitoring privileges on par with the DBSNMP or SYSDBA user accounts. This is useful when using a non-DBSNMP user account.

The collection is disabled by default.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 48 Hours	Not Defined	FALSE	Monitoring user %USER_NAME% does not have sufficient monitoring privileges under %ROLE% role. It must have monitoring privileges equal or higher than DBSNMP user.

QoS Management - Performance Satisfaction

Oracle Database Quality of Service (QoS) Management is an automated, policy-based product that monitors the workload requests for an entire system.

For more information, see *Oracle Database Quality of Service Management User's Guide*.

Negative PSM Duration (seconds)

This metric tracks the negative PSM duration and raises an alert when it exceeds its threshold.

Performance Satisfaction Metric (PSM) is a normalized numeric value that indicates how well a particular Performance Objective is being met, and which enables Oracle Database QoS Management to compare the performance of the system for widely differing Performance Objectives.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 5 Minutes	Not Defined	Not Defined	Negative PSM Duration value %value% for Performance Class %PC% has crossed the threshold.

Recovery

This metric category contains the metrics representing database recovery.

Corrupt Data Block Count

This metric represents the count of corrupt data blocks.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Metric Summary 9iR2 or higher Evaluated and Collected every 15 minutes Operator > Warning Threshold - 0 Critical Threshold - Not Defined Number of corrupt data blocks is %value%.

Data Source

The data source for this metric is the following command:

```
SELECT count(unique(file#))
  FROM v$database_block_corruption;
```

User Action

Perform a database recovery.

Missing Media File Count

This metric represents the count of missing media files.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Metric Summary 8i or higher Evaluated and Collected every 15 minutes Operator > Warning Threshold - 0 Critical Threshold - Not Defined Number of missing media files is %value%.

Data Source

This metric is calculated with the following command:

```
SELECT count(file#)
  FROM v$datafile_header
  WHERE recover ='YES' OR error is not null;
```

User Action

You should perform a database recovery.

Recovery Area

This metric category contains the Recovery Area metrics that enable you to monitor Fast Recovery Area usage. These metrics represent the respective space consumption as a percentage, and are database-level metrics that are evaluated by the database server every 15 minutes or during file creation, whichever occurs first. The metric data is also printed in the alert log. For cluster databases, these metrics are monitored at the cluster database target level and not by member instances.

Recovery Area Free Space (%)

This metric represents the recovery area free space as a percentage. The Critical Threshold is set for < 3% and the Warning Threshold is set for < 15%. You cannot customize these thresholds. An alert is returned the first time the alert occurs, and the alert is not cleared until the available space rises above 15%.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 15 minutes or during file creation, whichever occurs first	15% (cannot be changed)	3% (cannot be changed)	db_recovery_file_dest_size of N bytes is N% used, and has N remaining bytes available

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Recovery Area Used Space (%)

This metric represents the recovery area used space as a percentage. The Critical threshold is set for < 97% and the Warning threshold is set for < 85% and these can be customized. Any changes will directly be applied on the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11gR2, 11gR202, 12c, 12cR102, 12cR2	Every 15 minutes or during file creation, whichever occurs first	None	None	–
18c and later	Every 15 minutes or during file creation, whichever occurs first	85%	97%	The value of Recovery Area Used Space (%) for RECOVERY AREA is XX.YYY.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

SCN Growth Statistics

This metric category provides information about the Systems Change Number (SCN) in the database environment and reports on the health of the SCN growth in the database.

SCN Health

This metric displays the status of the SCN health.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
10*g*, 11*g*, 12*c*	Every Hour	62	10	The SCN health is %scn_health%.

SCN Max Statistics

This metric category provides information about the maximum value of the SCN.

Max SCN Jump in one second (last 24 hours)

This metric displays the maximum SCN jump in one second over the previous 24 hours.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
10*g*, 11*g*, 12*c*	Every Hour	Not Defined	Not Defined	The maximum SCN jump in one second (last 24 hours) is %scn_max_jump%.

Segment Advisor Recommendations

This metric category contains metrics related to the Automatic Segment Advisor job.

Oracle uses the Automatic Segment Advisor job to detect segment issues regularly within maintenance windows. It determines whether the segments have unused space that can be released. The Number of recommendations is the number of segments that have Reclaimable Space. The recommendations come from all runs of the automatic segment advisor job and any user-scheduled segment advisor jobs.

Number of recommendations

Oracle uses the Automatic Segment Advisor job to detect segment issues regularly within maintenance windows. It determines whether the segments have unused space that can be released. The Number of recommendations is the number of segments that have Reclaimable Space. The recommendations come from all runs of the automatic segment advisor job and any user-scheduled segment advisor jobs.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Session Suspended

This metric category contains the metrics that represent the number of resumable sessions that are suspended due to a correctable error.

Session Suspended by Data Object Limitation

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

This metric represents the session suspended by a data object limitation.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 5 Minutes

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Session Suspended by Quota Limitation

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

This metric represents the session suspended by a quota limitation.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 5 Minutes

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Session Suspended by Rollback Segment Limitation

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

This metric represents the session suspended by a rollback segment limitation.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 5 Minutes

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Session Suspended by Tablespace Limitation

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

This metric represents the session suspended by a tablespace limitation.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 5 Minutes

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Snapshot Too Old

This metric category contains the snapshot too old metrics.

Snapshot Too Old due to Rollback Segment Limit

This database-level metric represents snapshots too old because of the rollback segment limit. This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Snapshot Too Old due to Tablespace Limit

This database-level metric represents snapshots too old because of the tablespace limit. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

User Action

View the latest Automatic Database Diagnostic Monitor (ADDM) report. For a more detailed analysis, run ADDM from the Advisor Central link on the Database Home page.

Space Usage by Buffered Queues

This metric category monitors the space usage of buffered queues with respect to the streams pool size.

Queue Size (MB)

This metric display's the size of buffered queue, which is the total number of Mega bytes allocated for all messages and metadata.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Size of buffered queue %schema%.%queue_name% is %value% MB.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Schema Name and Queue Name objects.

If warning or critical threshold values are currently set for any unique combination of Schema Name and Queue Name objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Schema Name and Queue Name objects, use the Edit Thresholds page.

Data Source

The source of this metric is the INSTANCE_NAME column from GV$INSTANCE view.

User Action

When using queues for storing or propagating messages, monitor this metric to get the instance in which the buffered queue is available.

Space Usage of Buffered Queue With Respect to Streams Pool Size (%)

This metric gives the space usage percentage of buffered queue with respect to streams pool size per buffered queue.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Buffered queue %schema%.%queue_name% has consumed %value% percent of streams pool size.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Schema Name and Queue Name objects.

If warning or critical threshold values are currently set for any unique combination of Schema Name and Queue Name objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Schema Name and Queue Name objects, use the Edit Thresholds page.

Data Source

The source of this metric is the QUEUE_SIZE AND CURRENT_SIZE columns from GV$BUFFERED_QUEUES and GV$SGA_DYNAMIC_COMPONENTS views.

User Action

When using buffered queues for storing or propagating messages, monitor this metric to get the space usage percentage of buffered queue with respect to the allocated streams pool size.

Streams Apply Queue - Buffered

The metrics in the Streams Apply Queue - Buffered metrics category show the current total number of messages in a buffered queue to be dequeued by each apply process and the total number of messages to be dequeued by each apply process that have spilled from memory into the persistent queue table.

Streams Apply - (%)Spilled Messages

This metric usually indicates that transactions are staying longer in memory.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
11*g*R2, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Spilled messages for Apply process [%APPLY_NAME%] queue is %value% percent.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Apply Name object.

If warning or critical threshold values are currently set for any Apply Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Apply Name object, use the Edit Thresholds page.

Data Source

The source for this metric is the target database in the gv$buffered_queues and gv$buffered_subscribers tables.

User Action

Either increase Streams Pool size and /or increase Apply Parallelism to speed up Apply processing.

Streams Apply Queue - Persistent

The metrics in the Streams Apply Queue - Persistent metrics category show the number of messages in a persistent queue in READY state and WAITING state for each apply process.

Streams Apply - (%)Messages in Waiting State

This metric shows the percentage of messages in a wait state.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
9*i*R2, 10*g*, 11*g*, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Messages waiting for Apply process [%APPLY_NAME%] queue is %value% percent.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Apply Name and Messages Delivery Mode objects.

If warning or critical threshold values are currently set for any unique combination of Apply Name and Messages Delivery Mode objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Apply Name and Messages Delivery Mode objects, use the Edit Thresholds page.

Data Source

The data source for this metric is Target Database and Apply Queue.

User Action

No user action is required.

Streams Apply Reader Statistics

The reader server for an apply process dequeues messages from the queue. The reader server computes dependencies between LCRs and assembles messages into transactions.

The reader server then returns the assembled transactions to the coordinator, which assigns them to idle apply servers.

The metrics in this metric category show the total number of messages dequeued by the reader server for the apply process since the last time the apply process was started.

Rate at Which Messages Are Getting Spilled (Per Sec)

The reader server for an apply process dequeues messages from the queue. The reader server computes dependencies between LCRs and assembles messages into transactions. The reader server then returns the assembled transactions to the coordinator, which assigns them to idle apply servers.

This metric shows the rate at which message are getting spilled (per second) by the reader server for the apply process since the last time the apply process was started.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
9*i*R2, 10*g*, 11*g*, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Total number of spilled messages for Apply Process [%APPLY_NAME%] is %value% .

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Apply Name object.

If warning or critical threshold values are currently set for any Apply Name object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Apply Name object, use the Edit Thresholds page.

Data Source

For this metric, the data source is Target database, gv$streams_apply_reader view.

User Action

No user action is required.

Streams Capture Queue Statistics

The metrics in this metric category show the current total number of messages in a buffered queue that were enqueued by each capture process and the total number of messages enqueued by each capture process that have spilled from memory into the queue spill table.

If queue publishers other than the capture process enqueue messages into a buffered queue, then the values shown can include messages from these other queue publishers.

Streams Capture - (%)Spilled Messages

Queue spill indicates the messages are staying in memory longer. It can also indicate that the Propagation or Apply Process is slow to consume the enqueued messages.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
10*g*, 11*g*, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Spilled messages for Capture process %CAPTURE_NAME% queue is %value% percent.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each Capture Name object. If warning or critical threshold values are currently set for any Capture Name object, those thresholds can be viewed on the Metric Detail page for this metric. To specify or change warning or critical threshold values for each Capture Name object, use the Edit Thresholds page.

Data Source

The source of this metric is the target database in the gv$buffered_queues table view.

User Action

Increase Streams Pool Size to avoid queue spills.

Streams Latency and Throughput

The metrics in the Streams Latency and Throughput metrics category collect information about latency and throughput for each capture, propagation and apply component in the database. Latency and throughput are important indicators for the overall performance of the streams path.

Latency

High Latency indicates that the components are slow.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
10*g*R2, 11*g*, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Latency for Streams %streams_process_type% Process %streams_process_name% is %value% seconds.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects.

If warning or critical threshold values are currently set for any unique combination of Streams Process Name and Streams Process Type objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects, use the Edit Thresholds page.

Data Source

The data source for this metric is the target database in the gv$streams_capture, gv$propagation_sender, and gv$streams_apply_server views.

User Action

Identify and correct the least performing component in the streams configuration.

Throughput (per sec)

This metric collects information about throughput for each capture, propagation and apply component in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
10*g*R2, 11*g*, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Throughput for Streams %streams_process_type% Process %streams_process_name% is %value% messages/sec.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects.

If warning or critical threshold values are currently set for any unique combination of Streams Process Name and Streams Process Type objects, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each unique combination of Streams Process Name and Streams Process Type objects, use the Edit Thresholds page.

Data Source

Not available.

User Action

The required actions are specific to your site.

Streams Processes Count

The metrics in this metric category show the total number of Streams capture processes, propagations, and apply processes at the local database. This metric also shows the number of capture processes, propagations, and apply processes that have encountered errors.

Apply Processes Having Errors

This metric shows the number of apply processes that have encountered errors at the local database.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 10 Minutes

Data Source

The information in this metric is in the `DBA_APPLY` data dictionary view.

User Action

If an apply process has encountered errors, then correct the conditions that caused the errors.

Capture Processes Having Errors

This metric shows the number of capture processes that have encountered errors at the local database.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 10 Minutes

Data Source

The information in this metric is in the `DBA_CAPTURE` data dictionary view.

User Action

If a capture process has encountered errors, then correct the conditions that caused the errors.

Number of Apply Processes

This metric shows the number of apply processes at the local database.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 10 Minutes

Data Source

The information in this metric is in the `DBA_APPLY` data dictionary view.

User Action

Use this metric to determine the total number of apply processes at the local database.

Number of Capture Processes

This metric shows the number of capture processes at the local database.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 10 Minutes

Data Source

The information in this metric is in the `DBA_CAPTURE` data dictionary view.

User Action

Use this metric to determine the total number of capture processes at the local database.

Number of Propagation Jobs

This metric shows the number of propagations at the local database.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 10 Minutes

Data Source

The information in this metric is in the `DBA_PROPAGATION` data dictionary view.

User Action

Use this metric to determine the total number of propagations at the local database.

Propagation Errors

This metric shows the number of propagations that have encountered errors at the local database.

Target Version	Collection Frequency
10*g*, 11*g*, 12*c*	Every 10 Minutes

Data Source

The information in this metric is in the `DBA_PROPAGATION` data dictionary view.

User Action

If a propagation has encountered errors, then correct the conditions that caused the errors.

Streams Propagation - Message State Stats

The metrics in this metric category collect the number of messages in Ready and Waiting state for each Propagation process.

Streams Prop - (%)Messages in Waiting State

This metric collects the number of messages in Ready state for each Propagation process.

Target Version	Collection Frequency
9*i*R2, 10*g*, 11*g*, 12*c*	Every 30 Minutes

Data Source

The source of the data for this metric is the target database in the source and destination queues.

User Action

No user action is required.

Suspended Session

The metrics in this metric category contain the metrics that represent the number of resumable sessions that are suspended due to a correctable error.

Suspended Session Count

This metric represents the number of resumable sessions currently suspended in the database.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
9*i*	Every 5 Minutes	0	Not Defined	%value% session(s) are suspended.

Data Source

This metric is calculated with the following command:

```
SELECT count(*)
  FROM v$resumable
  WHERE status = 'SUSPENDED' and enabled = 'YES'
```

User Action

Query the v$resumable view to see what the correctable errors are that are causing the suspension. The method to correct each error depends on the nature of the error.

Tablespace Allocation

The metrics in this metric category check the amount of space used and the amount of space allocated to each tablespace. The used space can then be compared to the allocated space to determine how much space is unused in the tablespace. This metric is intended for reporting, rather than alerts. Historical views of unused allocated free space can help DBAs to correctly size their tablespaces, eliminating wasted space.

Tablespace Allocated Space (MB)

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

The allocated space of a tablespace is the sum of the current size of its data files. A portion of this allocated space is used to store data while some may be free space. If segments are added to a tablespace, or if existing segments grow, they will use the allocated free space. The allocated free space is only available to segments within the tablespace. If, over time, the segments within a tablespace are not using this free space, the allocated free space is not being used.

This metric calculates the space allocated for each tablespace. It is not intended to generate alerts. Rather it should be used in conjunction with the Allocated Space Used (MB) metric to produce a historical view of the amount of space being used and unused by each tablespace.

Data Source

Tablespace Allocated Space (MB) is calculated by looping though the tablespaces data files and totalling the size of the data files.

Tablespace Used Space (MB)

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

The allocated space of a tablespace is the sum of the current size of its data files. Some of this allocated space is used to store data, and some of it may be free space. If segments are added to a tablespace, or if existing segments grow, they will use the allocated free space. The allocated free space is only available to segments within the tablespace. If, over time, the segments within a tablespace are not using this free space, then the allocated free space is being wasted.

This metric calculates the space used for each tablespace. It is not intended to generate alerts. Rather it should be used in conjunction with the Tablespace Allocated Space (MB) metric to produce a historical view of the amount of space being used and unused by each tablespace.

Data Source

Tablespace Used Space (MB) is derived from Tablespace Allocated Space (MB) Tablespace Allocated Free Space (MB) where:

Tablespace Allocated Space (MB) is calculated by looping through the tablespaces data files and totaling the size of the data files.

Tablespace Allocated Free Space (MB) is calculated by looping through the tablespaces data files and totaling the size of the free space in each data file.

Tablespaces Full

The metrics in this metric category check for the amount of space used by each tablespace. The used space is then compared to the available free space to determine tablespace fullness. The available free space accounts for the maximum data file size as well as available disk space. This means that a tablespace will not be flagged as full if data files can extend and there is enough disk space available for them to extend.

Tablespace Free Space (MB)

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

As segments within a tablespace grow, the available free space decreases. If there is no longer any available free space, meaning data files reached their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

This metric checks for the total available free space in each tablespace. This metric is intended for larger tablespaces, where the Available Space Used (%) metric is less meaningful. If the available free space falls below the size specified in the threshold arguments, then a warning or critical alert is generated.

If the version of the monitored database target is Oracle Database 10*g* Release 1 or later and the tablespace uses Local Extent Management, then the Oracle Database Server evaluates this metric internally every 10 minutes. Alternatively, if the version of the monitored Database target is Oracle 9*i* or earlier, or the tablespace uses Dictionary Extent Management, then the Oracle Management Agent tests the value of this metric every 30 minutes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
8*i*, 9*i*	Every 30 Minutes	Not Defined	Not Defined	Tablespace [%name%] has [%value% mbytes] free
10*g*, 11*g*, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Generated By Database Server

Data Source

The source of the data for this metric is MaximumSize Total Used Space where:

- TotalUsedSpace: Total used space in MB of tablespace.
- MaximumSize: Maximum size (in MB) of the tablespace. The maximum size is determined by looping through the tablespaces data files, as well as additional free space on the disk that would be available for the tablespace should a data file autoextend.

User Action

Perform one of the following:

- Increase the size of the tablespace by: Enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.
- If the tablespace is suffering from tablespace free space fragmentation problems, consider reorganizing the entire tablespace.
- Relocate segments to another tablespace, thereby increasing the free space in this tablespace.
- Run the Segment Advisor on the tablespace.

Tablespace Space Used (%)

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

As segments within a tablespace grow, the available free space decreases. If there is no longer any available free space, meaning data files have reached their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

This metric checks the Available Space Used (%) for each tablespace. If the percentage of used space is greater than the values specified in the threshold arguments, then a warning or critical alert is generated.

If the version of the monitored database target is Oracle Database 10*g* Release 1 or later and the tablespace uses Local Extent Management, then the Oracle Database Server evaluates this metric internally every 10 minutes. Alternatively, if the version of the monitored Database target is Oracle 9*i* or earlier, or the tablespace uses Dictionary Extent Management, then the Oracle Management Agent tests the value of this metric every 30 minutes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
8*i*, 9*i*	Every 30 Minutes	85	97	Tablespace [%name%] is [%value% percent] full

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
10*g*, 11*g*, 12*c*	Every 30 Minutes	85	97	Generated By Database Server

Data Source

This metric is calculated with the following command where:

- TotalUsedSpace: total used space in MB of tablespace.
- MaximumSize: Maximum size (in MB) of the tablespace. The maximum size is determined by looping through the tablespaces data files, as well as additional free space on the disk that would be available for the tablespace should a data file autoextend.

(TotalUsedSpace / MaximumSize) * 100

For additional information about the data source, refer to the fullTbsp.pl Perl script located in the sysman/admin/scripts directory.

User Action

Perform one of the following:

- Increase the size of the tablespace by: Enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.
- If the tablespace is suffering from tablespace free space fragmentation problems, consider reorganizing the entire tablespace.
- Relocate segments to another tablespace, thus increasing the free space in this tablespace.
- Run the Segment Advisor on the tablespace.

Tablespaces Full (dictionary managed)

The metrics in this metric category check for the amount of space used by each tablespace. The used space is then compared to the available free space to determine tablespace fullness. The available free space accounts for the maximum data file size as well as available disk space. This means that a tablespace will not be flagged as full if data files can extend, and there is enough disk space available for them to extend.

Tablespace Free Space (MB) (dictionary managed)

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

As segments within a tablespace grow, the available free space decreases. If there is no longer any available free space, meaning data files have reached their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

This metric checks for the total available free space in each tablespace. This metric is intended for larger tablespaces, where the Available Space Used (%) metric is less meaningful. If the available free space falls below the size specified in the threshold arguments, then a warning or critical alert is generated.

If the version of the monitored database target is Oracle Database 10*g* Release 1 or later and the tablespace uses Local Extent Management, then the Oracle Database Server evaluates this metric internally every 10 minutes. Alternatively, if the version of the monitored Database target is Oracle 9*i* or earlier, or the tablespace uses Dictionary Extent Management, then the Oracle Management Agent tests the value of this metric every 30 minutes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
10*g*, 11*g*, 12*c*	Every 30 Minutes	Not Defined	Not Defined	Tablespace [%name%] has [%value% mbytes] free.

Data Source

The source of the data for this metric is MaximumSize Total Used Space where:

- TotalUsedSpace: Total used space in MB of tablespace.
- MaximumSize: Maximum size (in MB) of the tablespace. The maximum size is determined by looping through the tablespaces data files, as well as additional free space on the disk that would be available for the tablespace should a data file autoextend.

Tablespace Space Used (%) (dictionary managed)

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

As segments within a tablespace grow, the available free space decreases. If there is no longer any available free space, meaning data files have reached their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

This metric checks the Available Space Used (%) for each tablespace. If the percentage of used space is greater than the values specified in the threshold arguments, then a warning or critical alert is generated.

If the version of the monitored database target is Oracle Database 10*g* Release 1 or later and the tablespace uses Local Extent Management, then the Oracle Database Server evaluates this metric internally every 10 minutes. Alternatively, if the version of the monitored Database target is Oracle 9*i* or earlier, or the tablespace uses Dictionary Extent Management, then the Oracle Management Agent tests the value of this metric every 30 minutes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
10*g*, 11*g*, 12c	Every 30 Minutes	85	97	Tablespace [%name%] is [%value% percent] full

Data Source

The source of the data for this metric is (TotalUsedSpace / MaximumSize) * 100 where:

- TotalUsedSpace: Total used space in MB of tablespace.
- MaximumSize: Maximum size (in MB) of the tablespace. The maximum size is determined by looping through the tablespaces data files, as well as additional free space on the disk that would be available for the tablespace should a data file autoextend.

User Action

Perform one of the following:

- Increase the size of the tablespace by: Enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.
- If the tablespace is suffering from tablespace free space fragmentation problems, consider reorganizing the entire tablespace.
- Relocate segments to another tablespace, thereby increasing the free space in this tablespace.
- Run the Segment Advisor on the tablespace.

Tablespaces With Problem Segments

The metrics in this metric category check for the following:

- The largest chunk-free space in the tablespace. If any table, index, cluster, or rollback segment within the tablespace cannot allocate one additional extent, then an alert is generated.
- Whether any of the segments in the tablespace are approaching their maximum extents. If, for any segment, the maximum number of extents minus the number of existing extents is less than 2, an alert is generated.

Only the tablespaces with problem segments are returned as results.

Segments Approaching Maximum Extents Count

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

This metric checks for segments nearing the upper limit of the number of maximum extents. If the number of segments is greater than the values specified in the threshold arguments, a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	0	Not Defined	%value% segments in %name% tablespace approaching max extents.

Data Source

The source of the data for this metric is the number of segments for which the maximum number of extents minus the number of existing extents is less than 2.

For additional information about the data source, refer to the problemTbsp.pl Perl script located in the sysman/admin/scripts directory.

User Action

If possible, increase the value of the segments MAXEXTENTS storage parameter. Otherwise, rebuild the segment with a larger extent size ensuring the extents within a segment are the same size by using a locally managed tablespace. For a dictionary managed tablespace, specify STORAGE parameters where NEXT=INITIAL and PCTINCREASE = 0.

Segments Not Able to Extend Count

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

This metric checks for segments that cannot allocate an additional extent. If the number of segments is greater than the values specified in the threshold arguments, a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	0	Not Defined	%value% segments in %name% tablespace unable to extend.

Data Source

After checking for the largest chunk free space in the tablespace, this is the number of segments that cannot allocate an additional extent.

For additional information about the data source, refer to the problemTbsp.pl Perl script located in the sysman/admin/scripts directory.

User Action

Perform one of the following:

- Increase the size of the tablespace by enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.
- If the tablespace is suffering from tablespace free space fragmentation problems, consider reorganizing the entire tablespace.
- Relocate segments to another tablespace, thereby increasing the free space in this tablespace.

Temporary File Status

This metric category contains the Temporary File Status metric.

Temporary File Id

The absolute file number of the temporary file, used to join with other database tables and views to retrieve additional information.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
9*i*, 10*g*, 11*g*, 12*c*	Every 15 Minutes	OFFLINE	Not Defined	The temporary file %NAME% is %STATUS%.

Top Wait Events

This section provides information on the metrics in the Top Wait Events category.

Target Version	Evaluation and Collection Frequency
All versions	Every 15 minutes

Metric Name	Description
Average Foreground Wait Time (millisecond)	The average foreground wait time, in milliseconds.
Average Wait Time (millisecond)	The average wait time, in milliseconds.
Total Foreground Wait Time (second)	The total foreground wait time, in seconds.
Total Number of Foreground Waits	The total number of foreground waits.
Total Number of Waits	The total number of waits.
Total Wait Time (second)	The total wait time, in seconds.
Wait Class Name	The name of the wait class.

Total Objects by Schema

The metrics in this metric category contain the metric that provides the number of database objects in a schema.

Total Object Count

This metric displays the total number of database objects in a schema.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every 24 Hours	Not Defined	Not Defined	%value% object(s) exist in the %owner% schema.

Total Tables by Schema

The metrics in this metric category provide the number of tables in a schema.

Total Table Count

This metric displays the total number of tables in a schema.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every 24 Hours	Not Defined	Not Defined	%value% table(s) exist in the %owner% schema.

Unusable Indexes

This metric category represents the number of unusable indexes in the database.

Unusable Index Count

This metric represents the total unusable index count in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 hours	Not Defined	Not Defined	Unusable Index Count in the database is %value%

Data Source

The data is derived from the dba_indexes, dba_ind_partitions, and dba_ind_subpartitions views.

User Action

The “Rebuild Unusable Indexes” corrective action could be setup against the incident to automatically attempt to rebuild the unusable indexes in the database. This lets the user to specify various rebuild options and the schemas in which the indexes should be rebuilt. In addition, the incident also enables the user to rebuild the unusable indexes from the Enterprise Manager console.

Unusable Indexes by Schema

This metric category represents the number of unusable indexes in each schema.

Unusable Index Count by Schema

This metric represents the total number of unusable indexes per schema.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 hours	Not Defined	Not Defined	Unusable Index Count in %Unusable_Index_owner% schema is %value%

Multiple Thresholds

Different warning and critical threshold values could be set for each Unusable Index Owner (schema) object.

If warning or critical threshold values are currently set for any Unusable Index Owner object, those thresholds could be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each Unusable Index Owner object, use the Edit Thresholds page.

Data Source

The data is derived from the dba_indexes, dba_ind_partitions, and dba_ind_subpartitions views.

User Action

The “Rebuild Unusable Indexes” corrective action could be setup against the incident to automatically attempt to rebuild the unusable indexes in each Unusable Index Owner (schema) object. This lets the user to specify various rebuild options that should be used for the operation. In addition, the incident also enables the user to rebuild the unusable indexes from the Enterprise Manager console.

User Block

The metrics in this metric category contain the metrics that tell to what extent, and how consistently, a given session is blocking multiple other sessions.

Blocking Session Count

For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

This metric signifies that a database user is blocking at least one other user from performing an action, such as updating a table. An alert is generated if the number of consecutive blocking occurrences reaches the specified value. The sessions being blocked can come from different instances.

> **Note:**
>
> The catblock.sql script needs to be run on the managed database prior to using the User Blocks test. This script creates some additional tables, view, and public synonyms that are required by the User Blocks test.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every 5 Minutes	0	Not Defined	Session %sid% blocking %value% other sessions for all instances.

Data Source

This metric is calculated using the following command:

```
SELECT blocking_sid, num_blocked
  FROM ( SELECT blocking_sid, SUM(num_blocked) num_blocked
  FROM ( SELECT l.id1, l.id2,
                MAX(DECODE(l.block, 1, i.instance_name||'-'||l.sid,
                2, i.instance_name||'-'||l.sid, 0 )) blocking_sid,
                SUM(DECODE(l.request, 0, 0, 1 )) num_blocked
           FROM gv$lock l, gv$instance i
           WHERE ( l.block!= 0 OR l.request > 0 ) AND
                   l.inst_id = i.inst_id
                   GROUP BY l.id1, l.id2)
  GROUP BY blocking_sid
  ORDER BY num_blocked DESC)
  WHERE num_blocked != 0
```

User Action

Either have the user who is blocking other users rollback the transaction, or wait until the blocking transaction has been committed.

User Locks

The metrics in this metric category provide information regarding user locks.

Enterprise Manager will issue the alert when the Maximum Blocked Session Count or maximum blocked DB time (seconds) of transactional locks: TM, TX, UL reach the threshold.

Maximum Blocked DB Time (seconds)

This metric represents the maximum time wasted in any given lock chain, not for the total time wasted by everyone in any lock chain.

Target Version	Key	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
9*i*, 10*g*, 11*g*, 12*c*	lockType = TM	Every 10 Minutes	Not Defined	Not Defined	%value% seconds in DB Time is spent waiting for %lockType% lock.
9*i*, 10*g*, 11*g*, 12*g*	lockType = TX	Every 10 Minutes	Not Defined	Not Defined	%value% seconds in DB Time is spent waiting for %lockType% lock.
9*i*, 10*g*, 11*g*, 12*c*	lockType = UL	Every 10 Minutes	Not Defined	Not Defined	%value% seconds in DB Time is spent waiting for %lockType% lock.

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each User Lock Type object.

If warning or critical threshold values are currently set for any User Lock Type object, those thresholds can be viewed on the Metric Detail page for this metric.

To specify or change warning or critical threshold values for each User Lock Type object, use the Edit Thresholds page.

Data Source

The data for the metric is retrieved from database view gv$session.

User Action

You can set the threshold for warning alert or critical alert for maximum Blocked DB Time (seconds). When maximum time wasted in any given lock chain reaches the threshold, Enterprise Manager will issue the alert.

Maximum Blocked Session Count

This metric represents the maximum length of any lock chain, not for the total number of people stuck in lock chains.

Target Version	Key	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
9*i*, 10*g*, 11*g*, 12*c*	lockType = TM	Every 10 Minutes	Not Defined	Not Defined	%value% sessions are blocked by %lockType% lock.
9*i*, 10*g*, 11*g*, 12*c*	lockType = TX	Every 10 Minutes	Not Defined	Not Defined	%value% sessions are blocked by %lockType% lock.
9*i*, 10*g*, 11*g*, 12*c*	lockType = UL	Every 10 Minutes	Not Defined	Not Defined	%value% sessions are blocked by %lockType% lock.

3
Far Sync Instance

Oracle Enterprise Manager collects a subset of the Oracle Database Instance metrics for Far Sync instances (remote Oracle Data Guard destinations). The Far Sync metrics fall in the following categories:

- Archive Area
- Data Failure
- Data Guard Performance
  - Transport Lag (seconds)
  - Transport Lag Data Refresh Time
- Data Guard Status
- Dump Area
- Fast Recovery
- Incident
- Operational Error
- Response

4
Listener

This chapter provides information about the listener metrics.
For each metric, it provides the following information:

- Description
- Metric table

  The metric table can include some or all of the following: target version, default collection frequency, default warning threshold, default critical threshold, and alert text.

You can use Enterprise Manager to manage Oracle listener targets. From the Enterprise Manager Listener home page, you can monitor key metrics that can help determine the performance and availability of the listener and help you troubleshoot potential performance problems.

General Status

This metric category contains a set of metrics that provide general information about the listener target. For more information, see the section on Listener Administration in the *Oracle Database Net Services Administrator's Guide*.

Alias

This metric provides the alternative name for the listener. On the Metric Detail page, you can see the value of this metric only when you select one of the Real Time refresh options. The alias also appears on the Listener home page.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The data is derived from the STATUS command of the Listener Control Utility.

Security

This metric shows whether or not a password is required to run specific commands with the Listener Control Utility.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The data is derived from the STATUS command of the Listener Control Utility.

SID List

This metric lists the System Identifiers (SIDs) for the services monitored by the listener.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The list of SIDs for the listener is stored in the `listener.ora` configuration file.

SNMP Status

This metric indicates whether or not the listener can respond to queries from an SNMP-based network management system.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The data is derived from the STATUS command of the Listener Control Utility.

Start Date

This metric represents the date and time when the listener was last started. On the Metric Detail page, you can see the value of this metric only when you select one of the Real Time refresh options. This metric also appears on the Enterprise Manager Listener home page.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The data is derived from the STATUS command of the Listener Control Utility.

TNS Address

This metric displays the protocol, host, and port information for the listener. On the Metric Detail page, you can see the value of this metric only when you select one of the Real Time refresh options. The TNS address also appears on the Listener home page.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The TNS address of the Listener is defined in the `listener.ora` configuration file.

Trace Level

This metric represents the level of tracing currently enabled for the listener. Tracing can be used to troubleshoot problems with the listener by saving additional information to the trace file. For more information about the trace levels you can set for the listener, see the information about the Listener Control Utility in the *Oracle Database Net Services Reference Guide 10g Release 2 (10.2)*.

On the Metric Detail page, you can see the value of this metric only when you select one of the Real Time refresh options.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The data is derived from the STATUS command of the Listener Control Utility.

Version

This metric provides the version of the listener software. On the Metric Detail page, you can see the value of this metric only when you select one of the Real Time refresh options. This metric also appears on the Enterprise Manager Listener home page.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The data is derived from the STATUS command of the Listener Control Utility.

Listener Ports

This metric category collects configuration data for the listener such as ports, protocol, and host. It is calculated using the `lsnrctl status <listener Name>` command on all platforms.

Listener Services

This metric category collects configuration data for the services registered to the listener (for example, service name, SID, or service).

It is calculated using the lsnrctl status `<listener Name>` command on all platforms.

Load

This metric category contains a set of metrics that provide information about the number of connections supported by the listener over a period of time. For more information, see the section on Listener Administration in the *Oracle Database Net Services Administrator's Guide*.

Connections Established

This metric provides the number of connections established since the listener was last started.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The data is derived from the `SERVICES` command of the Listener Control Utility.

User Action

If you are experiencing performance issues with the database or other services supported by the listener, review the historical values of this metric to determine whether or not the performance problems are caused by excessive load on the listener or host.

Connections Established (per min)

This metric reports the average number of connections per minute that were established with the listener.

Data Source

The data is derived from the Listener Control Utility.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

User Action

If you are experiencing performance issues with the database or other services supported by the listener, review the historical values of this metric to determine whether or not the performance problems are caused by excessive load on the listener or host.

Connections Refused

This metric reports the number of connections to the listener that were refused. A connection can be refused for a variety of reasons, including situations where the database or other listener service is down, or if the connection timed out.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The data is derived from the `SERVICES` command of the Listener Control Utility.

User Action

If Enterprise Manager reports a high number of refused connections, check the availability and performance of the database or other services supported by the listener.

Connections Refused (per min)

This metric reports the average number of connections that were refused per minute.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The data is derived from the Listener Control Utility.

User Action

If Enterprise Manager reports a high number of refused connections, check the availability and performance of the database or other services supported by the listener.

Response

This metric category contains the response and status metrics that provide performance information about the Listener.

Response Time (msec)

This metric represents the time (in milliseconds) that it takes for the Listener to respond to a network request (ping).

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every 5 Minutes	400	1000	Listener response to a TNS ping is %value% msecs.

Data Source

The value of this metric is derived using the `TNSPING` command. For more information about the `TNSPING` command, see the *Oracle Database Net Services Administrator's Guide*.

User Action

If the listener response time consistently exceeds the threshold, then there can be a number of possible causes, such as slow DNS resolution, network congestion, or other network-specific factors. For more information about investigating these issues, see the My Oracle Support Note, *Oracle Net Performance Tuning* (Doc ID 67983.1):

https://support.oracle.com/epmos/faces/DocumentDisplay?id=67983.1

Status

This metric returns a value of 1 if the listener is up and running. It returns a 0 if the listener is unavailable.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every 5 Minutes	Not Defined	0	The listener is down: %oraerr%.

Data Source

The data is derived from the from the `STATUS` command in the Listener Control Utility. For more information, see the *Oracle Database Net Services Administrator's Guide*.

User Action

When the listener is down, users cannot access the database or other services on this host. Review the troubleshooting information in *Oracle Database Net Services Administrator's Guide*.

TNS Errors

This metric category contains metrics that perform incremental scanning of listener log files for security errors. For releases earlier than Oracle Database Plug-in Release 12.1.0.4, these error codes are predefined and can be any of the following:

- 1169
- 1189
- 1190
- 12508

TNSMsg

This metric reports the TNS error message.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
Not Available	Every Hour	Not Defined	TNS-[ ]*0*(1169|1189|12508|1190)	%tnserr% error occured.Please check log for details.

5
Pluggable Database

This chapter provides information about the Pluggable Database (PDB) metrics.
For each metric, it provides the following information:

- Description
- Metric table

  The metric table can include some or all of the following: target version, default collection frequency, default warning threshold, default critical threshold, and alert text.

Database Feature Usage

This metric category provides details on database feature usage metrics.

Count

This column stores feature-specific usage data in number format.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The AUX_COUNT column from the CDB_FEATURE_USAGE_STATISTICS view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

Currently Used

Indicates whether usage was detected the last time the system checked (TRUE) or not (FALSE)

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The CURRENTLY_USED column from the CDB_FEATURE_USAGE_STATISTICS view.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

DBID

Database identifier of the database being tracked

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The DBID column from the CDB_FEATURE_USAGE_STATISTICS view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

Detected Usages

Number of times the system has detected usage for the feature

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The DETECTED_USES column from the CDB_FEATURE_USAGE_STATISTICS view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

Feature Info

This column stores feature-specific usage data in character format.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The FEATURE_INFO column from the CDB_FEATURE_USAGE_STATISTICS view.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

Feature Name

Name of the feature

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The NAME column from the CDB_FEATURE_USAGE_STATISTICS view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

First Usage Date

First sample time the system detected usage of the feature

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The FIRST_USAGE_DATE column from the CDB_FEATURE_USAGE_STATISTICS view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

Last Sample Date

Amount of time (in seconds) between the last two usage sample times

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The LAST_SAMPLE_DATE column from the CDB_FEATURE_USAGE_STATISTICS view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

Last Sample Period

Amount of time (in hours) between the last two usage sample times

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The LAST_SAMPLE_PERIOD column from the CDB_FEATURE_USAGE_STATISTICS view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

Last Usage Date

Last sample time the system detected usage of the feature

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The LAST_USAGE_DATE column from the CDB_FEATURE_USAGE_STATISTICS view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

Total Samples

Number of times the system has woken up and checked for feature usage

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The TOTAL_SAMPLES column from the CDB_FEATURE_USAGE_STATISTICS view.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

Version

Database version in which the feature was tracked

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The VERSION column from the CDB_FEATURE_USAGE_STATISTICS view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information about which database features are being used.

Datafiles

This metric category provides details on the datafile metrics.

Autoextensible

Autoextensible indicator.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The AUTOEXTENSIBLE column of the CDB_DATA_FILES view if a permanent datafile or the AUTOEXTENSIBLE column of the CDB_TEMP_FILES view if a temporary file.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Datafile Name

Name of the database file or temporary file.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The FILE_NAME column of the CDB_DATA_FILES view if a permanent datafile. The FILE_NAME column of the CDB_TEMP_FILES view if a temporary file.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

File Size

Size of the file in bytes.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The BYTES column of the CDB_DATA_FILES view if a permanent datafile or the BYTES column of the CDB_TEMP_FILES view if a temporary file.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Initial File Size

Creation size of the file (in bytes)

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The CREATE_BYTES column of the V$DATAFILE view if a permanent datafile or the CREATE_BYTES column of the V$TEMPFILE view if a temporary file.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Increment By

Number of tablespace blocks used as autoextension increment. Block size is contained in the BLOCK_SIZE column of the CDB_TABLESPACES view.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The INCREMENT_BY column of the CDB_DATA_FILES view if a permanent datafile or the INCREMENT_BY column of the CDB_TEMP_FILES view if a temporary file.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Max File Size

Maximum file size in bytes

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The MAXBYTES column of the CDB_DATA_FILES view if a permanent datafile or the MAXBYTES column of the CDB_TEMP_FILES view if a temporary file.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Status

The ONLINE status of the database file (one of SYSOFF, SYSTEM, OFFLINE, ONLINE, RECOVER) or the status of the temporary file (one of OFFLINE, ONLINE, UNKNOWN).

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The ONLINE_STATUS column of the CDB_DATA_FILES view if a permanent datafile or the STATUS column of the CDB_TEMP_FILES view if a temporary file.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Storage Entity

The filesystem or the raw device used by this datafile or temporary file.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

Perl script utility to retrieve the full file path.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Tablespace

Name of the tablespace to which the file belongs

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The TABLESPACE column of the CDB_DATA_FILES view if a permanent datafile or the TABLESPACE column of the CDB_TEMP_FILES view if a temporary file.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Database Job Status

The metrics in this category represent the health of database jobs registered through the DBMS_SCHEDULER interface.

Broken Job Count

The Oracle Server job queue is a database table that stores information about local jobs such as the PL/SQL call to execute for a job such as when to run a job. Database replication is also managed by using the Oracle job queue mechanism using jobs to push deferred transactions to remote master sites, to purge applied transactions from the deferred transaction queue or to refresh snapshot refresh groups.

A job can be broken in two ways:

- Oracle failed to successfully execute the job after a specified number of attempts (defined in the job).
- The job is explicitly marked as broken by using the procedure DBMS_JOB.BROKEN.

This metric checks for broken DBMS jobs. A critical alert is generated if the number of broken jobs exceeds the value specified by the threshold argument.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	0	Not Defined	%value% job(s) are broken.

Failed Job Count

The Oracle Server job queue is a database table that stores information about local jobs such as the PL/SQL call to execute for a job such as when to run a job. Database replication is also managed by using the Oracle job queue mechanism using jobs to push deferred transactions to remote master sites, to purge applied transactions from the deferred transaction queue or to refresh snapshot refresh groups.

If a job returns an error while Oracle is attempting to execute it, the job fails. Oracle repeatedly tries to execute the job doubling the interval of each attempt. If the job fails after a specified number of times (specified in the job definition), Oracle automatically marks the job as broken and no longer tries to execute it.

This metric checks for failed DBMS jobs. An alert is generated if the number of failed job exceeds the value specified by the threshold argument.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	0	Not Defined	%value% job(s) have failed.

Database Scheduler Jobs

This section provides information on the metrics in the Database Scheduler Jobs category, which report the current status of DBMS jobs registered through the DBMS_SCHEDULER interface. Using these metrics, you can monitor long running jobs and obtain alerts on individual jobs.

Elapsed Running Time (in Minutes)

The duration of time the current DBMS job has been running, in minutes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	> 5 Minutes	> 30 Minutes	DBMS job %job_name% for %owner% has been running for %value% minutes.

Failure Count

The number of times the DBMS job has failed during the last collection period.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	> 0	Not Defined	DBMS job %job_name% for %owner% has failed %value% time(s) during last collection period.

State

The current state of the DBMS job.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 5 Minutes	DISABLED	BROKEN	DBMS job %job_name% for %owner% is in %state% state.

Database Services

The metrics in this category include the service CPU time and service response time.

Service CPU Time (per user call) (microseconds)

This metric represents the average CPU time, in microseconds, for calls to a particular database service.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	CPU per call for service %keyValue% is %value% microseconds

Service Response Time (per user call) (microseconds)

This metric represents the average elapsed time, in microseconds, for calls to a particular database service.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Elapsed time per call for service %keyValue% is %value% microseconds

Datafile Allocation

This section provides information on the metrics in the Datafile Allocation category.

The allocated space is the current size of the datafile. A portion of this allocated space is used to store data while some may be free space. The metrics in this category check the amount of space used and the amount of space allocated to each datafile. The used space can then be compared to the allocated space to determine how much space is unused in the datafile. This metric is not intended for alerts. Rather it is intended for reporting. Historical views of unused allocated free space can help DBAs to correctly size their datafiles, eliminating wasted space.

Target Version	Evaluation and Collection Frequency
All versions	Every 24 hours

Metric Name	Description	Data Source
Allocated File Size (MB)	The space allocated to the datafile. This metric should be used in conjunction with the Used File Size (MB) metric to produce a historical view of the amount of space being used and unused in each datafile.	• Datafile: dba_data_files • Tempfile: dba_temp_files

Metric Name	Description	Data Source
Used File Size (MB)	The space used in the datafile. This metric should be used in conjunction with the Allocated File Size (MB) metric to produce a historical view of the amount of space being used and unused in each datafile.	• Datafile: Subtract allocated free space (dba_lmt_free_space and dba_dmt_free_space) from allocated file size (dba_data_files) • Tempfile: gv$temp_extent_pool.bytes_used

Failed Logins

The metrics in this category check for the number of failed logins on the target database. This check is performed every interval specified by the collection frequency and returns the number of failed logins for the last 30 minutes. These metrics will only work for databases where the audit_trail initialization parameter is set to DB or XML and the session is being audited.

Failed Login Count

This metric checks for the number of failed logins on the target database. This check is performed every interval specified by the collection frequency and returns the number of failed logins for the last 30 minutes. This metric will only work for databases where the audit_trail initialization parameter is set to DB or XML and the session is being audited.

If the failed login count crosses the values specified in the threshold arguments, then a warning or critical alert is generated. Because it is important to know every time a significant number of failed logins occurs on a system, on every collection, this metric determines the number of failed login attempts in the last 30 minutes and overrides the current alert instead of a new alert. You can manually clear these alerts.They will not automatically cleared after the next collection.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 30 Minutes	150	300	Number of failed login attempts exceeds threshold value.

Invalid Objects

This metric category contains the metrics associated with invalid objects.

Total Invalid Object Count

This metric represents the total invalid object count.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined*	%value% object(s) are invalid in the database.

Invalid Objects by Schema

This metric category contains the metrics that represent the number of invalid objects in each schema.

Owner's Invalid Object Count

This metric represents the invalid object count by owner.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	%value% object(s) are invalid in the %owner% schema in pluggable database %pdbName%

Data Source

For each metric index:

```
SELECT count(1)
```

User Action

View the status of the database objects in the schema identified by the Invalid Object Owner metric. Recompile objects as necessary.

Messages per buffered queue

The metrics in this category monitor the age and state of the first (top of the queue) message for each buffered queue in the database except for the system queues. Queues that are in the schema of SYS, SYSTEM, DBSNMP, and SYSMAN are defined as system level queues.

Average age of messages per buffered queue (seconds)

This metric provides the average age (in seconds) of the messages in the buffered queue for all nonsystem queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Average age of messages in %schema%.%queue_name% queue is %value% seconds.

Spilled Messages

This metric displays the current number of overflow messages spilled to disk from the buffered queue.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Current number of overflow messages spilled to disk from the buffered queue %schema%.%queue_name% is %value%

First message age in the buffered queue per queue (seconds)

This metric gives the age (in seconds) of the first message in the buffered queue for all non-system queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Age of first message in %schema%.%queue_name% buffered queue is %value% seconds.

Messages processed per buffered queue (%)

This metric gives the messages processed percentage per minute per buffered queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Messages processed for queue %schema%.%queue_name% is %value% percent.

Total Messages Processed per Buffered Queue per Minute

This metric gives the total number of messages processed per minute per buffered queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Total messages processed per minute in the last interval for queue %schema%.%queue_name% is %value%

Total messages received per buffered queue per minute

This metric gives the total number of messages received or enqueued into the buffered queue per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Total messages processed per minute in the last interval for queue %schema%.%queue_name% is %value%

Total number of messages processed

This metric gives the total number of messages processed in the last collection interval of the metric.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Total number of messages received

This metric gives the total number of messages received in the last collection interval of the metric.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Messages per buffered queue per subscriber

This metric category monitors the messages for buffered queues per subscriber in the database.

Average age of messages/buffered queue/subscriber (seconds)

This metric display's the average age of messages in the buffered queue per queue in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Average age of messages for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

First message age in buffered queue per subscriber (seconds)

This metric displays the age of the first message in the buffered queue per queue per subscriber in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Age of first message for subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

Messages processed/buffered queue/subscriber (%)

This metric gives the messages processed percentage for the buffered queue per subscriber. Messages processed percent is calculated as the percent of the total number messages processed or dequeued to the total number of messages received or enqueued.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Messages processed for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% percent.

Messages processed/buffered queue/subscriber per minute (%)

This metric gives the total number of messages processed per minute per buffered queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value.

Total messages processed/buffered queue/subscriber per minute

This metric gives the total number of messages processed per minute per buffered queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Total messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value%

Total messages received/buffered queue/subscriber per minute

This metric gives the total number of messages received or enqueued into the queue per subscriber per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Total messages received per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value%

Total number of messages processed

This metric gives the total number of messages processed in the last collection interval of the metric.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Total number of messages received

This metric gives the total number of messages received in the last collection interval of the metric.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Messages per persistent queue

The metrics in this category monitor the age and state of the first (top of the queue) message for each persistent queue in the database except for the system queues. Queues that are in the schema of SYS, SYSTEM, DBSNMP, and SYSMAN are defined as system level queues.

Average age of messages per persistent queue (seconds)

This metric displays the average age of messages in the persistent queue per queue in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Average age of messages in %schema%.%queue_name% queue is %value% seconds.

First message age in persistent queue/queue (seconds)

This metric gives the age (in seconds) of the first message in the persistent queue for all non-system queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Age of first message in %schema%.%queue_name% queue is %value% seconds.

Global Database Name

This metric provides the unique global database name.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Messages processed per persistent queue (%)

This metric gives the messages processed percentage for the persistent queue. Messages processed percent is calculated as the percent of the total number messages processed or dequeued to the total number of messages received or enqueued.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Messages processed for queue %schema%.%queue_name% is %value% percent.

Messages processed per persistent queue per minute (%)

This metric gives the messages processed percentage per minute per persistent queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Messages processed per minute in the last interval for queue %schema%.%queue_name% is %value%

Total messages processed per persistent queue per minute

This metric gives the total number of messages processed per minute per persistent queue in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Total messages processed per minute in the last interval for queue %schema%.%queue_name% is %value%

Total messages received per persistent queue per minute

This metric gives the total number of messages received or enqueued into the queue per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Total messages received per minute in the last interval for queue %schema%.%queue_name% is %value%

Total number of messages processed

This metric gives the total number of messages processed in the last collection interval of the metric.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Total number of messages received

This metric gives the total number of messages received in the last collection interval of the metric.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Messages per persistent queue per subscriber

The metrics in this category monitor the age and state of the first (top of the queue) message for each persistent queue per queue subscriber in the database except for the system queues. Queues that are in the schema of SYS, SYSTEM, DBSNMP, and SYSMAN are defined as system level queues.

Average age of messages/persistent queue/subscriber (seconds)

This metric display's the average age of messages in the persistent queue per queue in seconds.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Average age of messages for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

First message age in persistent queue/subscriber (seconds)

This metric gives the age (in seconds) of the first message in the persistent queue per subscriber for all non-system queues in the database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Age of first message for subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% seconds.

Global Database Name

This metric provides the unique global database name.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Messages processed/persistent queue/subscriber per minute (%)

This metric gives the messages processed percentage per minute per persistent queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value%

Messages processed/persistent queue/subscriber (%)

This metric gives the messages processed percentage for the persistent queue per subscriber. Messages processed percent is calculated as the percent of the total number messages processed or dequeued to the total number of messages received or enqueued.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Messages processed for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value% percent.

Total messages processed/persistent queue/subscriber per minute (%)

This metric gives the messages processed percentage per minute per persistent queue subscriber in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Total messages processed per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value%.

Total messages received/persistent queue/subscriber per minute

This metric gives the total number of messages received or enqueued into the queue per subscriber per minute in the last collection interval of the metric.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 24 Hours	Not Defined	Not Defined	Total messages received per minute in the last interval for the subscriber %subs_name% %subs_address% in %schema%.%queue_name% queue is %value%.

Total number of messages processed

This metric gives the total number of messages processed in the last collection interval of the metric.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Total number of messages received

This metric gives the total number of messages received in the last collection interval of the metric.

Target Version	Collection Frequency
All Versions	Every 24 Hours

PDB Mode

This metric category provides the current Open or Restricted mode of a PDB target in a RAC instance. It covers the READ ONLY, READ WRITE, MOUNTED, READ ONLY RESTRICTED, and READ WRITE RESTRICTED modes.

Target Version	Collection Frequency
All versions	Collection is based on the collection schedule of the parent CDB or is triggered when PDB open or close events are received from the RAC instance.

Metric Name	Description
Any Restricted	Indicates if the PDB is open in restricted modes (Read Only Restricted or Read Write Restricted). Possible values are: • YES • NO **Default Warning Threshold**: Yes **Default Critical Threshold**: Not defined **Number of Occurrences**: 1 **Alert Text**: Pluggable Database %target% is open in restricted mode on database instance %keyValue%.
Mode	Open mode of the PDB. Possible values are: • READ ONLY • READ WRITE • MOUNTED **Default Warning Threshold**: MOUNTED **Default Critical Threshold**: Not defined **Number of Occurrences**: 1 **Alert Text**: Pluggable Database %target% is CLOSED (%value%) on database instance %keyValue%.

Metric Name	Description
Read Only Restricted	Indicates if the PDB is in Read Only Restricted mode. Possible values are: • YES • NO **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Number of Occurrences**: 1 **Alert Text**: Pluggable Database %target% is open in read only restricted mode on database instance %keyValue%.
Read Write Restricted	Indicates if the PDB is in Read Write Restricted mode. Possible values are: • YES • NO **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Number of Occurrences**: 1 **Alert Text**: Pluggable Database %target% is open in read write restricted mode on database instance %keyValue%.

Resource Usage

This metric category provides the resource usage statistics of a PDB.

Target Version	Evaluation and Collection Frequency
12cR2, 18c, 19c, 20c, 21c	Every 1 hour

Metric Name	Description	Unit
Average Buffer Cache Usage (MB)	Average usage of the buffer cache by the PDB, in megabytes. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average buffer cache usage by sessions in Pluggable Database on database instance '%inst_name%' has been over the %severity% setting for past %num_of_occur% collection(s).	Megabytes
Average CPU Utilization (%)	Average percentage of the CPU consumed by the consumer group. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average CPU usage on database instance '%inst_name%' has been lower than the %severity% setting for the past %num_of_occur% collection(s).	Percentage
Average Exempted I/O (per second)	Average I/O per second that was exempted from throttling in the PDB.	NA
Average I/O Count (per second)	Average I/O operations per second for the PDB. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average I/O operation frequency detected on database instance '%inst_name%' has been lower than the %severity% setting the past %num_of_occur% collection(s).	NA

Metric Name	Description	Unit
Average I/O Data Size (MB/s)	Average megabytes of I/O per second for the PDB. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average I/O operation data size detected on database instance '%inst_name%' has been lower than the %severity% setting for the past %num_of_occur% collection(s).	Megabytes
Average I/O Throttle Time (ms)	Average throttle time per I/O operation for the PDB, in milliseconds. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average throttle length per I/O operation on database instance '%inst_name%' has been over the %severity% setting for past %num_of_occur% collection(s).	Milliseconds
Average Number of Active Parallel Servers	Average number of parallel servers that are actively running as part of a parallel statement. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average number of actively running parallel servers on database instance '%inst_name%' has been lower than the %severity% setting for past %num_of_occur% collection(s).	NA
Average Number of Parallel Servers Requested	Average number of parallel servers that are requested by queued parallel statements. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average number of queued parallel server requests on database instance '%inst_name%' has been over the %severity% setting for past %num_of_occur% collection(s).	NA
Average Number of Queued Parallel Statements	Average number of parallel statements that are queued. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average queued parallel statement requests on database instance '%inst_name%' has been over the %severity% setting for past %num_of_occur% collection(s).	NA
Average Number of Running Parallel Statements	Average number of parallel statements that are running. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average active running parallel statement on database instance '%inst_name%' has been lower than the %severity% setting for past %num_of_occur% collection(s).	NA
Average Number of Running Sessions	Average number of sessions in the consumer group that are currently running. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average number of running sessions on database instance '%inst_name%' has been lower than the %severity% setting for past %num_of_occur% collection(s).	NA

Metric Name	Description	Unit
Average Number of Waiting Sessions	Average number of sessions in the consumer group that are waiting for CPU due to resource management. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average number of waiting sessions on database instance '%inst_name%' has been over the %severity% setting for past %num_of_occur% collection(s).	NA
Average PGA Usage (MB)	Average PGA usage by the PDB, in megabytes. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average PGA usage by sessions in Pluggable Database on database instance '%inst_name%' has been over the %severity% setting for past %num_of_occur% collection(s).	Megabytes
Average SGA Usage (MB)	Average SGA usage by the PDB, in megabytes. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average SGA usage by sessions in Pluggable Database on database instance '%inst_name%' has been over the %severity% setting for past %num_of_occur% collection(s).	Megabytes
Average Shared Pool Usage (MB)	Average usage of the shared pool by the PDB, in megabytes. **Default Warning Threshold**: Not defined **Default Critical Threshold**: Not defined **Alert Text**: Average shared pool usage by sessions in Pluggable Database on database instance '%inst_name%' has been over the %severity% setting for past %num_of_occur% collection(s).	Megabytes
Average Exempted I/O Data Size (MB/s)	Average megabytes of I/O executed per second that were exempted from throttling in the current PDB.	Megabytes
Buffer Cache Usage (MB)	Buffer cache usage by the PDB, in megabytes.	Megabytes
CPU Consumption Time (ms)	Cumulative amount of CPU time consumed by all sessions in the consumer group, in milliseconds.	Milliseconds
CPU Count	Total number of CPUs available to the PDB.	NA
CPU Utilization Limit (%)	Maximum percentage of CPU that the consumer group can use at any time.	Percentage
CPU Wait Time (ms)	Cumulative amount of time that sessions waited for CPU because of resource management.	Milliseconds
High Buffer Cache Usage (MB)	Highest usage of the buffer cache by the PDB, in megabytes.	Megabytes
High PGA Usage (MB)	Highest usage of the PGA by the PDB, in megabytes.	Megabytes
High SGA Usage (MB)	Highest usage of the SGA by the PDB, in megabytes.	Megabytes
High Shared Pool Usage (MB)	Highest usage of the shared pool by the PDB, in megabytes.	Megabytes
Low Buffer Cache Usage (MB)	Lowest usage of the buffer cache by the PDB, in megabytes.	Megabytes

Metric Name	Description	Unit
Low PGA Usage (MB)	Lowest usage of the PGA by the PDB, in megabytes.	Megabytes
Low SGA Usage (MB)	Lowest usage of the SGA by the PDB, in megabytes.	Megabytes
Low Shared Pool Usage (MB)	Lowest usage of the shared pool by the PDB, in megabytes.	Megabytes
Parallel Servers Limit	Number of parallel servers that can be used by this consumer group.	NA
PGA Usage (MB)	PGA usage by the PDB, in megabytes.	Megabytes
Running Sessions Limit	Maximum number of sessions in the consumer group that can run simultaneously.	NA
SGA Usage (MB)	SGA usage by the PDB, in megabytes.	Megabytes
Shared Pool Usage (MB)	Usage of the shared pool by the PDB, in megabytes.	Megabytes

Response

This metric category represent the responsiveness of the Oracle Server, with respect to a client. For PDB, it is also indicative of the state of the PDB. For example, an Open state maps to Up and a Closed/Mounted state maps to Down. If the PDB target exists but the PDB has been dropped/unplugged from the CDB, the target will be in Metric Collection Error.

State

This metric represents the state of the pluggable database.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 5 Minutes	MOUNTED	DOWN¦UNKNOWN*	The database status is %value%.

User Action

Check the listener configured for the CDB to make sure it is running and serves the CDB. If listener is fine, check if the PDB is in the mount/closed state and open the PDB.

Status

This metric checks whether a new connection can be established to a database. If the maximum number of users is exceeded or the listener is down, this test is triggered.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 5 Minutes	Not Defined	0	Failed to connect to database instance: %oraerr%.

User Action

Check the listener configured for the CDB to make sure it is running and serves the CDB. If listener is fine, check if the PDB is in the mount/closed state and open the PDB.

ORA- Error

This is a user-readable message, including any Ora-error, encountered during response evaluation.

Target Version	Collection Frequency
All Versions	Every 5 Minutes

User Action

Check the listener configured for the CDB to make sure it is running and serves the CDB. If listener is fine, check if the PDB is in the mount/closed state and open the PDB.

Rollback Segments

This metric category provides the rollback metrics.

Name

Name of the rollback segment

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The SEGMENT_NAME column from the CDB_ROLLBACK_SEGS view.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Aveactive

Current size of active extents, averaged over time.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The AVEACTIVE column from the V$ROLLSTAT view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Aveshrink

Average shrink size

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The AVESHRINK column from the V$ROLLSTAT view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Extents

Number of extents in the rollback segment

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The EXTENTS column from the V$ROLLSTAT view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Hwmsize

High watermark of the rollback segment size

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The HWMSIZE column from the V$ROLLSTAT view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Initial Size

Initial extent size in bytes

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The INITIAL_EXTENT column from the CDB_ROLLBACK_SEGS view.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Maximum Size

Maximum number of extents

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The MAX_EXTENTS column from the CDB_ROLLBACK_SEGS view.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Minimum Extents

Minimum number of extents

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The MIN_EXTENTS column from the CDB_ROLLBACK_SEGS view.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Next Size

Secondary extent size in bytes

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The NEXT_EXTENT column from the CDB_ROLLBACK_SEGS view.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Optsize

Optimal size of the rollback segment

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The OPTSIZE column from the V$ROLLSTAT view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Pct Increase

Percent increase for extent size

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The PCT_INCREASE column from the CDB_ROLLBACK_SEGS view.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Size

Size (in bytes) of the rollback segment.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The RSSIZE column from the V$ROLLSTAT view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Shrinks

Number of times the size of a rollback segment decreases

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The SHRINKS column from the V$ROLLSTAT view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Status

The status of the rollback segment (one of OFFLINE, ONLINE, NEEDS RECOVERY, PARTLY AVAILABLE, UNDEFINED).

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The STATUS column from the CDB_ROLLBACK_SEGS view.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Tablespace Name

Name of the tablespace containing the rollback segment

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The TABLESPACE_NAME column from the CDB_ROLLBACK_SEGS view.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Wraps

Number of times the rollback segment is wrapped

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The WRAPS column from the V$ROLLSTAT view

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Segment Advisor Recommendations

This metric category provides segment advisor recommendations. Oracle uses the Automatic Segment Advisor job to detect segment issues regularly within maintenance windows. It determines whether the segments have unused space that can be released. The Number of recommendations is the number of segments that have Reclaimable Space. The recommendations come from all runs of the automatic segment advisor job and any user-scheduled segment advisor jobs.

Number of recommendations

Oracle uses the Automatic Segment Advisor job to detect segment issues regularly within maintenance windows. It determines whether the segments have unused space that can be released. The Number of recommendations is the number of segments that have Reclaimable Space. The recommendations come from all runs of the automatic segment advisor job and any user-scheduled segment advisor jobs.

This is a database-level metric. For cluster databases, this metric is monitored at the cluster database target level and not by member instances.

Target Version	Collection Frequency
All Versions	Every 60 Minutes

Shard Apply Lag

The metrics in this category provide information regarding the shard replication apply coordinator process and replication unit apply lag.

Target Version	Evaluation and Collection Frequency
23ai and later	Every 15 minutes

Note that the collection for this metric is disabled by default and you can either enable it on the metric page or by creating a custom monitoring template. For information on monitoring templates, see Monitoring Templates in *Oracle Enterprise Manager Monitoring Guide*.

Metric Name	Description	Unit
Apply Coordinator Process Name	Name of the apply coordinator process for the shard.	None

Metric Name	Description	Unit
Apply Coordinator Process State	State of the apply coordinator process for the shard. Possible values are: • IDLE • APPLYING • ABORTING • SHUTTING DOWN CLEANLY	None
Apply Time	Time at which the Logical Change Record (LCR) with the highest log index was recorded.	None
High Watermark LCR Log Index	LCR with the highest commit log index, which means that no LCR with a higher commit log index has been applied.	None
Last Processed Logical Change Record (LCR) Log Index	LCR log index last processed by the apply coordinator.	None
Message Creation Time	Time when the LCRs corresponding to the high watermark were created.	None
Shard Apply Lag (sec)	Replication unit apply lag (in seconds) for the shard.	Seconds

Shard Replication Units

The metrics in this category provide details of the replication units in the shard.

Target Version	Evaluation and Collection Frequency
23ai and later	Every 15 minutes

Note that the collection for this metric is disabled by default and you can either enable it on the metric page or by creating a custom monitoring template. For information on monitoring templates, see Monitoring Templates in *Oracle Enterprise Manager Monitoring Guide*.

Metric Name	Description	Unit
Leader ID	Leader's `SHARD_ID` for the corresponding replication unit.	None
Leader Shard	Name of the leader shard of the corresponding replication unit.	None
Replication Unit Ready	Indicates that the replication unit is open for workload.	None
Replication Unit Recovery Time	Time taken for recovery in the shard.	Seconds
Replication Unit Role	Role of the shard for the replication unit. Possible values are: • 1 = leader • 2 = follower • 3 = candidate	None
Replication Unit Role Change Time	Time when the role for a replication unit in a shard was changed.	None
Replication Unit Role Changed	Indicates whether the replication unit role changed in the shard since the last metric collection. If the role has changed, the value denotes the new role, otherwise, the value is "Unchanged".	None
Role Change Duration	Time taken (in seconds) to switch the role of a replication unit.	Seconds

Metric Name	Description	Unit
Role Change Elapsed Time	Time taken for the role change to complete. The purpose of this metric is to generate an alert if the role of the replication unit has changed, along with the Role Change Duration metric value (time taken to switch the role).	Seconds

Shard Transport Lag

The metrics in this category provide information regarding the shard replication transport lag.

Target Version	Evaluation and Collection Frequency
23ai and later	Every 15 minutes

Note that the collection for this metric is disabled by default and you can either enable it on the metric page or by creating a custom monitoring template. For information on monitoring templates, see Monitoring Templates in *Oracle Enterprise Manager Monitoring Guide*.

Metric Name	Description	Unit
Flow Controlled Followers	Shard ID of the flow controlled followers shard.	None
Follower Shard Database Name	Name of the follower shard database.	None
LCR Producer Startup Time	Start time of the Logical Change Record (LCR) producer process.	None
LCR Producer State	State of the LCR producer in the shard.	None
LCR Propagation Time	Time taken (in seconds) to propagate the last LCR.	Seconds
Network Sender Connection Errors Count	Number of network sender connection errors.	Count
Network Sender Startup Time	Start time of the network sender process.	None
Network Sender State	State of the network sender.	None
Transport Lag (# log indexes)	Transport lag of the follower replication unit in the number of log indexes.	None
Transport Lag (sec)	Replication unit transport lag (in seconds) for the follower shard.	Seconds

Tablespaces

This metric category provides details on the tablespace metrics.

Allocation Type

Type of extent allocation in effect for the tablespace (one of SYSTEM, UNIFORM, USER).

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The ALLOCATION_TYPE column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Big File

Indicates whether the tablespace is a bigfile tablespace (YES) or a smallfile tablespace (NO).

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The BIGFILE column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Block Size

Tablespace block size.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The BLOCK_SIZE column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Extent Management

Indicates whether the extents in the tablespace are dictionary managed (DICTIONARY) or locally managed (LOCAL).

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The EXTENT_MANAGEMENT column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Increment By

Default percent increase for extent size.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The PCT_INCREASE column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Initial Ext Size

Default initial extent size.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The INITIAL_EXTENT column from CDB_TABLESPACES

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Logging

Default logging (one of LOGGING, NOLOGGING).

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The LOGGING column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Max Extents

Default maximum number of extents

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The MAX_EXTENTS column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Minimum Extent Size

Minimum extent size for this tablespace.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The MIN_EXTLEN column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Minimum Extents

Default minimum number of extents.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The MIN_EXTENTS column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Next Extent

Default incremental extent size.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The NEXT_EXTENT column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Segment Space Management

Indicates whether the free and used segment space in the tablespace is managed using free lists (MANUAL) or bitmaps (AUTO).

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The SEGMENT_SPACE_MANAGEMENT column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Size

Allocated size in bytes of the tablespace.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

Sum of BYTES column from CDB_DATA_FILES for permanent and undo tablespaces or the sum of BYTES column from CDB_TEMP_FILES for temporary tablespaces.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Status

Tablespace status (one of ONLINE, OFFLINE, READ ONLY).

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The STATUS column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Tablespace Name

Name of the tablespace.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The TABLESPACE_NAME column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Type

One of UNDO, PERMANENT or TEMPORARY.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

The CONTENTS column from CDB_TABLESPACES.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Used Size(B)

Size in bytes of the used space of the tablespace.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

Sum of BYTES column from CDB_DATA_FILES minus the sum of BYTES from CDB_FREE_SPACE for permanent and undo tablespaces or the sum of USED_BLOCKS FROM GV$SORT_SEGMENT times BLOCK_SIZE from CDB_TABLESPACES for temporary tablespaces.

User Action

This is a configuration metric so this column is purely informative and collected to provide DBAs with information necessary to manage the current database.

Tablespace Used Space (MB)

This metric calculates the space used for each tablespace. It is not intended to generate alerts. Rather it should be used in conjunction with the Tablespace Allocated Space (MB) metric to produce a historical view of the amount of space being used and unused by each tablespace.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

Tablespace Used Space (MB) is Tablespace Allocated Space (MB)− Tablespace Allocated Free Space (MB) where:

Tablespace Allocated Space (MB) is calculated by looping through the tablespace's data files and totaling the size of the data files.

Tablespace Allocated Free Space (MB) is calculated by looping through the tablespace's data files and totaling the size of the free space in each data file.

Tablespace Allocation

The metrics in this metric category check the amount of space used and the amount of space allocated to each tablespace. The used space can then be compared to the allocated space to determine how much space is unused in the tablespace. This metric is intended for reporting, rather than alerts. Historical views of unused allocated free space can help DBAs to correctly size their tablespaces, eliminating wasted space.

Tablespace Allocated Space (MB)

The allocated space of a tablespace is the sum of the current size of its data files. A portion of this allocated space is used to store data while some may be free space. If segments are added to a tablespace, or if existing segments grow, they will use the allocated free space. The allocated free space is only available to segments within the tablespace. If, over time, the segments within a tablespace are not using this free space, the allocated free space is not being used.

This metric calculates the space allocated for each tablespace. It is not intended to generate alerts. Rather it should be used in conjunction with the Allocated Space Used (MB) metric to produce a historical view of the amount of space being used and unused by each tablespace.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Data Source

Tablespace Allocated Space (MB) is calculated by looping though the tablespace's data files and totalling the size of the data files.

Tablespace Used Space (MB)

The allocated space of a tablespace is the sum of the current size of its datafiles. Some of this allocated space is used to store data and some of it may be free space. If segments are added to a tablespace, or if existing segments grow, they will use the allocated free space. The allocated free space is only available to segments within the tablespace. If, over time, the segments within a tablespace are not using this free space, then the allocated free space is being wasted.

This metric calculates the space used for each tablespace. It is not intended to generate alerts. Use in conjunction with the Tablespace Allocated Space (MB) metric to produce a historical view of the amount of space being used and unused by each tablespace.

Target Version	Collection Frequency
All Versions	Every 24 Hours

Tablespaces Full

The metrics in this metric category check for the amount of space used by each tablespace. The used space is then compared to the available free space to determine tablespace fullness. The available free space accounts for the maximum data file size as well as available disk space. This means that a tablespace will not be flagged as full if data files can extend and there is enough disk space available for them to extend.

Tablespace Free Space (MB)

As segments within a tablespace grow, the available free space decreases. If there is no longer any available free space, meaning data files reached their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

This metric checks for the total available free space in each tablespace. This metric is intended for larger tablespaces, where the Available Space Used (%) metric is less meaningful. If the available free space falls below the size specified in the threshold arguments, then a warning or critical alert is generated.

If the version of the monitored database target is Oracle Database 10*g* Release 1 or later and the tablespace uses Local Extent Management, then the Oracle Database Server evaluates this metric internally every 10 minutes. Alternatively, if the version of the monitored Database target is Oracle 9*i* or earlier, or the tablespace uses Dictionary Extent Management, then the Oracle Management Agent tests the value of this metric every 30 minutes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 30 Minutes	Not Defined	Not Defined	Tablespace [%name%] only has [%value% megabytes] free space.

Data Source

MaximumSize Total Used Space where:

- TotalUsedSpace: Total used space in MB of tablespace
- MaximumSize: Maximum size (in MB) of the tablespace. The maximum size is determined by looping through the tablespaces data files, as well as additional free space on the disk that would be available for the tablespace should a data file autoextend.

User Action

Perform one of the following:

- Increase the size of the tablespace by: Enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.
- If the tablespace is suffering from tablespace free space fragmentation problems, consider reorganizing the entire tablespace.
- Relocate segments to another tablespace, thereby increasing the free space in this tablespace.
- Run the Segment Advisor on the tablespace.

Tablespace Space Used (%)

As segments within a tablespace grow, the available free space decreases. If there is no longer any available free space, meaning data files have reached their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

This metric checks the Available Space Used (%) for each tablespace. If the percentage of used space is greater than the values specified in the threshold arguments, then a warning or critical alert is generated.

If the version of the monitored database target is Oracle Database 10*g* Release 1 or later and the tablespace uses Local Extent Management, then the Oracle Database Server evaluates this metric internally every 10 minutes. Alternatively, if the version of the monitored Database target is Oracle 9*i* or earlier, or the tablespace uses Dictionary Extent Management, then the Oracle Management Agent tests the value of this metric every 30 minutes.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 30 Minutes	85	97	Tablespace [%name%] is [%value% percent] full

Data Source

(TotalUsedSpace / MaximumSize) * 100 where:

- TotalUsedSpace: total used space in MB of tablespace
- MaximumSize: Maximum size (in MB) of the tablespace. The maximum size is determined by looping through the tablespace□s data files, as well as additional free space on the disk that would be available for the tablespace should a data file autoextend.

For additional information about the data source, refer to the fullTbsp.pl Perl script located in the sysman/admin/scripts directory.

User Action

Perform one of the following:

- Increase the size of the tablespace by: Enabling automatic extension for one of its existing data files, manually resizing one of its existing data files, or adding a new data file.
- If the tablespace is suffering from tablespace free space fragmentation problems, consider reorganizing the entire tablespace.
- Relocate segments to another tablespace, thus increasing the free space in this tablespace.
- Run the Segment Advisor on the tablespace.

Tablespaces Full (Temp)

The metrics in this category check for the amount of space used by each locally managed temporary tablespace. The used space is then compared to the available free space to determine tablespace fullness. The available free space takes into account the maximum data file size as well as available disk space. This means that a tablespace will not be flagged as full if data files can extend and there is enough disk space available for them to extend.

Tablespace Free Space (MB) (Temp)

As segments within a tablespace grow, the available free space decreases. If there is no more free space available, that is, the data files have hit their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

This metric checks for the total available free space in each temporary tablespace. This metric is intended for larger temporary tablespaces, where the Available Space Used (%) metric is less meaningful. If the available free space falls below the size specified in the threshold arguments, then a warning or critical alert is generated.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 30 Minutes	Not Defined	Not Defined	Tablespace [%name%] has [%value% mbytes] free

Tablespace Space Used (%) (Temp)

As segments within a tablespace grow, the available free space decreases. If there is no more free space available, that is, the data files have hit their maximum size or there is no more disk space, then the creation of new segments or the extension of existing segments will fail.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 30 Minutes	Not Defined	Not Defined	Tablespace [%name%] is [%value% percent] full

Temporary File Status

This metric category provides the temporary file status metrics.

File Name

The name of the temporary file.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The NAME column of the V$TEMPFILE view.

User Action

This information is purely Informative.

Status

The status of the temporary file, either ONLINE or OFFLINE.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 15 Minutes	OFFLINE	Not Defined	The temporary file %NAME% in pluggable database %pdbName% is %STATUS%.

Data Source

The STATUS column of the V$TEMPFILE view.

User Action

If the default settings are still in use, an offline temporary file will generate an incident (alert). Determine whether this file is expected to be offline and if not, switch it back online. Also, if this file is expected to be offline, ensure that the temporary tablespace it belongs to has other available online temporary files.

Temporary File Id

The absolute file number of the temporary file, used to join with other database tables and views to retrieve additional information.

Target Version	Collection Frequency
All Versions	Every 15 Minutes

Data Source

The FILE# column of the V$TEMPFILE view.

User Action

This information is purely informative.

Total Objects by Schema

The metrics in this category contain the metric that provides the number of database objects in a schema.

Total Object Count

This metric displays the total number of database objects in a schema.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 15 Minutes	Not Defined	Not Defined	%value% object(s) exist in the %owner% schema.

Total Tables by Schema

The metrics in this category contain the metric that provides the number of tables in a schema.

Total Table Count

This metric displays the total number of tables in a schema.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All Versions	Every 15 Minutes	Not Defined	Not Defined	%value% table(s) exist in the %owner% schema.

Wait Count

This metric category contains the metrics that approximate the percentage of time spent waiting by user sessions.

Active Sessions: CPU

This metric represents the active sessions using CPU.

Target Version	Collection Frequency
All versions	Every 15 Minutes

Active Sessions: I/O

This metric represents the active sessions waiting for I/O.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

Active Sessions: Other

This metric represents all the waits that are neither idle nor user I/O.

Target Version	Evaluation and Collection Frequency	Default Warning Threshold	Default Critical Threshold	Alert Text
All versions	Every 10 Minutes	Not Defined	Not Defined	The Management Agent generates the alert text.[1]

1 For releases earlier than Oracle Database plug-in release 12.1.0.6, the Database Server generates this alert text.

6
Autonomous Database

This chapter provides information about the Autonomous Database metrics.

Metric Category	Metric Name	Unit	Collection Frequency	Description
CPU Usage	CPU Utilization	Percentage	Every 5 minutes	The CPU utilization aggregated across all consumer groups. This metric is reported based on the number of CPUs the database is allowed to use, which is two times the number of OCPUs.
CPU Usage	CPU Usage Seconds	Seconds	Every 5 minutes	The average rate of accumulation of CPU time by foreground sessions in the database over the time interval. The CPU time component of Average Active Sessions.
CPU Utilization by Consumer Group	CPU Utilization by Consumer Group	Percentage	Every 5 minutes	The CPU utilization in each consumer group.
CPU Utilization by Consumer Group	Queued Statements	NA	Every 5 minutes	The number of queued SQL statements aggregated across all consumer groups, during the selected interval.
CPU Utilization by Consumer Group	Running Statements	NA	Every 5 minutes	The number of running SQL statements aggregated across all consumer groups, during the selected interval.
Database Load Profile	Average Active Session	NA	Every 5 minutes	The average rate of accumulation of database time (CPU + Wait) by foreground sessions in the database over the time interval. This is also known as Average Active Sessions.
Database Load Profile	DB Time	Seconds	Every 5 minutes	The amount of time spent in performing database user-level calls.
Database Load Profile	Parse Elapsed Time	Seconds	Every 5 minutes	The amount of time spent parsing SQL statements.
Database Load Profile	Transaction Response Time	Seconds	Every 5 minutes	The amount of time spent in database operations per transaction. It is derived from the total time that user calls spend in the database (DB time) and the number of commits and rollbacks performed.
Database Throughput	Executions per second	NA	Every 5 minutes	The number of user and recursive calls that executed SQL statements per second.
Database Throughput	Logons per second	NA	Every 5 minutes	The number of successful logons per second.
Database Throughput	Total Parse Count per second	NA	Every 5 minutes	The number of hard and soft parses per second.

Metric Category	Metric Name	Unit	Collection Frequency	Description
Database Throughput	Transactions per second	NA	Every 5 minutes	The number of transactions per second.
Database Throughput	User Calls per second	NA	Every 5 minutes	The number of logins, parses, or execute calls per second.
SQL Execution Time	SQL Execution Time	Seconds	Every 5 minutes	The amount of time spent in executing SQL statements.
Storage	Storage Total	Terabytes	Every 1 hour	The maximum amount of space allocated to the database during the interval.
Storage	Storage Used	Terabytes	Every 1 hour	The amount of storage space used.
Storage	Storage Used	Percentage	Every 1 hour	The percentage of storage space used.
Wait Count	Active Sessions : CPU	NA	Every 5 minutes	The active sessions using CPU.
Wait Count	Active Sessions : I/O	NA	Every 5 minutes	The active sessions using I/O.
Wait Count	Active Sessions : Other	NA	Every 5 minutes	The active sessions that are neither idle nor using CPU or I/O.
Waits by Wait Class	Foreground Wait Time	Seconds	Every 5 minutes	The average rate of accumulation of non-idle wait time by foreground sessions in the database over the time interval. The wait time component of Average Active Sessions.

Configuration Metric

Metric Category	Metric Name	Unit	Collection Frequency	Description
Other Collected Items	Initialization Parameters	NA	Every 24 hours	Oracle Enterprise Manager collects Initialization Parameter configuration information for Autonomous Databases, thereby allowing you to compare and monitor configuration history for changes. This also helps in performing root cause analysis and impact analysis by comparing the changes. To manage configuration information, click **<Autonomous Database Type>** menu > **Configuration**.

7

Global Data Services

This chapter provides information about the Global Data Services metrics.

Global Data Services (GDS) provide dynamic load balancing, failover, and centralized service management for a set of replicated databases that offer common services. The sections in this chapter provide information about the metrics available for the following GDS targets:

- Global Service Manager
- Global Data Service Pool

Global Service Manager

This section provides information about the metric categories available for the Global Service Manager target.

A **Global Service Manager** is the central software component of GDS, which provides service-level load balancing, failover, and centralized management of services in the GDS configuration.

Shard Global Service Performance

The metrics in this category report the shard global service performance details.

Note that this metric is disabled by default and you can enable it in the Oracle Enterprise Manager console or using a Monitoring template.

Target Version	Evaluation and Collection Frequency
12c and later	Every 15 minutes

Metric Name	Description	Unit
Average Active Sessions	Ratio of the DB time to wall clock time.	None
Service Time (m/s)	Average time spent per user call.	Milliseconds
Throughput (calls/sec)	Number of user calls per second.	Operations per second

Load

This section provides information on the metrics in the Load category.

Target Version	Evaluation and Collection Frequency
12c and later	Every 5 minutes

Metric Name	Description	Unit
Connections Established (per min)	Count of connections established per minute.	None
Connections Refused (per min)	Count of connections refused per minute.	None

General Status

This section provides information on the metrics in the General Status category.

Target Version	Evaluation and Collection Frequency
12c and later	Every 5 minutes

Metric Name	Description	Unit
Host	GSM host	None
Port	GSM port	None
Region	Region name	None
Region Master	Regional master GSM	None
Start Date	GSM start time	Date
TNSPING Response Time	Response time of TNSPING test from GDS home to GSM server.	Milliseconds
TNSPING Status	Status of TNSPING test from GDS home to GSM server.	Status (0 or 1)

Global Data Service Pool

This section provides information about the metric category available for the Global Data Service Pool target.

A **Global Data Services Pool** is a named subset of databases within a GDS configuration that provides a unique set of global services and belongs to the same administrative domain. Partitioning of GDS configuration databases into pools simplifies service management and provides higher security by allowing each pool to be administered by a different administrator.

Global Service Status

This section provides information on the metrics in the Global Service Status category.

Target Version	Evaluation and Collection Frequency
12c and later	Every 5 minutes

Metric Name	Description	Unit
Preferred Database(s)	Preferred list of instances where the service should run. If the role is specified to a service, then all the instances matching the role in the GDS configuration are considered preferred.	None
Running Database(s)	Instances where the service is running.	None

Metric Name	Description	Unit
Service Name	Global service name	None
Service Status	Status of global service based on running and preferred instances: • If the service is running on all preferred instances then it is "Up". • If the service is running on some of preferred instances then it is "Up with Warning". • If the service is not running on any of preferred instances then it is "Down".	None

8
Globally Distributed Database

This chapter provides information about the Globally Distributed Database metrics.

A distributed database is a method of partitioning data to distribute the computational and storage workload. This is achieved using a technology called database sharding, in which segments of a data set or a shard is distributed across lots of databases on lots of different computers. All the shards together make up a single logical database, called a *distributed database*. For more information, see Oracle Globally Distributed Database Overview in *Oracle Globally Distributed Database*.

Shard Director

This section provides information about the metric categories available for the Shard Director target.

Shard Director is a Global Service Manager, which supports the routing of connections based on data location.

Global Service Performance

This section provides information on the metrics in the Global Service Performance category.

Target Version	Evaluation and Collection Frequency
12c and later	Every 15 minutes

Metric Name	Description	Unit
Average Response Time	Elapsed time per user call.	Milliseconds
CPU IO Violations	Count of CPU IO threshold violations.	None
CPU Load	CPU utilization	Percentage
Calls Per Second	Number of logins, parses, or execute calls per second during the sample period.	None
Collected Region	Name of the GSM region.	None
Latency	Network latency between database region and GSM region.	Milliseconds
Latency Region	Name of the GSM region for which latency is computed.	None
Master	Global Master GSM	None
Region Master	Regional Master GSM	None
Runtime Load Balance	Percentage of workload for runtime load balancing purposes.	None
Session Count	Current session count over the past 30 seconds.	None

Load

This section provides information on the metrics in the Load category.

Target Version	Evaluation and Collection Frequency
12c and later	Every 5 minutes

Metric Name	Description	Unit
Connections Established (per min)	Count of connections established per minute.	None
Connections Refused (per min)	Count of connections refused per minute.	None

General Status

This section provides information on the metrics in the General Status category.

Target Version	Evaluation and Collection Frequency
12c and later	Every 5 minutes

Metric Name	Description	Unit
Host	GSM host	None
Port	GSM port	None
Region	Region name	None
Region Master	Regional master GSM	None
Start Date	GSM start time	Date
TNSPING Response Time	Response time of TNSPING test from GDS home to GSM server.	Milliseconds
TNSPING Status	Status of TNSPING test from GDS home to GSM server.	Status (0 or 1)

Chunk Performance

The metric in this category reports the chunk-level performance.

Note that this metric is disabled by default and you can enable it in the Oracle Enterprise Manager console or using a Monitoring template.

Target Version	Evaluation and Collection Frequency
12c and later	Every 15 minutes

Metric Name	Description	Unit
Throughput (calls/sec)	Number of calls per second in a chunk.	Operations per second

Response

The metric in this category reports the status of the Globally Distributed Database.

Target Version	Evaluation and Collection Frequency
12c and later	Event-driven

Metric Name	Description	Unit
Status	Status of the Globally Distributed Database.	None

Service Distribution

The metrics in this category report the service distribution details.

Target Version	Evaluation and Collection Frequency
12c and later	Every 5 minutes

Metric Name	Description	Unit
Calls per second distribution (Instance:::Database:::Region)	Number of calls distribution per service per shard database per region.	Operations per second
Region Name	Name of the region.	None
Response time distribution (Instance:::Database:::Region)	Response time distribution per service per shard database per region.	Milliseconds

Service Status

The metrics in this category report the service status details.

Target Version	Evaluation and Collection Frequency
12c and later	Every 5 minutes

Metric Name	Description	Unit
Preferred Database(s)	Preferred list of instances where the service should run. If the role is specified to a service, then all role matching instances are preferred.	None
Running Database(s)	Instances where the service is running.	None

Metric Name	Description	Unit
Status	Status of the global service based on running and preferred instances. • If the service is running on all preferred instances, then its status is Up. • If the service is running on some of the preferred instances, then its status is Up with Warning. • If the service is not running on any of the preferred instances, then its status is Down.	None

Shard Chunk Size (GB)

The metric in this category reports the size of a chunk per shard.

Note that this metric is disabled by default and you can enable it in the Oracle Enterprise Manager console or using a Monitoring template.

Target Version	Evaluation and Collection Frequency
12c and later	Every 15 minutes

Metric Name	Description	Unit
Chunk Size (GB)	Size of a chunk per shard.	GB

Shard Global Service Performance

The metrics in this category report the shard global service performance details.

Note that this metric is disabled by default and you can enable it in the Oracle Enterprise Manager console or using a Monitoring template.

Target Version	Evaluation and Collection Frequency
12c and later	Every 15 minutes

Metric Name	Description	Unit
Average Active Sessions	Ratio of the DB time to wall clock time.	None
Service Time (m/s)	Average time spent per user call.	Milliseconds
Throughput (calls/sec)	Number of user calls per second.	Operations per second

Shard Replication Units Summary

The metrics in this category provide a summary of the replication units in an Oracle Globally Distributed Database. Replication units in an Oracle Globally Distributed Database are created by Raft replication and distributed automatically to handle chunk assignment, chunk

movement, workload distribution, and balancing upon scaling (addition or removal of shards), which includes planned or unplanned shard availability changes.

Target Version	Evaluation and Collection Frequency
23ai and later	Every 15 minutes

Metric Name	Description	Unit
Replication Unit Number	Number of the replication unit.	None
Shard DB Number	Number of the shard that contains the replication unit.	None
Replication Unit Leader Shard DB Number	Number of the shard that leads the replication process.	None
Replication Unit Role	Role of the shard for this replication unit: • 1 = leader • 2 = follower • 3 = candidate	None

Data Source

```
SELECT rp.ru_number AS RU_NUMBER,
       rp.db_number AS SHARD_DB_NUMBER,
       rp.role AS RU_ROLE_IN_SHARD,
       ru.leaderid  As RU_LEADER_ID
FROM
       gsmadmin_internal.ru_peers rp,
       gsmadmin_internal.replication_unit ru
WHERE
       rp.ru_number = ru.ru_number
ORDER BY
       rp.ru_number
```

Shard Size (GB)

The metrics in this category report the allocated size of the shard.

Note that this metric is disabled by default and you can enable it in the Oracle Enterprise Manager console or using a Monitoring template.

Target Version	Evaluation and Collection Frequency
12c and later	Every 15 minutes

Metric Name	Description	Unit
Number of Chunks Change Rate	Rate of change in the number of chunks per shard during metric collection.	Percentage
Data Size Change Rate	Rate of change in data size per shard during metric collection.	Percentage
Data Size (GB)	Size of a shard.	GB
Number of Chunks	Number of chunks per shard.	None

Metric Name	Description	Unit
Number of Tablespaces	Number of tablespaces per shard.	None

Shard Status Details

The metrics in this category report shard status details.

Target Version	Evaluation and Collection Frequency
12c and later	Every 15 minutes

Metric Name	Description	Unit
Connect String	Shard database connect string used in the Shard Director (GSM) configuration.	None
Convert State	Displays shard database role change possibility.	None
DDL Error	Displays DDL error in GSM operations, if any.	None
DDL Number	DDL operation number in GSM.	None
Data Guard Broker ID	Data guard broker ID shared across the primary and standby databases.	None
Database Status	Status of the shard database in GSM.	None
Database Type	Shard database type. For example, RAC or Non-RAC.	None
Database Up	Status of the shard database in Oracle Enterprise Manager.	None
Deployment Status	Deployment status of the shard database from the GSM. For example, Deployed or Pending.	None
Host	Shard database host.	None
Is Primary	Shard database role.	None
Oracle Home	Oracle home path for the shard database.	None
Region Name	Region in which the shard database is deployed.	None
Shardgroup Name	Name of the shardgroup to which the shard database belongs.	None
Shardspace Name	Name of the shardspace to which the shard database belongs.	None

Shard Tablespace Summary

The metrics in this category report the shard tablespace summary.

Note that this metric is disabled by default and you can enable it in the Oracle Enterprise Manager console or using a Monitoring template.

Target Version	Evaluation and Collection Frequency
12c and later	Every 15 minutes

Metric Name	Description	Unit
Tablespace Set Name	Tablespace set name.	None

Metric Name	Description	Unit
Chunk Number	ID of the chunk.	None
Shardspace ID	ID of the shardspace.	None

Unique Keys Count

The metrics in this category report the details of the unique sharding keys per shard.

Note that this metric is disabled by default and you can enable it in the Oracle Enterprise Manager console or using a Monitoring template.

Target Version	Evaluation and Collection Frequency
12c and later	Every 1 hour

Metric Name	Description	Unit
Number of Unique Keys Change Rate	Rate of change of the number of unique keys per shard during metric collection.	Percentage
Unique Keys Count	Number of unique sharding keys present in the shard.	None