

Oracle® Enterprise Manager

Oracle Fusion Middleware Metric Reference Manual

11g Release 1 (11.1.0.1)

E18807-01

February 2011

Copyright © 2006, 2011, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	li
Audience	li
Documentation Accessibility	li
Related Documents	lii
Conventions	lii
How to Use This Manual	liii
Structure of the Oracle Enterprise Manager Oracle Fusion Middleware Metric Reference Manual....	liii
Background Information on Metrics, Thresholds, and Alerts	lv
1 WebLogic Server	
Application Overview Metrics	1-1
Application - Cached Beans.....	1-1
Application - Cache Accesses (per minute).....	1-1
Application - Bean Activations (per minute)	1-1
Application - Cache Hits (%)	1-1
Application - Cache Misses (per minute)	1-1
Application - Bean Accesses (per minute).....	1-1
Application - Bean Access Successes (%).....	1-1
Application - Free Bean Instances.....	1-2
Application - Bean Destroys (per minute).....	1-2
Application - Bean Access Failures (per minute)	1-2
Application - Beans in Use.....	1-2
Application - Bean Transaction Commits (%).....	1-2
Application - Bean Transaction Commits (per minute)	1-2
Application - Bean Transaction Rollbacks (per minute).....	1-2
Application - Bean Transaction Timeouts (per minute).....	1-2
Application - MDB Messages (per minute)	1-2
Application - Requests (per minute)	1-2
Application - Request Processing Time (ms)	1-2
Application - Active Sessions	1-2
Application - Work Manager Pending Requests	1-3
Application - Work Manager Requests (per minute)	1-3
Application Work Manager - Work Manager Pending Requests	1-3

Application Work Manager - Work Manager Requests (per minute).....	1-3
Application Work Manager - Work Manager Stuck Threads	1-3
Application Metrics	1-3
Application - Status.....	1-3
Datasource Metrics	1-3
Datasource - Available Connections	1-3
Datasource - Connections Created (per minute)	1-3
Datasource - Connections in Use	1-4
Datasource - Connection Leaks (per minute).....	1-4
Datasource - Connection Pool Size.....	1-4
Datasource - Connection Refresh Failures (per minute)	1-4
Datasource - Connection Request Failures (per minute).....	1-4
Datasource - Connection Requests (per minute).....	1-4
Datasource - Successful Connections (%).....	1-4
Datasource - Unavailable Connections	1-4
Datasource - Connection Requests Waiting	1-4
Datasource - Failed Waiting Connection Requests (per minute).....	1-4
Datasource - Connection Requests that Waited (per minute)	1-5
Datasource - Connection Wait Successes (%).....	1-5
Datasource - Connection Wait Successes (per minute)	1-5
Datasource - State.....	1-5
Datasource - Statements Added to Cache (per minute)	1-5
Datasource - Statements Discarded from Cache (per minute)	1-5
Datasource - Cached Statements not Used (per minute)	1-6
Datasource - Cached Statements Used (%)	1-6
Datasource - Cached Statements Used (per minute)	1-6
Datasource - Statement Cache Size.....	1-6
EJB Module Metrics	1-6
EJB Module - Cached Beans.....	1-6
EJB Module - Cache Accesses (per minute).....	1-6
EJB Module - EJB Activations (per minute)	1-6
EJB Module - EJB Cache Hits (%).....	1-6
EJB Module - EJB Cache Misses (per minute)	1-6
EJB Module - EJB Passivations (per minute).....	1-7
EJB Module - Bean Accesses (per minute).....	1-7
EJB Module - Bean Successes (%).....	1-7
EJB Module - Free Bean Instances.....	1-7
EJB Module - Bean Destroys (per minute).....	1-7
EJB Module - Failed Bean Accesses (per minute).....	1-7
EJB Module - Beans in Use.....	1-7
EJB Module - Bean Transaction Commits (per minute)	1-7
EJB Module - Bean Transaction Rollbacks (per minute)	1-7
EJB Module - Bean Transaction Timeouts (per minute).....	1-7
EJB Module - MDB Messages (per minute).....	1-7
EJB Pool Metrics	1-8
EJB Pool - Bean Accesses (per minute).....	1-8
EJB Pool - Successful Bean Accesses (%).....	1-8

EJB Pool - Available Beans.....	1-8
EJB Pool - Bean Destroys (per minute).....	1-8
EJB Pool - Bean Access Failures (per minute)	1-8
EJB Pool - Beans in Use.....	1-8
EJB Transaction Metrics	1-8
EJB Transaction - Bean Transaction Commits (%).....	1-8
EJB Transaction - Bean Transaction Commits (per minute)	1-8
EJB Transaction - Bean Transaction Rollbacks (per minute)	1-8
EJB Transaction - Bean Transaction Timeouts (per minute).....	1-9
JMS Metrics	1-9
JMS Connections	1-9
JMS Servers	1-9
JMS Server Metrics	1-9
Bytes Pageable	1-9
Bytes Paged In (per minute)	1-9
Bytes Paged Out (per minute)	1-9
Bytes Pending	1-9
Bytes Received (per minute).....	1-9
Bytes Stored.....	1-9
Destinations	1-10
Messages Paged In (per minute).....	1-10
Messages Paged Out (per minute).....	1-10
Messages Pending	1-10
Messages Received (per minute)	1-10
Messages Stored	1-10
Session Pools	1-10
JTA Metrics	1-10
Abandoned Transactions (per minute).....	1-10
Committed Transactions (%).....	1-10
Committed Transactions (per minute).....	1-10
Committed Transaction Time (seconds).....	1-11
Heuristic Transactions (per minute).....	1-11
Transactions Processed (per minute)	1-11
Transaction Rollbacks (per minute).....	1-11
Transaction Rollbacks due to Application Error (per minute).....	1-11
Transaction Rollbacks due to Resource Error (per minute).....	1-11
Transaction Rollbacks due to System Error (per minute)	1-11
Transaction Rollbacks due to Timeout (per minute)	1-11
Active Transactions.....	1-11
JVM Metrics	1-11
JVM Class Loading	1-11
Classes Currently Loaded.....	1-12
Classes Loaded (since startup).....	1-12
Classes Unloaded (since startup).....	1-12
JVM Compilation Metrics	1-12
Compilation Time (since startup) (ms)	1-12
CPU Usage (%)	1-12

Active Daemon Threads.....	1-12
Heap Usage (MB)	1-12
JDK Vendor	1-12
JDK Version.....	1-13
JVM Garbage Collectors	1-13
Garbage Collector - Garbage Collection Invocations (since startup)	1-13
Garbage Collector - Execution Time (since startup) (ms)	1-13
JVM Memory Pools	1-13
Memory Pool - Current Usage (KB)	1-13
Memory Pool - Peak Usage (KB).....	1-13
JVM Memory Usage	1-13
Heap Memory Usage (KB).....	1-13
Non-Heap Memory Usage (KB).....	1-13
Objects Pending Finalization.....	1-13
Nursery Size (MB)	1-14
JVM Runtime Metrics	1-14
VM Name	1-14
VM Version	1-14
JVM ThreadContention	1-14
Blocked Count	1-14
Blocked Time (since startup) (seconds).....	1-14
CPU Usage (since startup) (seconds)	1-14
Lock Name	1-14
Thread State	1-14
Wait Count	1-15
Wait Time (since startup) (seconds)	1-15
Active Threads	1-15
JVM Threads	1-15
Active Daemon Threads.....	1-15
Deadlocked Threads	1-15
Peak Threads.....	1-15
Active Threads.....	1-15
Threads Started (since startup)	1-15
Response	1-15
Status.....	1-15
Server Metrics	1-16
Listen Port	1-16
Active Sockets.....	1-16
Start Time	1-16
State	1-16
Server Datasource Metrics	1-16
Server Datasource - Available Connections	1-16
Server Datasource - Connections Created (per minute).....	1-17
Server Datasource - Connections in Use.....	1-17
Server Datasource - Connection Pool Size.....	1-17
Server Datasource - Connection Requests (per minute).....	1-17
Server Datasource - Successful Connections (%).....	1-17

Server Datasource - Connection Requests Waiting.....	1-17
Server Datasource - Connection Requests that Waited (per minute).....	1-17
Server Datasource - Connection Wait Successes (%).....	1-17
Server Datasource - Connection Wait Successes (per minute).....	1-17
Server Datasource - Statements Added to Cache (per minute).....	1-17
Server Datasource - Cached Statements not Used (per minute).....	1-17
Server Datasource - Cached Statements Used (%).....	1-17
Server Datasource - Cached Statements Used (per minute).....	1-18
Server Datasource - Statement Cache Size.....	1-18
Server Overview Metrics	1-18
JDBC Connection Creates (per minute).....	1-18
Open JDBC Connections.....	1-18
Cached Beans.....	1-18
Cache Accesses (per minute).....	1-18
Bean Activations (per minute).....	1-18
Cache Hits (%).....	1-18
Cache Misses (per minute).....	1-18
Bean Accesses (per minute).....	1-18
Bean Successes (%).....	1-18
Free Bean Instances.....	1-19
Bean Destroys (per minute).....	1-19
Bean Access Failures (per minute).....	1-19
Beans in Use.....	1-19
Bean Transaction Commits (%).....	1-19
Bean Transaction Commits (per minute).....	1-19
Bean Transaction Rollbacks (per minute).....	1-19
Bean Transaction Timeouts (per minute).....	1-19
Committed JTA Transactions (per minute).....	1-19
JTA Transaction Rollbacks (per minute).....	1-19
Active JTA Transactions.....	1-19
MDB Messages (per minute).....	1-20
Pending Messages.....	1-20
Current Messages.....	1-20
JMS Servers.....	1-20
Requests (per minute).....	1-20
Request Processing Time (ms).....	1-20
Active Sessions.....	1-20
Work Manager Pending Requests.....	1-20
Work Manager Requests (per minute).....	1-20
Server Work Manager Metrics	1-20
Server Work Manager - Work Manager Pending Requests.....	1-20
Server Work Manager - Work Manager Requests (per minute).....	1-20
Server Work Manager - Work Manager Stuck Threads.....	1-21
Servlet/JSP Metrics	1-21
Servlet/JSP - Reloads (per minute).....	1-21
Servlet/JSP - Requests (per minute).....	1-21
Servlet/JSP - Request Processing Time (ms).....	1-21

Servlet Rollup	1-21
Servlet - Active Invocations	1-21
Servlet - Current Client Processing Time	1-21
Servlet - Average Execution Time	1-22
Servlet - Total Execution Time	1-22
Servlet - Invocations Per Second	1-22
Servlet - Invocation Total Count	1-23
Web Module Metrics	1-23
Web Module - Requests (per minute)	1-23
Web Module - Request Processing Time (ms)	1-24
Web Module - Active Sessions	1-24
Java JEE Web Service Endpoints	1-24
Dispatch Time Total	1-24
Execution Time Total	1-24
Invocation Count	1-24
Response Count	1-24
Response Error Count	1-24
Response Time Total	1-24
Work Manager Metrics	1-24
Work Manager Pending Requests	1-24
Work Manager Requests (per minute)	1-24
Work Manager Stuck Threads	1-25

2 Oracle Coherence Metrics

Cache Performance - Aggregated Over Nodes	2-1
Cache Hits Delta Sum	2-1
Cache Hits Millis Delta Sum	2-2
Cache Misses Delta Sum	2-2
CacheMissesMillisDeltaSum	2-2
StoreReadMillisDeltaSum	2-3
StoreReadsDeltaSum	2-3
StoreWriteMillisDeltaSum	2-3
StoreWritesDeltaSum	2-4
Total Gets Delta Sum	2-4
TotalGetsMillisDeltaSum	2-5
TotalPutsDeltaSum	2-5
TotalPutsMillisDeltaSum	2-6
Cache Configuration	2-6
Batch Factor	2-6
Description	2-6
Expiry Delay	2-6
Flush Delay	2-7
High Units	2-7
Low Units	2-7
Persistence Type	2-7
Queue Delay	2-8
Refresh Factor	2-8

Refresh Time	2-8
Requeue Threshold	2-8
Cache Performance	2-9
Average Gets Millis.....	2-9
Average Get Millis Delta.....	2-9
Average Hit Millis.....	2-9
Average Hit Millis Delta	2-9
Average Miss Millis	2-10
Average Miss Millis Delta.....	2-10
Average Put Millis	2-10
Average Put Millis Delta	2-10
Cache Hits	2-11
Cache Hits Delta.....	2-11
Cache Hits Millis	2-11
Cache Hits Millis Delta.....	2-12
Cache Misses.....	2-12
Cache Misses Delta	2-12
Cache Misses Millis.....	2-13
Cache Misses Millis Delta	2-13
Hit Probability	2-13
Queue Size.....	2-13
Size	2-14
Store Average Batch Size.....	2-14
Store Average Read Millis	2-14
Store Failures	2-14
Store Failures Delta.....	2-15
StoreRead Millis	2-15
Store Read Millis Delta	2-15
Store Reads.....	2-15
Store Reads Delta	2-15
Store Write Millis	2-16
Store Write Millis Delta	2-16
Store Writes.....	2-16
Store Writes Delta	2-17
Total Gets.....	2-17
Total Gets Delta	2-17
Total Gets Millis	2-18
Total Gets Millis Delta	2-18
Total Puts.....	2-18
Total Puts Delta	2-19
Total Puts Millis.....	2-19
Total Puts Millis Delta	2-19
Units	2-20
Cluster	2-20
Cluster Name	2-20
Cluster Size.....	2-20
License Mode	2-20

Local Member Id	2-21
Members	2-21
Oldest Member ID.....	2-21
Running	2-21
Cluster Aggregated	2-22
Cache Size.....	2-22
Cache Size Delta	2-22
Cluster Size.....	2-22
Cluster Size Delta	2-23
Publisher Success Rate.....	2-23
Receiver Success Rate	2-24
Cluster Alerts	2-24
Cluster Name	2-24
Cluster Nodes	2-24
Change in No. of Nodes Since Last Collection	2-25
Change in Number of Nodes	2-25
Connection	2-26
Connection Time Millis	2-26
Outgoing Byte Backlog.....	2-26
Outgoing Message Backlog	2-26
Refresh Time	2-27
Remote Address	2-27
Remote Port.....	2-27
Timestamp.....	2-27
Total Bytes Received.....	2-28
Total Bytes Sent	2-28
Total Messages Received.....	2-29
Total Messages Sent	2-29
Connection Manager	2-29
Connection Count	2-29
Host IP	2-29
Incoming Buffer Pool Capacity	2-30
Incoming Buffer Pool Size.....	2-30
Outgoing Buffer Pool Capacity	2-30
Outgoing Buffer Pool Size.....	2-30
Outgoing Byte Backlog.....	2-30
Outgoing Message Backlog	2-31
RefreshTime	2-31
Total Bytes Received.....	2-31
Total Bytes Sent	2-32
Total Messages Received.....	2-32
Total Messages Sent	2-32
Garbage Collector after GC	2-33
Name.....	2-33
Node ID	2-33
GC Thread Count	2-33
Start Time (ms)	2-33

End Time (ms)	2-34
Key of Memory Usage after GC	2-34
Committed of Memory Usage after GC	2-34
Init of Memory Usage after GC	2-34
Max of Memory Usage after GC	2-34
Used of Memory Usage after GC	2-35
Garbage Collector Before GC	2-35
Name	2-35
Node ID	2-35
GC Thread Count	2-35
Start Time (ms)	2-36
End Time (ms)	2-36
Key of Memory Usage Before GC	2-36
Committed of Memory Usage Before GC	2-37
Init of Memory Usage Before GC	2-37
Max of Memory Usage Before GC	2-38
Used of Memory Usage Before GC	2-38
Garbage Collector Collection	2-39
Name	2-39
Node ID	2-39
Collection Count	2-39
Garbage Collection Rate	2-40
Collection Time	2-41
HTTP Session	2-41
Application ID	2-41
Node ID	2-41
Collection Class Name	2-41
Factory Class Name	2-41
Local Attribute Cache Name	2-42
Number of Local Attributes in Local Attribute Cache	2-42
Local Session Cache Name	2-42
Number of Local Attributes in Local Session Cache	2-42
Overflow Average Size	2-43
Overflow Cache Name	2-43
Overflow Max Size	2-43
Overflow Threshold	2-43
Updates to Attributes Stored in Overflow Cache	2-43
Updates to Attributes Stored in Overflow Cache Per Min -Per Min	2-44
Servlet Context Cache Name	2-44
Servlet Context Name	2-44
Session Average Lifetime	2-44
Session Cache Name	2-44
Session ID Length	2-45
Session Object Maximum Size (bytes)	2-45
Number of Http Session Objects	2-45
Session Timeout	2-45
Updates to Object Store in Session Cache -Per Node	2-45

Updates to Object Store in Session Cache Per Min -Per Node	2-46
HTTP Session - Aggregated Over Nodes	2-46
Application ID	2-46
No. of Attributes in Local Attribute Cache	2-47
No. of Sessions in Local Session Cache	2-47
Updates to Attributes Stored in Overflow Cache.....	2-48
Updates to Attributes Stored in Overflow Cache Per Min	2-48
Updates to Object Store in Session Cache.....	2-49
Updates to Object Store in Session Cache Per Min	2-50
Memory Performance	2-50
Node ID	2-50
Heap Memory Committed for JVM to use (bytes)	2-50
Heap Memory that JVM initially Requests from OS (bytes)	2-51
Maximum Heap Memory That Can Be Used (bytes).....	2-51
Heap Memory Used (bytes).....	2-52
Non Heap Memory Committed for JVM to use (bytes)	2-52
Non Heap Memory that JVM initially Requests from OS (bytes).....	2-53
Non Heap Memory Used (bytes).....	2-53
Object Pending Finalization Count	2-54
Verbose	2-55
Node Performance - Aggregate Over Caches.....	2-55
Total Gets Sum.....	2-55
Total Gets Sum Delta	2-55
Total Puts Sum.....	2-56
Total Puts Sum Delta	2-56
Node Configuration	2-56
Buffer Publish Size	2-56
Buffer Receive Size	2-57
Burst Count	2-57
Burst Delay	2-57
Flow Control Enabled.....	2-57
Logging Format.....	2-58
Logging Level	2-58
Logging Limit	2-58
Machine Id.....	2-58
Machine Name.....	2-59
Member Name	2-59
Memory Max (MB).....	2-59
Multicast Address	2-59
Multicast Enabled.....	2-60
Multicast Port.....	2-60
Multicast Threshold	2-60
Multicast TTL.....	2-60
Nack Enabled	2-60
Process Name.....	2-61
Product Edition.....	2-61
Rack Name	2-61

RefreshTime	2-61
Resend Delay	2-62
Role Name	2-62
Site Name	2-62
Timestamp.....	2-62
Traffic Jam Count	2-63
Traffic Jam Delay	2-63
Unicast Address	2-63
Unicast Port.....	2-63
Node Performance	2-64
Cpu Count.....	2-64
Machine Id.....	2-64
Machine Name.....	2-64
Member Name	2-64
Memory Available (MB).....	2-65
Packet Delivery Efficiency	2-65
Packets Bundled	2-65
Packets Bundled (Delta Time).....	2-66
Packets Received	2-66
Packets Received (Delta Time).....	2-66
Packets Repeated.....	2-66
Packets Repeated (Delta Time).....	2-67
Packets Resent.....	2-67
Packets Resent (Delta Time)	2-67
Packets Resent Early	2-67
Packets Resent Early (Delta Time).....	2-68
Packets Resent Excess	2-68
Packets Resent Excess (Delta Time).....	2-68
Packets Sent.....	2-68
Packets Sent (Delta Time)	2-68
Process Name.....	2-69
Publisher Success Rate.....	2-69
Publisher Success Rate (Delta Time)	2-69
Rack Name	2-70
Receiver Success Rate (Delta Time).....	2-70
Receiver Success Rate Delta.....	2-70
RefreshTime	2-71
Role Name	2-71
Send Queue Size	2-71
Site Name	2-72
Tcp Ring Failures	2-72
Tcp Ring Failures (Delta Time)	2-73
Tcp Ring Timeouts	2-73
Tcp Ring TimeOuts (Delta Time).....	2-73
Up Since.....	2-73
Weakest Channel.....	2-74
Point to Point	2-74

Deferred Packets.....	2-74
Deferring.....	2-74
LastIn.....	2-74
LastOut	2-75
Last Slow	2-75
Outstanding Packets	2-75
Paused.....	2-75
Pause Rate	2-76
Publisher Success Rate.....	2-76
Publisher Success Rate Delta	2-76
ReceiverSuccessRate	2-76
ReceiverSuccessRateDelta.....	2-77
Threshold.....	2-77
Viewed Member Id	2-77
Operating System	2-77
Node ID	2-77
Architecture.....	2-78
Available Processors	2-78
Committed Virtual Memory Size (bytes).....	2-78
Free Physical Memory Size (bytes).....	2-79
Free Swap Space Size (bytes).....	2-79
Max File Descriptor Count.....	2-80
Name	2-80
Open File Descriptor Count.....	2-80
Process Cpu Time.....	2-81
Total Physical Memory Size (bytes)	2-81
Total Swap Space Size (bytes)	2-81
Version.....	2-81
Resource Usage	2-82
Memory Usage (MB).....	2-82
Response	2-82
Status.....	2-82
Runtime	2-82
Row ID	2-82
Node ID	2-83
Input Arguments.....	2-83
VM Name	2-83
VM Vendor.....	2-83
VM Version	2-84
Service Configuration	2-84
Partitions All	2-84
Requests Timeout Millis.....	2-84
Running	2-84
Statistics	2-84
High Availability Status.....	2-85
Storage Enabled	2-85
Task Hung Threshold Millis.....	2-85

Task Timeout Millis	2-85
Thread Count.....	2-86
Type.....	2-86
Service Performance	2-86
Backup Count	2-86
Owned Partitions Backup	2-86
Owned Partitions Primary.....	2-87
Partitions Endangered.....	2-87
Partitions Unbalanced	2-87
Partitions Vulnerable.....	2-87
Refresh Time	2-88
Request Average Duration	2-88
Request Max Duration	2-88
Request Pending Count.....	2-88
Request Pending Duration.....	2-88
Request Timedout Count	2-89
Request Timeout Count Delta.....	2-89
Request Total Count	2-89
Request Total Count Delta.....	2-90
Status HA	2-90
Storage Enabled Count.....	2-91
Task Average Duration	2-91
Task Backlog	2-91
Task Count	2-92
Task Count Delta.....	2-92
Task Hung Count.....	2-92
Task Hung Duration.....	2-92
Task Hung Task Id.....	2-92
Task Max Backlog	2-93
Task Timeout Count	2-93
Task Timeout Count Delta.....	2-93
Thread Abandoned Count.....	2-93
Thread Abandoned Count Delta	2-94
Thread Average Active Count	2-94
Thread Count.....	2-94
Thread Idle Count.....	2-95
Thread Idle Count Delta.....	2-95
Eviction Count.....	2-95
Insert Count	2-96
Listener Filter Count.....	2-96
Listener Key Count	2-96
Listener Registrations	2-96
Locks Granted.....	2-97
Locks Pending.....	2-97
Remove Count	2-97
Service Performance - Aggregated Over Nodes	2-97
Backup Count	2-97

Owned Partitions Backup	2-98
Owned Partitions Primary	2-98
Partitions Endangered	2-98
Partitions Unbalanced	2-98
Partitions Vulnerable	2-99
Request Average Duration	2-99
Request Max Duration.....	2-99
Request Pending Count.....	2-99
Request Pending Duration.....	2-99
Request Timeout Count	2-100
Request Timeout Count Delta	2-100
Request Total Count	2-100
Request Total Count Delta	2-101
Storage Enabled Count.....	2-101
Task Average Duration	2-101
Task Backlog	2-102
Task Count	2-102
Task Count Delta.....	2-102
Task Hung Count	2-102
Task Hung Duration	2-103
Task Max Backlog	2-103
Task Timeout Count	2-103
Task Timeout Count Delta.....	2-103
Thread Abandoned Count.....	2-103
Thread Abandoned Count Delta	2-104
Thread Average Active Count	2-104
Thread Count.....	2-104
Thread Idle Count	2-105
Thread Idle Count Delta.....	2-105
Storage Manager	2-105
Events Dispatched - Per Node	2-106
Eviction Count - Per Node.....	2-106
Insert Count - Per Node	2-106
Listener Filter Count - Per Node.....	2-106
Listener Key Count - Per Node	2-107
Listener Registrations - Per Node.....	2-107
Locks Granted - Per Node.....	2-107
Locks Pending - Per Node	2-107
Remove Count - Per Node	2-108
Store Manager - Aggregated Over Nodes	2-108
Events Dispatched.....	2-108
Eviction Count Sum	2-108
Insert Count	2-108
Listener Filter Count Sum	2-109
Listener Key Count Sum	2-109
Listener Registrations Sum	2-109
Locks Granted Sum.....	2-109

Locks Pending Sum	2-110
Remove Count Sum	2-110
Resource Usage.....	2-110
Memory Usage (MB).....	2-110
Threading	2-110
Node ID	2-110
Thread Count.....	2-110
Peak Thread Count	2-111

3 HTTP Server

OHS Process Metrics	3-1
Heap Usage (MB)	3-1
Oracle HTTP Server General Metrics	3-1
Start Time (ms since Epoch)	3-1
OHS Module Metrics	3-2
Active Requests for a Module	3-2
Request Handling Time, for a Module (seconds).....	3-2
Request Handling Throughput, for a Module (requests per second)	3-2
OHS Response Code Metrics	3-3
HTTP 4xx errors	3-3
HTTP 5xx errors	3-3
OHS Server Metrics	3-3
Busy Processes	3-4
Percentage of Busy Processes	3-4
Active HTTP Connections	3-4
Connection Duration (seconds).....	3-5
Error Rate (%)	3-5
Idle Processes	3-6
Active HTTP Requests.....	3-6
Request Processing Time (seconds).....	3-7
Request Processing Time (milliseconds).....	3-7
Request Throughput (requests per second)	3-7
Response Data Processed (KB per response)	3-7
Response Data Throughput (KB per second).....	3-8
OHS Virtual Host Metrics	3-8
Active Requests for a Virtual Host	3-8
Request Processing Time for a Virtual Host (seconds).....	3-9
Request Throughput for a Virtual Host (requests per second)	3-10
Response Data Processed For a Virtual Host (KB per response)	3-10
Response Data Throughput For a Virtual Host (KB per second).....	3-11
Virtual Host Type.....	3-11
Resource Usage	3-11
CPU Usage (%)	3-11
CPU Idle Time (%)	3-12
Other CPU Usage (%)	3-12
Memory Usage (MB).....	3-12
Free Memory (MB).....	3-13

Other Memory Usage (MB)	3-13
Total Memory (MB)	3-13
Memory Usage (%)	3-14
Free Memory (%).....	3-14
Other Memory Usage (%)	3-14
Start Time (ms since Epoch)	3-15
Up Time (ms since Epoch)	3-15
Response	3-15
Response - Status.....	3-15

4 Java EE Application

ADFC Taskflow Metrics	4-1
Total Active Taskflows	4-1
Active Taskflows	4-1
Active throughput.....	4-2
Active Taskflows	4-2
Total Entered Taskflows.....	4-2
Entered Taskflows.....	4-2
Invoke Taskflows	4-3
Invoked Taskflows completed	4-3
Taskflow Invoke time	4-3
Average Time.....	4-4
Maximum Time	4-4
Minimum Time.....	4-4
Module Removals	4-4
Metric Summary	4-4
Pool Checkouts	4-5
Metric Summary	4-5
Average Creation time(ms)	4-5
Metric Summary	4-5
User Action.....	4-5
Free Instances	4-5
Metric Summary	4-5
Maximum Creation Time (ms)	4-5
Metric Summary	4-5
Request Count	4-6
Metric Summary	4-6
Available Instances in last 5 Minutes	4-6
Metric Summary	4-6
User Action.....	4-6
Available Instances in last 1 min	4-6
Metric Summary	4-6
User Action.....	4-6
Unavailable Instances in last 5 Minutes	4-6
Metric Summary	4-7
User Action.....	4-7
Unavailable Instances in last 1 Minute	4-7

Metric Summary.....	4-7
User Action	4-7
Activations	4-7
Metric Summary.....	4-7
User Action	4-7
Pool check Ins	4-7
Metric Summary.....	4-7
User Action	4-8
Pool Check Outs	4-8
Metric Summary.....	4-8
User Action	4-8
Pool Check Out Failures	4-8
Metric Summary.....	4-8
Referenced Modules Reused	4-8
Metric Summary.....	4-8
Reference Modules Reused	4-8
Metric Summary.....	4-8
Unreference Modules Recycled	4-9
Metric Summary.....	4-9
Datasource Metrics	4-9
Datasource - Available Connections	4-9
Datasource - Connections Created (per minute)	4-9
Datasource - Connections in Use	4-9
Datasource - Connection Leaks (per minute).....	4-9
Datasource - Connection Pool Size.....	4-9
Datasource - Connection Refresh Failures (per minute)	4-9
Datasource - Connection Request Failures (per minute)	4-10
Datasource - Connection Requests (per minute).....	4-10
Datasource - Successful Connections (%).....	4-10
Datasource - Unavailable Connections	4-10
Datasource - Connection Requests Waiting	4-10
Datasource - Failed Waiting Connection Requests (per minute).....	4-10
Datasource - Connection Requests that Waited (per minute)	4-10
Datasource - Connection Wait Successes (%).....	4-10
Datasource - Connection Wait Successes (per minute)	4-11
Datasource - State.....	4-11
Datasource - Statements Added to Cache (per minute).....	4-11
Datasource - Statements Discarded from Cache (per minute)	4-11
Datasource - Cached Statements not Used (per minute)	4-11
Datasource - Cached Statements Used (%)	4-11
Datasource - Cached Statements Used (per minute)	4-11
Datasource - Statement Cache Size.....	4-12
EJB Module Metrics	4-12
EJB Module - Cached Beans.....	4-12
EJB Module - Bean Activations (per minute)	4-12
EJB Module - Cache Hits (%).....	4-12
EJB Module - Cache Misses (per minute)	4-12

EJB Module - Bean Accesses (per minute).....	4-13
EJB Module - Bean Access Successes (%).....	4-13
EJB Module - Free Bean Instances.....	4-13
EJB Module - Bean Destroys (per minute).....	4-13
EJB Module - Bean Access Failures (per minute)	4-14
EJB Module - Beans In Use	4-14
EJB Module - Bean Transaction Commits (per minute)	4-14
EJB Module - Bean Transaction Rollbacks (per minute)	4-14
EJB Module - Bean Transaction Timeouts (per minute).....	4-15
EJB Transaction Metrics	4-15
EJB Transaction - Bean Transaction Commits (%).....	4-15
EJB Transaction - Bean Transaction Commits (per minute)	4-15
EJB Transaction - Bean Transaction Rollbacks (per minute).....	4-15
EJB Transaction - Bean Transaction Timeouts (per minute).....	4-15
Overview Metrics	4-15
Cached Beans	4-15
Cache Accesses (per minute)	4-15
Bean Activations (per minute)	4-16
Cache Hits (%)	4-16
Cache Misses (per minute).....	4-16
Bean Accesses (per minute)	4-16
Bean Successes (%)	4-17
Free Bean Instances.....	4-17
Bean Destroys (per minute)	4-17
Bean Access Failures (per minute).....	4-17
Beans In Use.....	4-17
Bean Transaction Commits (%).....	4-18
Bean Transaction Commits (per minute).....	4-18
Bean Transaction Rollbacks (per minute).....	4-18
Bean Transaction Timeouts (per minute)	4-18
MDB Messages (per minute)	4-19
Requests (per minute).....	4-19
Request Processing Time (ms).....	4-19
Active Sessions	4-19
Work Manager Pending Requests	4-19
Work Manager Requests (per minute).....	4-20
Servlet/JSP Metrics	4-20
Servlet/JSP - Reloads (per minute).....	4-20
Servlet/JSP - Requests (per minute).....	4-20
Servlet/JSP - Request Processing Time (ms).....	4-20
Status Metrics	4-21
Server Name.....	4-21
State	4-21
Web Module Metrics	4-21
Web Module - Requests (per minute)	4-21
Web Module - Request Processing Time (ms)	4-22
Web Module - Active Sessions	4-22

MDS Metrics	4-22
Metadata Object Gets Per Second	4-22
Metadata Object Get Processing Time (seconds).....	4-22
IOs Per Document Get.....	4-23
IOs Per MO Content Get	4-24
IOs Per Metadata Object Get	4-24
Metadata Updates Per Second	4-25
Metadata Update Processing Time (seconds).....	4-25
UpDown Status	4-25

5 Oracle Service Bus

Oracle Service Bus	5-1
Flow Component Resource Details Metrics	5-1
Average Response Time (Flow Component)	5-1
Open Instance Count (FLOW_COMPONENT).....	5-2
Error Instance Count (Flow Component).....	5-3
End Point URI Resource Details Metrics	5-3
Average Response Time (End Point URI)	5-3
Open Instance Count (End Point URI).....	5-4
Error Instance Count (End Point URI).....	5-5
End Point Operation Resource Details Metrics	5-5
Average Response Time (End Point Operation)	5-5
Open Instance Count (End Point Operation).....	5-6
Error Instance Count (End Point Operation)	5-7
OSB Service Details Metrics	5-7
Contains Sensitive Data.....	5-8
Created User	5-8
Created Time.....	5-8
End Point URI.....	5-8
Last Modified User	5-8
Last Modified Time.....	5-8
Protocol.....	5-8
Status	5-8
Version	5-8
OSB Service Metrics	5-8
Average Response Time.....	5-8
Total Average Response Time.....	5-9
Open Instance Count	5-10
Total Open Instance Count.....	5-10
Error Instance Count.....	5-11
Total Error Instance Count	5-11
Failure Rate	5-12
Maximum Response Time	5-12
Minimum Response Time.....	5-13
Service Name	5-13
Path.....	5-13
Project Name.....	5-13

Service Type.....	5-14
WebService Security Violation Instance Count	5-14
Total WebService Security Violation Instance Count	5-14
OSB Service Dependency Metrics	5-15
Dependent Service ID	5-15
Response	5-15
Status.....	5-15
SLA Alert Metrics	5-16
Service Name	5-16
Project Name.....	5-16
SLA Severity Critical Alerts.....	5-16
Total SLA Severity Critical Alerts.....	5-17
SLA Severity Fatal Alerts	5-17
Total SLA Severity Fatal Alerts	5-18
SLA Severity Major Alerts	5-18
Total SLA Severity Major Alerts	5-18
SLA Severity Minor Alerts.....	5-19
Total SLA Severity Minor Alerts.....	5-19
SLA Severity Warning Alerts	5-19
Total SLA Severity Warning Alerts	5-20
Service Type.....	5-20

6 Oracle SOA Infra

B2B Active Entities Metrics	6-1
Number of Active Agreements	6-1
Number of Active Documents	6-1
Number of Active Trading Partners.....	6-1
B2B Document Metrics Table	6-2
Number of Inbound Messages	6-2
Number of inbound message faults	6-2
Inbound Message Size (KB)	6-2
Inbound Messages processing time.....	6-2
B2B Document Name	6-2
Number of Outbound Messages.....	6-3
Number of outbound message faults.....	6-3
Outbound Message Size (KB).....	6-3
Outbound Messages processing time	6-3
B2B Inbound Endpoint Metrics	6-3
B2B Inbound Endpoint Status	6-4
B2B Outbound Endpoint Metrics	6-4
B2B Outbound Endpoint Status	6-4
B2B Trading Partner Metrics	6-4
Number of inbound messages	6-4
Number of inbound message faults	6-4
Inbound Message Size (KB)	6-5
Processing time of Inbound Messages	6-5
Trading partner name.....	6-5

Number of outbound messages	6-5
Number of outbound message faults.....	6-5
Outbound Message Size (KB).....	6-6
Processing time of outbound messages	6-6
BPEL Engine Requests Rate Metric Table	6-6
Active request throughput in the last 5 minutes	6-6
Last active request value	6-6
Scheduled request throughput in the last 5 minutes	6-6
Last scheduled request value	6-7
Business Rules Engine Cache Metrics Table	6-7
Total Access Count since server startup	6-7
Last value of access count	6-7
Total Count since server startup	6-7
Last value of total count	6-8
Business Rule Engine Request Breakdown Rate Metric Table	6-8
Request count since server startup	6-8
Throughput of requests in the last 5 minutes	6-8
Last value of request.....	6-8
Average request processing time.....	6-8
Number of completed requests.....	6-9
Response.....	6-9
UpDown Status	6-9
SOA Infrastructure Bindings Rate Metric Table.....	6-9
Active request throughput in the last 5 minutes	6-9
Number of faults since server startup.....	6-10
Average request processing time.....	6-10
Total number of processed requests since server startup	6-10
SOA Composite Bindings/References Rate Metric	6-10
Total number of faults since server start	6-10
Throughput of faults in the last 5 minutes	6-11
Average Incoming Messages Processing Time.....	6-11
Total completed incoming messages since server start.....	6-11
Incoming messages throughput in the last 5 minutes	6-11
Total Incoming Messages Processing Time.....	6-11
Total Incoming Message processing time.....	6-12
Total number of incoming messages since server start.....	6-12
Incoming message throughput in the last 5 minutes.....	6-12
Average Outbound Messages Processing Time	6-12
Total completed outgoing messages since server start.....	6-12
Outbound messages throughput in the last 5 minutes.....	6-13
Total Outbound Messages Processing Time since server start	6-13
Total Outbound Message processing time	6-13
Total number of outgoing messages since server start.....	6-13
Outbound message events throughput in the last 5 minutes.....	6-13
BPEL Dispatcher Rate Metric Table.....	6-14
Total number of messages processed per thread since server startup.....	6-14
Throughput of total messages in the last 5 minutes	6-14

Last value of total message	6-14
Throughput of average thread allocation time in the last 5 minutes	6-14
Last value of average thread allocation time	6-15
Throughput of average thread lifetime in the last 5 minutes	6-15
Last value of average thread lifetime	6-15
Throughput of maximum threads in the last 5 minutes	6-15
Last value of maximum thread	6-15
Throughput of minimum threads in the last 5 minutes	6-16
Last value of minimum thread.....	6-16
Number of active threads since server startup	6-16
Throughput of thread count in the last 5 minutes	6-16
Last value of thread count	6-17
Number of thread highs since server startup	6-17
Throughput of thread high in the last 5 minutes	6-17
Last value of thread high	6-17
Number of pending threads since server startup.....	6-17
Throughput of pending threads in the last 5 minutes.....	6-18
Last value of pending thread.....	6-18
Total number of threads since server startup	6-18
Throughput of total threads in the last 5 minutes.....	6-18
Last value of total thread metric	6-18
BPMN Dispatcher Rate Metric Table	6-19
Last value of maximum thread	6-19
Number of active threads since server startup	6-19
Throughput of thread count in the last 5 minutes	6-19
SOA Mediator Engine Cache	6-19
Last value of metadata cache.....	6-19
Last value of transformation cache.....	6-19
SOA Mediator Engine Requests.....	6-20
Callback Requests	6-20
Delay Requests	6-20
Dequeue Requests.....	6-20
Timeout Requests.....	6-20
SOA Mediator Engine Request Breakdown Rate Metric Table	6-21
enqueue-phase.active	6-21
Avg Enqueue Time (ms)	6-21
Enqueued.....	6-21
Throughput of Enqueue phase.....	6-21
Total Enqueue Time (ms).....	6-21
invoke-phase.active.....	6-22
Avg Invoke Time (ms).....	6-22
Invoked	6-22
Throughput of Invoke phase	6-22
Total Invoke Time (ms)	6-22
invokeOneway-phase.active.....	6-23
Avg Oneway Invoke Processing Time (ms).....	6-23
Oneway Invocations	6-23

Throughput of Oneway Invocations	6-23
Total Oneway Invocation Time (ms).....	6-24
publish-phase.active	6-24
Avg Publication Time (ms).....	6-24
Publications.....	6-24
Throughput of Publication phase	6-24
Total Publication Time (ms)	6-25
transformation-phase.active	6-25
Avg Transformation Time (ms)	6-25
Transformations	6-25
Throughput of Transformations	6-25
Total Transformation Time (ms).....	6-26
SOA Infra Mesh	6-26
Number of active messages in the mesh.....	6-26
Total number of messages in the mesh since server startup	6-26
Number of faulted messages in the mesh	6-26
Throughput of messages in the mesh in the last 5 minutes.....	6-27
SOA Infra Mesh Message Count Table	6-27
Total number of messages in the mesh since server startup	6-27
Mesh Name	6-27
Throughput of faulted asynchronous messages in the last 5 minutes.....	6-27
Number of active asynchronous messages	6-27
Average processing time of asynchronous messages.....	6-28
Number of completed asynchronous messages	6-28
Throughput of asynchronous messages in the last 5 minutes.....	6-28
Total processing time of asynchronous messages.....	6-28
Throughput of faulted synchronous messages in the last 5 minutes	6-29
Number of active synchronous messages	6-29
Average processing time of synchronous messages.....	6-29
Number of completed synchronous messages	6-29
Throughput of synchronous messages in the last 5 minutes.....	6-29
Total processing time of synchronous messages.....	6-30
SOA Message Processing Rate Metric Table	6-30
Total number of active requests since server startup	6-30
Throughput of the active requests in the last 5 minutes	6-30
Last value of active request	6-30
Total number of BPEL messages since startup	6-31
Engine Name	6-31
Decision Service Engine	6-31
Total number of decision service messages since startup.....	6-31
Engine Name	6-31
Number of faults since server startup.....	6-31
Throughput of faulted requests in the last 5 minutes.....	6-32
Rate of processing faulted requests.....	6-32
Total number of mediator messages since startup.....	6-32
Engine Name	6-32
Total number of messages since server startup.....	6-32

Total number of asynchronous requests since server startup	6-33
Throughput of asynchronous invocations in the last 5 minutes	6-33
Rate of processing asynchronous invocations	6-33
Average asynchronous message processing time since server startup.....	6-33
Total number of synchronous requests since server startup	6-33
Synchronous invocations throughput in the last 5 minutes	6-34
Rate of processing synchronous invocations	6-34
Average synchronous message processing time since server startup.....	6-34
Workflow Service Engine.....	6-34
Total number of workflow messages since startup	6-34
Engine Name.....	6-35
Total Composite Instances	6-35
Total number of instances since server startup	6-35
Domain Name.....	6-35
Total Composite Faults	6-35
Total number of faults since server startup.....	6-35
Workflow Request Breakdown Rate Metrics	6-36
Number of invocations since server startup	6-36
Service Name	6-36
Number of active requests.....	6-36
Average request processing time.....	6-36
Number of completed requests.....	6-37

7 SOA Composite

SOA Composite B2B Inbound Metrics	7-1
InBound Message Count.....	7-1
Inbound Messages Faults.....	7-1
Inbound Messages Size	7-1
Inbound Message Processing Time	7-1
ServerName.....	7-1
Total Outbound Messages Count	7-1
SOA Composite B2B Outbound Metrics	7-1
Name.....	7-1
OutBound Message Count.....	7-1
Outbound Messages Faults.....	7-2
Outbound Messages Size	7-2
Outbound Message Processing Time	7-2
ServerName.....	7-2
Total Outbound Messages Count	7-2
BPEL Activity Time Distribution	7-2
Average Execution Time (ms)	7-2
Activity Count	7-2
Activity Name.....	7-2
Server Name.....	7-3
BPMN Activity Time Distribution	7-3
Average Execution Time (ms)	7-3
Activity Count	7-3

Activity Name.....	7-3
Server Name.....	7-4
SOA Composite - Response Metrics.....	7-4
ServerName.....	7-4
Composite Status.....	7-4
SOA Composite Bindings/References Rate Metric.....	7-4
Total number of faults since server start	7-4
Fault throughput in last 5 minutes	7-5
Average Incoming Messages Processing Time.....	7-5
Total Incoming Messages Processing Time.....	7-5
Total Incoming Message processing time.....	7-5
Total number of incoming messages since server start.....	7-5
Incoming message throughput in last 5 minutes	7-6
Average Outbound Messages Processing Time	7-6
Total Outbound Messages Processing Time since server start	7-6
Total Outbound Message processing time	7-6
Total number of outgoing messages since server start.....	7-6
Outbound message throughput in last 5 minutes.....	7-7
SOA Component References	7-7
Average Message Processing Time (secs) since server start up	7-7
Number of Faults	7-7
Number of Messages	7-7
Average Message Processing Time (secs) in last 5 mins	7-8
Reference Name.....	7-8
Component References.....	7-8
Component Type.....	7-8
SOA Component Rollup Metric.....	7-8
Business faults throughput in the last 5 minutes	7-9
Number of active faulted instances	7-9
Average processing time for faulted instances	7-9
Number of faulted instances	7-9
Throughput of faulted instances in the last 5 minutes	7-9
Total time spent on processing faulted instances.....	7-10
Total average faulted processing time	7-10
Instance throughput in the last 5 minutes.....	7-10
Server Name.....	7-10
Number of active successful instances.....	7-10
Average processing time for successful instances.....	7-11
Number of successful instances	7-11
Throughput of successful instances in the last 5 minutes.....	7-11
Total time spent on processing successful instances	7-11
Total average successful processing time.....	7-12
System faults throughput in the last 5 minutes	7-12
Total faults throughput in the last 5 minutes.....	7-12
SOA Composite - Rate Metrics.....	7-12
Instance throughput in the last 5 minutes.....	7-12
Asynchronous message event throughput in the last 5 minutes	7-12

Average asynchronous message Processing time	7-13
Asynchronous messages throughput in the last 5 minutes	7-13
Total asynchronous message processing time	7-13
Synchronous message event throughput in the last 5 minutes	7-13
Average synchronous message processing time	7-14
Synchronous messages throughput in the last 5 minutes	7-14
Total synchronous message processing time	7-14
Server Name.....	7-14
Mediator Case	7-14
Average processing time of invocations	7-15
Invocation throughput in the last 5 minutes	7-15
Total processing time for invocations	7-15
Failed filtered messages throughput in the last 5 minutes	7-15
Average processing time of faulted cases.....	7-15
Number of faulted cases.....	7-16
Faulted case throughput in the last 5 minutes.....	7-16
Total processing time for faulted cases	7-16
Filtered messages throughput in the last 5 minutes	7-16
Invocation Count throughput in the last 5 minutes.....	7-16
Component Type.....	7-17
Composite Label.....	7-17
Case Target.....	7-17
Mediator Flow	7-17
Flow Type.....	7-17
Average processing time of successful cases	7-18
Number of successful cases	7-18
Successful case throughput in the last 5 minutes.....	7-18
Total processing time for successful cases.....	7-18
Mediator Flow Metric	7-19
Deferred message count throughput in last 5 minutes	7-19
Failed filter case throughput in last 5 minutes	7-19
Average faulted instance processing time.....	7-19
Faulted instances throughput in last 5 minutes.....	7-19
Total faulted instance processing time.....	7-19
Instance throughput in last 5 minutes	7-20
Component Type.....	7-20
Composite Label.....	7-20
Flow Type.....	7-20
Average successful instance processing time	7-21
Successful instances throughput in last 5 minutes.....	7-21
Total successful instance processing time	7-21
Total Composite Instances	7-21
Total Composite Faults	7-21

8 Forms

Forms Resource Usage	8-1
CPU Usage (%)	8-1

Total CPU Time (millisec)	8-2
instname.value	8-2
Total Private Memory (KB).....	8-2
Forms Application Deployments	8-3
Forms Application Context Root	8-3
instname.value.....	8-3
Forms Application Internal Name.....	8-3
Number of User Sessions	8-3
Application State	8-3
Application status	8-3
Forms Application Version.....	8-4
FormsUserSessions	8-4
appname.value	8-4
config.....	8-4
connect.value	8-4
CPU Usage (%)	8-4
CPU Time (millisec).....	8-4
dbname.value	8-5
executable.value	8-5
form.value	8-5
instname.value.....	8-5
ip.value	8-5
jsessionid.value.....	8-5
Private Memory (KB).....	8-5
servername.value	8-6
threadid.value.....	8-6
tnsentry.value	8-6
tracefile.value.....	8-6
tracing.value.....	8-6
user.value	8-6
JVM Controllers	8-7
CPU Usage (%)	8-7
CPU Time (millisec).....	8-7
Number of JVMs running for this JVM Controller	8-7
Private Memory (KB).....	8-7
UP Time (millisec).....	8-7
Number Of Sessions	8-7
Number Of Sessions	8-7
Response	8-8
instname.value.....	8-8
Status.....	8-8
Servlet ResponseTime	8-8
instname.value.....	8-9
Average Response Time (millisec).....	8-9
instname.value.....	8-9
Servlet Name.....	8-9
Response Time (millisec).....	8-9

Number Of Tasks	8-10
Total Time.....	8-10
Web Module Name.....	8-10

9 Portal

Portlets (All)	9-1
Cache Hit (%).....	9-1
Average Time (ms).....	9-1
Requests (per minute).....	9-1
Availability (%).....	9-2
Producer Type	9-2
Portlets (Top 5)	9-2
Cache Hit (%).....	9-2
Average Time (ms).....	9-2
Requests (per minute).....	9-2
Availability (%).....	9-3
Producer Type	9-3
Producers (All)	9-3
Cache Hit (%).....	9-3
Average Time (ms).....	9-3
Requests (per minute).....	9-4
Availability (%).....	9-4
Producer Type	9-4
Producers (Top 5)	9-5
Cache Hit (%).....	9-5
Average Time (ms).....	9-5
Requests (per minute).....	9-5
Availability (%).....	9-5
Producer Type	9-5
Portal Cache	9-6
Average Expire Hit Time (ms)	9-6
Expire Hit (%)	9-6
Average New Time (ms)	9-6
New (%)	9-7
Average Ping Hit Time (ms).....	9-7
Ping Hit (%).....	9-7
Requests (per minute).....	9-8
Average Stale Time (ms)	9-8
Stale (%)	9-8
Database Connection Pool Statistics	9-9
Average Fetch Time (ms)	9-9
Requests (per minute).....	9-9
Hit (%).....	9-9
New (%).....	9-10
Stale (%).....	9-10
Database Repository Response Code Statistics	9-11
Responses (per minute).....	9-11

Average Processing Time (ms).....	9-11
Page Engine Statistics	9-11
Average Queue Wait Time (ms)	9-12
Average Page Processing Time (ms)	9-12
Pages (per minute)	9-12
Average Page Metadata Fetch Time (ms).....	9-13
Page Metadata Fetches (per minute).....	9-13
Average Page Metadata Wait Time (ms).....	9-13
Queue Timeouts (per minute).....	9-14
HTTP200-HTTP299 (%)	9-14
HTTP300-HTTP399 (%).....	9-15
HTTP400-HTTP499 (%).....	9-15
HTTP500-HTTP599 (%).....	9-15
Timeout (%).....	9-16
WebCache Page Metadata Hits (%).....	9-16
WebCache Page Metadata Misses (%)	9-16
WebCache Page Metadata Non-Cacheables (%)	9-17
Response	9-17
UpDown Status	9-17

10 Discoverer

Discoverer	10-1
Discoverer Components	10-1
Total CPU Time (millisec).....	10-1
CPU Usage (%)	10-1
Memory Usage (MB).....	10-2
Total Private Memory (MB).....	10-2
Number of Sessions	10-2
Discoverer Sessions	10-2
Component Name.....	10-2
Connection Details	10-3
Total CPU Time (millisec).....	10-3
CPU Usage (%)	10-3
Memory Usage (MB).....	10-3
Port Number	10-3
Total Private Memory (MB).....	10-4
Session ID	10-4
SSO User Name	10-4
Discoverer Application	10-4
CPU Idle (%)	10-5
Total CPU Time (millisec).....	10-5
CPU Usage (%)	10-5
Available Physical Memory (MB).....	10-5
Host	10-5
Memory Usage (MB).....	10-6
Total Private Memory (MB).....	10-6
Number of Sessions	10-6

Total Physical Memory (MB).....	10-6
Discoverer Components	10-7
CPU Usage (%)	10-7
Memory Usage (MB).....	10-7
Number of Sessions	10-7
Total Discoverer CPU Usage	10-7
Total Discoverer CPU Usage (%)	10-7
Total Discoverer Memory Usage	10-8
Total Discoverer Memory Usage (MB)	10-8
Discoverer Plus Sessions	10-8
Component Type.....	10-8
CPU Usage (%)	10-8
DBUser@DB - EUL.....	10-9
Memory Usage (MB).....	10-9
OS Process ID.....	10-9
SSO User	10-9
Discoverer Portlet Provider Sessions	10-9
Component Type.....	10-9
CPU Usage (%)	10-9
DBUser@DB - EUL.....	10-9
Memory Usage (MB).....	10-9
OS Process ID.....	10-9
SSO User	10-9
Discoverer Sessions	10-10
Component Type.....	10-10
CPU Usage (%)	10-10
DBUser@DB - EUL.....	10-10
Memory Usage (MB).....	10-10
OS Process ID.....	10-10
SSO User	10-10
Total Number Of Discoverer Plus Sessions	10-10
Total Number Of Discoverer Plus Sessions	10-10
Total Number Of Discoverer Portlet Provider Sessions	10-10
Total Number Of Discoverer Portlet Provider Sessions.....	10-11
Total Number Of Discoverer Sessions	10-11
Total Number Of Discoverer Sessions	10-11
Total Number Of Discoverer Viewer Sessions	10-11
Total Number Of Discoverer Viewer Sessions	10-11
Discoverer Viewer Sessions	10-11
Component Type.....	10-11
CPU Usage (%)	10-11
DBUser@DB - EUL.....	10-11
Memory Usage (MB).....	10-11
OS Process ID.....	10-11
SSO User	10-12
Response	10-12
UpDown Status	10-12

11 Reports Bridge

Log Message Alerts	11-1
Log Monitoring Rules.....	11-1
Diagnostic Log Summary	11-2
Log Error Messages (over 15 minutes).....	11-2
Log Incident Error Messages (over 15 minutes).....	11-2
Log Notification Messages (over 15 minutes).....	11-3
Log Trace Messages (over 15 minutes)	11-3
Log Unknown Messages (over 15 minutes).....	11-4
Log Warning Messages (over 15 minutes)	11-4
Remote Elements Discovered	11-5
Host.value	11-5
Name.....	11-5
Discovery Time (ms).....	11-5
Type of Element.....	11-5
Resource Usage	11-5
CPU Usage (%)	11-6
Memory Usage (MB).....	11-6
Memory Usage (%)	11-6
Response	11-6
Up or Down Status.....	11-6

12 Virtual Directory

Adapter Metrics	12-1
Open Connections.....	12-1
Total No of Connections Processed.....	12-1
Total No of Connections Reused	12-1
Enabled	12-1
Add Operations.....	12-2
Total No of Operation Binds	12-2
Operations Count.....	12-2
Delete Operations.....	12-2
Operational Version.....	12-2
Modify Operations.....	12-2
Total No of Operations Renamed	12-2
Search Operations	12-3
Provisioned version	12-3
Type.....	12-3
Average Time to complete a search request.....	12-3
Maximum Time taken to complete a search request	12-3
Minimum Time taken to complete a search request.....	12-3
Connected IPs	12-3
Current Connections.....	12-4
Total Connections.....	12-4
getUsers	12-4
Current Connections.....	12-4

Total Connections.....	12-4
Resource Usage	12-4
CPU Usage (%)	12-5
CPU Idle Time (%)	12-5
Other CPU Usage (%)	12-5
Heap Usage (MB)	12-5
Memory Usage (MB).....	12-5
Free Memory (MB).....	12-5
Other Memory Usage (MB)	12-5
Total Memory (MB)	12-6
Memory Usage (%)	12-6
Free Memory (%).....	12-6
Other Memory Usage (%)	12-6
Start Time (ms since Epoch)	12-6
Up Time (ms since Epoch)	12-6
Response	12-6
UpDown Status	12-7
Server Latency Metrics	12-7
Total Operations.....	12-7
Average time to complete an LDAP search request	12-7
Maximum Time to complete a search request	12-7
Minimum Time to complete a search request.....	12-8
Number of LDAP Add Requests	12-8
Number of LDAP Binds Requests	12-8
Number of LDAP Delete Requests.....	12-8
Number of LDAP Modify Requests	12-8
Number of LDAP Rename Requests.....	12-8
Number of LDAP Search Requests.....	12-8
Server wide Connection Metrics	12-9
Total No of IPs Currently Connected.....	12-9
Total No of Users Currently Connected	12-9
Total No of Operations.....	12-9
Total No of Open Connections.....	12-9

13 Internet Directory

Oracle Internet Directory	13-1
(Resource Statistics) LDAP Server's Active Database Connections	13-1
Active Database Sessions	13-1
Metric Summary	13-1
Closed LDAP Logon Session Statistics	13-1
Total Closed Logon Sessions	13-2
(Historical Resource Statistics)LDAP Server's Database Usage	13-2
Active Database Sessions	13-2
User Action.....	13-2
Date and Time Stamp	13-2
User Action.....	13-2
Instance Number	13-2

User Action	13-3
Open Database Sessions	13-3
User Action	13-3
Failed Bind Operations	13-3
Metric Summary	13-3
(Critical Events)General System Resource Events	13-3
Critical General Sys Resource Event Occurrences	13-3
Critical General Sys Resource Event Type	13-4
LDAP Load	13-4
Server Load	13-4
Metric Summary	13-4
Historical LDAP Load and Response	13-5
Date and Time Stamp	13-5
Instance Number	13-5
Server Load	13-5
Server Average Response Time (in Milliseconds).....	13-5
(Resource Statistics)LDAP Server Memory Growth	13-5
Average memory growth (%).....	13-5
New LDAP Logon Session Statistics	13-6
Total New Logon Sessions.....	13-6
(Resource Statistics) LDAP Server's Open Database Connections	13-6
Open Database Sessions	13-6
Open LDAP Logon Session Statistics	13-7
Total Open Logon Sessions.....	13-7
Historical LDAP Operation Response Time	13-7
Bind Operation Response Time	13-8
Compare Operation Response Time	13-8
Date and Time Stamp	13-8
Instance Number	13-8
Messaging Search Operation Response Time	13-8
LDAP Operation Response Time	13-8
Bind Operation Response Time	13-8
Compare Operation Response Time	13-9
Messaging Search Operation Response Time	13-9
LDAP Server Resource Usage	13-10
Total CPU Usage (%)	13-10
Total Memory Usage (KB)	13-11
Total Memory Usage (%)	13-11
LDAP Response	13-12
Server Response.....	13-12
Running instances of LDAP Server	13-12
Downtime Count.....	13-12
Host Name	13-12
Instance Number	13-13
Up Since.....	13-13
Non SSL Port Number.....	13-13
SSL Port Number.....	13-13

(Historical Resource Statistics)LDAP Server and System Memory	13-13
Date and Time Stamp	13-13
Instance Number	13-13
LDAP Server's average memory growth (%).....	13-13
System's Total Free Physical Memory (KB).....	13-13
(Critical Events)System Resource Events(3113 Errors).....	13-14
Number of 3113 Error Occurrences	13-14
(Critical Events)System Resource Events(3114 Errors).....	13-14
Number of 3114 Error Occurrences	13-14
(Historical Critical Events)System Resource Events	13-14
Last Event Time	13-14
Number of Event Occurrence.....	13-15
Event Name.....	13-15
Instance Number	13-15
Process ID	13-15
(Critical Events)System Resource Events(Ora Errors).....	13-15
Critical Ora Error Occurrences.....	13-15
Historical LDAP Logon Session Statistics.....	13-16
Date and Time Stamp	13-16
Instance Number	13-16
Total LDAP Logon Sessions	13-16
Total Memory Usage	13-16
Total Mem. Size(in KB).....	13-16
Historical LDAP Operations Profile.....	13-17
Date and Time Stamp	13-17
Instance Number	13-17
Completed Abandon Operations.....	13-17
Completed Add Operations	13-17
Completed Bind Operations	13-17
Completed Compare Operations.....	13-18
Completed Delete Operations	13-18
Completed Modify Operations	13-18
Completed Modrdn Operations	13-18
Completed Search Operations.....	13-18
Completed Unbind Operations.....	13-18
Failed Bind Operations.....	13-18
LDAP Operations Profile.....	13-19
Completed Abandon Operations.....	13-19
Completed Add Operations	13-19
Completed Bind Operations	13-19
Completed Compare Operations.....	13-20
Completed Delete Operations	13-20
Completed Modify Operations	13-21
Completed Modrdn Operations	13-21
Total Operations.....	13-21
Completed Search Operations.....	13-22
Completed Unbind Operations.....	13-22

LDAP Server Total User Sessions	13-22
Total Users Sessions.....	13-22
Directory Integration Profiles	13-23
Last Execution Time.....	13-23
Last Applied Change Number	13-23
Execution Errors	13-23
State	13-24
Execution Status	13-24
Schedule (secs).....	13-24
Type.....	13-24
Directory Integration Server	13-24
Downtime Count.....	13-24
Directory Integration Server Host	13-24
Start Time	13-25
LDAP Server	13-25
Size of Audit Log Purge Queue	13-25
Total Number of Audit Log Objects in Purge Queue	13-25
Size of General Statistics Purge Queue	13-25
Total Number of General Statistics Objects in Purge Queue.....	13-25
Size of Health Statistics Purge Queue	13-26
Total Number of Health Statistics Objects in Purge Queue.....	13-26
Total Number ChangeLogs in Purge Queue	13-27
Total Number of Local Changelogs in Purge Queue.....	13-27
Total Number Remote ChangeLogs in Purge Queue	13-28
Total Number of Remote Changelogs in Purge Queue.....	13-28
Total Number of HIQ ChangeLogs	13-28
HIQ Changelog Count.....	13-28
Replication Server Configuration Set Information	13-29
Number of Threads per Supplier for Change Processing.....	13-29
Number of Threads per Supplier for Transporting Changes	13-30
Running instances of LDAP Replication Server	13-30
Downtime Count.....	13-30
Oracle Directory Server	13-30
Start Time	13-30
Total Number of Local ChangeLogs	13-30
Total Number of Local Changelogs.....	13-31
Total Number of New ChangeLogs	13-31
New Changelog Count.....	13-31
Total number ChangeLogs to be processed	13-32
Total Changelogs to be processed	13-32
Total Number of Retry ChangeLogs	13-33
Retry Changelog Count.....	13-33
OID Replication Server Virtual Memory Size	13-34
Replication Server Memory Size.....	13-34
OID Replication Server Percentage CPU	13-35
Replication Server Percentage CPU.....	13-35
Replication Supplier Details	13-36

Replication Agreement Type.....	13-36
HIQ Changelog Count.....	13-36
New Changelog Count.....	13-37
Retry Changelog Count.....	13-37
Name of the Supplier Replica.....	13-37
Size of Security Refresh events Purge Queue	13-37
Total Number of Security Refresh events Objects in Purge Queue	13-37
Historical LDAP Entry Cache Hit Ratio.....	13-38
User Action.....	13-38
Date and Time Stamp	13-38
Instance Number	13-38
Server Entry Cache Hit Ratio	13-39
Process ID	13-39
LDAP Entry Cache Hit Ratio.....	13-39
Server Entry Cache Hit Ratio	13-39
OID Server Virtual Memory Size.....	13-40
OID Server Memory Size	13-40
Stopped instances of LDAP Server.....	13-40
Config Set Number	13-41
Host Name	13-41
OID Server Percentage CPU Utilization.....	13-41
OID Server Percentage CPU	13-41
(Historical Critical Events)Security Events	13-41
Event Time	13-41
Event Result	13-42
Event Name.....	13-42
Instance Number	13-42
Process ID.....	13-42
User DN.....	13-42
(Critical Event)Super User Failed Logins	13-42
Failed LDAP Super User Login.....	13-42
(Critical Event)Super User Successful Logins	13-43
Successful LDAP Super User Login	13-43
Size of System Resource events Purge Queue.....	13-44
Total Number of System Resource events Objects in Purge Queue	13-44
Size of Tombstone Purge Queue	13-44
Total Number of Tombstone Objects in Purge Queue	13-44
TopProcesses.....	13-45
Process Name.....	13-45
Resource Usage	13-45
CPU Usage (%)	13-45
CPU Idle Time (%)	13-45
Other CPU Usage (%)	13-45
Memory Usage (MB).....	13-45
Free Memory (MB).....	13-45
Other Memory Usage (MB)	13-46
Total Memory (MB)	13-46

Memory Usage (%)	13-46
Free Memory (%).....	13-46
Other Memory Usage (%)	13-46
Start Time (ms since Epoch)	13-46
Response	13-46
Status.....	13-46
Total Number.....	13-46
User LDAP Operations Stats	13-47
Abandon Operations	13-47
Failed Add Operations	13-47
Failed Base Search Operations	13-47
Failed Bind Operations.....	13-47
Failed Compare Operations.....	13-47
Failed Delete Operations.....	13-47
Failed Modify Operations	13-47
Failed ModRdn Operations	13-47
Failed OneLevel Search Operations	13-48
Failed Proxy Bind Operations	13-48
Failed Subtree Search Operations.....	13-48
Successful Add Operations.....	13-48
Successful Base Search Operations	13-48
Successful Bind Operations	13-48
Successful Compare Operations	13-48
Successful Delete Operations	13-48
Successful Modify Operations.....	13-48
Successful ModRdn Operations.....	13-48
Successful OneLevel Search Operations.....	13-48
Successful Proxy Bind Operations.....	13-49
Successful Subtree Search Operations.....	13-49
Unbind Operations	13-49

14 Directory Integration Platform

DIP Aggregated Metrics	14-1
Total Changes Attempted	14-1
Total Successful Changes	14-1
Average LDAP Search Time.....	14-1
Average LDAP Write Time.....	14-2
Average Profile Execution Time	14-2
DIP All Profiles (Consolidated)	14-2
Host	14-2
Last Change Number.....	14-3
Last Execution Error	14-3
Last Execution Time.....	14-3
Last Successful Execution Time	14-3
Status of Last Sync Operation	14-4
DIP Profile Name	14-4
DIP Process.....	14-4

Average Processing Time.....	14-5
Maximum Processing Time	14-5
Minimum Processing Time.....	14-5
Profile Job Status	14-5
DIP Prov profile runtime operational metrics	14-6
Failed Add Group operations	14-6
Failed Add Identity operations	14-6
Failed Add Subscription operations.....	14-6
Failed Add User operations.....	14-6
Failed Delete Group operations	14-7
Failed Delete Identity operations.....	14-7
Failed Delete Subscription operations	14-7
Failed Delete User operations	14-8
Failed Modify Group operations	14-8
Failed Modify Identity operations.....	14-8
Failed Modify User operations.....	14-8
Profile Name	14-9
Successful Add Group operations	14-9
Successful Add Identity operations.....	14-9
Successful Add Subscription operations	14-9
Successful Add User operations	14-10
Successful Delete Group operations.....	14-10
Successful Delete Identity operations	14-10
Successful Delete Subscription operations	14-10
Successful Delete User operations	14-11
Successful Modify Group operations.....	14-11
Successful Modify Identity operations	14-11
Successful Modify User operations	14-12
DIP Sync profile runtime operational metrics	14-12
Skipped Operations	14-12
Failed Add Operations	14-12
Failed Delete Operations.....	14-12
Failed Modify Operations	14-13
Profile Name	14-13
Successful Add Operations.....	14-13
Successful Delete Operations	14-14
Successful Modify Operations.....	14-14
DIP Health	14-14
Host	14-14
MBean Status	14-14
Name	14-15
Quartz Scheduler Status.....	14-15
WLS Server Name	14-15
DIP Provisioning Profiles.....	14-16
DIP Provisioning Profile Name.....	14-16
Host	14-16
Name	14-16

DIP Synchronization Profiles	14-16
DIP Synchronization Profile Name	14-17
Host	14-17
Name	14-17
Resource Usage	14-17
CPU Usage (%)	14-17
Memory Usage (%)	14-18
DIP server status	14-18
Status	14-18

15 WebCenter

Response	15-1
UpDown Status	15-1
WebCenter Spaces (Aggregated)	15-2
Application Name	15-2
Successful Invocations (%)	15-2
Invocations (per minute)	15-3
Page Throughput (per minute)	15-3
Average Page Processing Time (ms)	15-4
Service Name	15-4
WebCenter Content Repositories (Aggregated)	15-5
Application Name	15-5
Successful Invocations (%)	15-5
Invocations (per minute)	15-6
Average Login Time (ms)	15-6
Logins (per minute)	15-7
Average Processing Time (ms)	15-7
Service Name	15-8
WebCenter Events (Aggregated)	15-8
Application Name	15-8
Successful Invocations (%)	15-9
Invocations (per minute)	15-9
Average Processing Time (ms)	15-10
Service Name	15-11
WebCenter External Applications (Aggregated)	15-11
Application Name	15-11
Successful Invocations (%)	15-11
Invocations (per minute)	15-12
Page Throughput (per minute)	15-13
Average Processing Time (ms)	15-13
Service Name	15-13
WebCenter Spaces Pages (Aggregated)	15-14
Application Name	15-14
Successful Invocations (%)	15-14
Invocations (per minute)	15-15
Page Throughput (per minute)	15-15
Average Page Processing Time (ms)	15-15

Service Name	15-16
WebCenter Portlets (Aggregated)	15-16
Application Name	15-16
Successful Invocations (%).....	15-16
Average HTTP 2xx Time (ms).....	15-17
Invocations (per minute).....	15-17
Average Processing Time (ms).....	15-18
Service Name	15-19
WebCenter Producers (Aggregated)	15-19
Application Name	15-19
Successful Invocations (%).....	15-19
Average HTTP 2xx Time (ms).....	15-20
Invocations (per minute).....	15-21
Average Processing Time (ms).....	15-21
Service Name	15-22
WebCenter Search (Aggregated)	15-22
Application Name	15-22
Successful Invocations (%).....	15-22
Invocations (per minute).....	15-23
Average Processing Time (ms).....	15-24
Service Name	15-24
WebCenter Services	15-25
Application Name	15-25
Successful Invocations (%).....	15-25
Invocations (per minute).....	15-26
Average Processing Time (ms).....	15-26
WebCenter BPEL Worklists (Aggregated)	15-27
Application Name	15-27
Successful Invocations (%).....	15-27
Invocations (per minute).....	15-28
Average Processing Time (ms).....	15-29
Service Name	15-29
WebCenter Portlets	15-29
Application Name	15-29
Successful Invocations (%).....	15-30
Average HTTP 2xx Time (ms).....	15-30
Invocations (per minute).....	15-31
Average Processing Time (ms).....	15-31
Producer Type	15-32
WebCenter Portlets (Top 5)	15-32
Application Name	15-32
Successful Invocations (%).....	15-33
Average HTTP 2xx Time (ms).....	15-33
Invocations (per minute).....	15-34
Average Processing Time (ms).....	15-35
Producer Type	15-35
WebCenter Producers	15-36

Application Name	15-36
Successful Invocations (%).....	15-36
Average HTTP 2xx Time (ms).....	15-37
Invocations (per minute).....	15-37
Average Processing Time (ms).....	15-38
Producer Type	15-38
WebCenter Producers (Top 5)	15-39
Application Name	15-39
Successful Invocations (%).....	15-39
Average HTTP 2xx Time (ms).....	15-40
Invocations (per minute).....	15-41
Average Processing Time (ms).....	15-41
Producer Type	15-42
WebCenter Announcements	15-42
Application Name	15-42
Successful Invocations (%).....	15-43
Invocations (per minute).....	15-43
Average Login Time (ms)	15-44
Logins (per minute)	15-44
Average Processing Time (ms).....	15-45
Service Name	15-46
WebCenter Spaces	15-46
Application Name	15-46
Successful Invocations (%).....	15-46
Invocations (per minute).....	15-47
Page Throughput (per minute)	15-47
Average Page Processing Time (ms)	15-48
WebCenter Spaces (Top 5)	15-48
Application Name	15-48
Successful Invocations (%).....	15-49
Invocations (per minute).....	15-49
Page Throughput (per minute)	15-50
Average Page Processing Time (ms)	15-51
WebCenter Content Repositories	15-51
Application Name	15-51
Successful Invocations (%).....	15-52
Average Get Item Time (ms)	15-52
Get Item	15-53
Total Get Item Time (ms)	15-53
Invocations (per minute).....	15-53
Average List Folder Time (ms).....	15-54
List Folder.....	15-54
Total List Folder Time (ms)	15-55
Average Login Time (ms)	15-55
Logins (per minute)	15-56
Average Processing Time (ms).....	15-56
WebCenter Events	15-57

Application Name	15-57
Successful Invocations (%).....	15-57
Invocations (per minute)	15-58
Average Processing Time (ms).....	15-58
WebCenter External Applications	15-59
Application Name	15-59
Invocations (per minute).....	15-59
WebCenter External Applications	15-60
Application Name	15-60
Name.....	15-60
WebCenter Discussion Forum	15-60
Application Name	15-60
Successful Invocations (%).....	15-61
Invocations (per minute).....	15-61
Average Login Time (ms)	15-62
Logins (per minute)	15-62
Average Processing Time (ms).....	15-63
Service Name	15-64
WebCenter Spaces Pages	15-64
Application Name	15-64
Successful Invocations (%).....	15-64
Invocations (per minute).....	15-65
Page Throughput (per minute)	15-65
Average PPR Processing Time (ms)	15-65
Average Page Processing Time (ms)	15-66
WebCenter Spaces Pages (Top 5)	15-66
Application Name	15-66
Successful Invocations (%).....	15-66
Invocations (per minute).....	15-67
Page Throughput (per minute)	15-67
Average Page Processing Time (ms)	15-68
WebCenter Import/Export	15-68
Application Name	15-68
Service Name	15-68
WebCenter Lists	15-68
Application Name	15-69
Successful Invocations (%).....	15-69
Invocations (per minute).....	15-69
Average Processing Time (ms).....	15-70
Service Name	15-71
WebCenter Mail	15-71
Application Name	15-71
Successful Invocations (%).....	15-71
Invocations (per minute).....	15-72
Average Login Time (ms)	15-72
Logins (per minute)	15-73
Average Processing Time (ms).....	15-74

Service Name	15-74
WebCenter Notes	15-75
Application Name	15-75
Successful Invocations (%)	15-75
Invocations (per minute)	15-75
Average Processing Time (ms)	15-76
Service Name	15-77
WebCenter Page Service	15-77
Application Name	15-77
Successful Invocations (%)	15-77
Invocations (per minute)	15-78
Average Processing Time (ms)	15-79
Service Name	15-79
WebCenter People Connections	15-79
Application name	15-80
Successful Invocations (%)	15-80
Invocations (per minute)	15-80
Average Processing Time (ms)	15-81
Service Name	15-81
WebCenter Polls	15-81
Application Name	15-81
Successful Invocations (%)	15-82
Invocations (per minute)	15-82
Average Processing Time (ms)	15-83
Service Name	15-83
WebCenter Recent Activity	15-83
Application Name	15-83
Successful Invocations (%)	15-84
Invocations (per minute)	15-84
Average Processing Time (ms)	15-85
Service Name	15-85
WebCenter RSS News Feeds	15-86
Application Name	15-86
Successful Invocations (%)	15-86
Invocations (per minute)	15-87
Average Processing Time (ms)	15-87
Service Name	15-88
WebCenter Instant Messaging and Presence	15-88
Application Name	15-88
Successful Invocations (%)	15-88
Invocations (per minute)	15-89
Average Login Time (ms)	15-90
Logins (per minute)	15-90
Average Processing Time (ms)	15-91
Service Name	15-91
WebCenter Search	15-92
Application Name	15-92

Successful Invocations (%).....	15-92
Invocations (per minute).....	15-93
Average Processing Time (ms).....	15-93
WebCenter Search (Top 5)	15-94
Application Name.....	15-94
Successful Invocations (%).....	15-95
Invocations (per minute).....	15-95
Average Processing Time (ms).....	15-96
WebCenter BPEL Worklists	15-97
Application Name.....	15-97
Successful Invocations (%).....	15-97
Invocations (per minute).....	15-98
Average Processing Time (ms).....	15-98

16 Web Cache

Performance by Web Cache Instance	16-1
Instance.....	16-1
Request Processing Time (milli seconds).....	16-1
Request Throughput (per second).....	16-1
Resource Usage	16-1
CPU Usage (%).....	16-1
CPU Idle Time (%).....	16-2
Other CPU Usage (%).....	16-2
Memory Usage (MB).....	16-2
Free Memory (MB).....	16-3
Other Memory Usage (MB).....	16-3
Total Memory (MB).....	16-3
Memory Usage (%).....	16-3
Free Memory (%).....	16-4
Other Memory Usage (%).....	16-4
Start Time (ms since Epoch).....	16-4
Up Time (ms since Epoch).....	16-4
Response	16-4
Response Admin.....	16-4
Admin Process Status.....	16-4
Response - Status.....	16-5
Performance of each Site with Summary	16-5
Total Number of Bytes Saved by Compression.....	16-6
Cache Size.....	16-6
Total Number Error Pages Served.....	16-6
Hit Rate (in %).....	16-6
Number of Cached.....	16-6
Currently Open connections.....	16-6
Total Number of Requests Denied.....	16-7
Total Number of Requests Served.....	16-7
WebCache_nontimedHost	16-7
Capacity.value.....	16-7

Load.value	16-7
MaxLoad.value	16-7
Port.value	16-7
WebCache_timedHost	16-7
AvgLatency.value	16-7
Latency.value	16-8
Request.value	16-8
RequestFailure.value	16-8

17 Business Activity Monitoring

Report Server Active Data Page	17-1
Open Connections	17-1
Open Reports	17-1
Open Viewsets	17-1
Activity Message Rate (per sec)	17-2
Average Time To Open Viewsets (in sec)	17-2
Response	17-2
UpDown Status	17-2

18 Business Activity Monitoring Server

ADC Client Requests For All DataObjects	18-1
Client Requests In Progress	18-1
Client Requests Total	18-1
ADC Waiting Threads For All DataObjects	18-2
Deletes Per Second	18-2
Inserts Per Second	18-2
Updates Per Second	18-2
Upserts Per Second	18-2
Deletes Per Second	18-3
Inserts Per Second	18-3
Updates Per Second	18-3
Upserts Per Second	18-4
ADC Waiting Threads For All DataObjects	18-4
Waiting Threads For All DataObjects	18-4
ADC Waiting Threads For All DataObjects	18-4
Waiting Threads For All DataObjects	18-4
ADC Batch Size For All DataObjects	18-4
Current Delete Batch Size	18-5
Current Insert Batch Size	18-5
Current Update Batch Size	18-5
Current Upsert Batch Size	18-5
Current Update Batch Size	18-6
Current Insert Batch Size	18-6
Current Delete Batch Size	18-6
Current Upsert Batch Size	18-6
Current Delete Batch Size	18-7

Current Delete Batch Size	18-7
Current Insert Batch Size.....	18-7
Current Insert Batch Size.....	18-7
Waiting Threads For All DataObjects	18-8
Active Data Cache Transaction Information	18-8
Committed Transactions	18-8
Completed Transactions	18-8
Open Transaction	18-9
Rolledback Transactions	18-9
EMS Metrics	18-9
Total number of received messages.	18-9
Total number of rejected messages.....	18-10
Total delete operations.	18-10
Total insert operations.....	18-10
Total update operations.	18-10
Total upsert operations.....	18-11
Total number of received messages.	18-11
Total number of rejected messages.....	18-11
Delete operations rate.....	18-12
Insert operations rate.....	18-12
Update operations rate.....	18-12
Upsert operations rate.....	18-12
Total received messages.....	18-13
Total rejected messages.....	18-13
Delete operations rate.....	18-13
Insert operations rate.....	18-14
Update operations rate.....	18-14
Upsert operations rate.....	18-14
Total received messages.....	18-14
Total rejected messages.....	18-15
Total delete operations.....	18-15
Total insert operations.....	18-15
Total update operations.....	18-16
Total upsert operations.....	18-16
Event Engine General Information	18-16
Active Rules	18-16
Action Queue Size.....	18-17
Alert History Queue Size	18-17
Rule Fire Queue Size.....	18-17
Event Engine Activity Rule Fire Rate.....	18-17
Events Accepted Rate (per sec)	18-17
Events Fire Rate (per sec).....	18-18
Events Rejected Rate (per sec).....	18-18
Active Data Reports Information	18-18
Active Data Reports	18-18
Active Data Sessions.....	18-19
Outgoing Message Rate (per sec).....	18-19

File Count	18-19
Element Count	18-19
Viewset Average Open Time (in sec)	18-20
Independent Viewsets	18-20
Open Viewsets	18-20
Shared Viewsets	18-21
Sharing Viewsets	18-21
Response	18-21
UpDown Status	18-21

Preface

This manual is a compilation of the Oracle Fusion Middleware target metrics provided in Oracle Enterprise Manager.

Audience

This document is intended for Oracle Enterprise Manager users interested in Oracle Database and database-related target metrics.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/support/contact.html> or visit <http://www.oracle.com/accessibility/support.html> if you are hearing impaired.

Related Documents

For more information, see the following manuals in the Oracle Enterprise Manager 11g Release 1 documentation set:

- *Oracle Enterprise Manager Grid Control Basic Installation Guide*
- *Oracle Enterprise Manager Grid Control Advanced Installation and Configuration Guide*
- *Oracle Enterprise Manager Concepts*
- *Oracle Enterprise Manager Grid Control Quick Start Guide*
- *Oracle Enterprise Manager Administration*
- *Oracle Enterprise Manager Framework, Host, and Services Metric Reference Manual*

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.
<i>italic</i>	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 Fusion Middleware Metric Reference Manual* lists all the Oracle Fusion Middleware target metrics that Enterprise Manager monitors. This manual compiles in one place all the Oracle Fusion Middleware target metric help available online, eliminating the need to have the Grid Control Console up and running.

This preface describes:

- [Structure of the Oracle Enterprise Manager Oracle Fusion Middleware Metric Reference Manual](#)
- [Background Information on Metrics, Thresholds, and Alerts](#)

Structure of the Oracle Enterprise Manager Oracle Fusion Middleware Metric Reference Manual

This manual contains a chapter for each Oracle Fusion Middleware target for which there are metrics.

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

Metric Information

The information for each metric comprises a description, summary of the metric's "vital statistics", data source (if available), and user action. The following list provides greater detail:

- **Description**

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

Explains in table format the target version, collection frequency, upload frequency, operator, default warning threshold, default critical threshold, consecutive number of occurrences preceding notification, and alert text for the metric. Examples follow.
- **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 you will see in the *Oracle Enterprise Manager Oracle Fusion Middleware Metric Reference Manual*.

When default thresholds are not defined for a metric, only the target version and 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	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	10000000	12500000	1	Bytes sent by the server are %value%

The following table shows a metric where the server evaluation frequency is different from the collection frequency.

Target Version	Server Evaluation Frequency	Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
10.1.0.x	Every Minute	Every 5 Minutes	After Every Sample	>	Not Defined	Not Defined	2	Generated By Database Server

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, 9.0.2.x and 10.1.0.x. The x at the end of a version (for example, 9.0.2.x) represents the subsequent patchsets associated with that release.
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.
Server Evaluation Frequency	The rate at which the metric is evaluated to determine whether it has crossed its threshold. For server-generated alerts, the evaluation frequency is determined by Oracle Database internals. For example, if the evaluation frequency is 10 minutes, then when the Average File Write Time degrades to the point an alert should trigger, it could be almost 10 minutes before Enterprise Manager receives indication of the alert. This column is present in the Metric Collection Summary table only for Oracle Database 10g metrics.
Collection Frequency	The rate at which the Management Agent collects data. The collection frequency for a metric comes from the Enterprise Manager default collection file for that target type.

Column Header	Column Definition
Upload Frequency	The rate at which the Management Agent moves data to the Management Repository. For example, upload every n th collection. The upload frequency for a metric comes from the Enterprise Manager default collection file for that target type. This column is present in the Metric Collection Summary table only when the Upload Frequency is different from the Collection Frequency.
Comparison Operator	The comparison method Enterprise Manager uses to evaluate the metric value against the threshold values.
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.
Consecutive Number of Occurrences Preceding Notification	Consecutive number of times a metric's value reaches either the warning threshold or critical threshold before a notification is sent.
Alert Text	Message indicating why the alert was generated. Words that display between percent signs (%) denote variables. For example, Disk Utilization for %keyValue% is %value%% could translate to Disk Utilization for d0 is 80%.

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
Database	Oracle Database
Listener	Oracle Listener

Background Information on 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. For example, for each disk device associated with the Disk Utilization (%) metric, you can define a different warning and critical threshold. Some of the thresholds are predefined by Oracle, others are not.

Once a threshold is reached, an alert is generated. 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.

See Also: See the *Oracle Enterprise Manager Concepts* manual and the Enterprise Manager online help for additional information about metrics, thresholds, and alerts

Editing

Out of the box, Enterprise Manager comes with thresholds for critical metrics. Warning and critical thresholds are used to generate an alert, 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, the key is to choose acceptable values to avoid unnecessary alerts, while still being notified of issues in a timely manner.

You can establish thresholds that will provide pertinent information in a timely manner by defining metric baselines that reflect how your system runs for a normal period of time.

The metrics listed on the Edit Thresholds page are either default metrics provided by Oracle or metrics whose thresholds you can change.

Specifying Multiple Thresholds

The Specifying Multiple Thresholds functionality allows 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. What benefit will be realized by defining a more stringent or lax threshold for that particular device, mount point, and so on?

For example, using the Average Disk I/O Service Time metric, you can define warning and critical thresholds to be applied to all disks (sd0 and sd1), or you can define different warning and critical thresholds for a specific disk (sd0). This allows you to adjust the thresholds for sd0 to be more stringent or lax for that particular disk.

Accessing Metrics Using the Grid Control Console

To access metrics in the Grid Control Console, use the All Metrics page associated with a particular target by doing the following:

1. From the Grid Control Console, choose the target.
2. On the target's home page, click All Metrics in the Related Links section.
3. On the All Metrics page, choose the metric of interest and click **Help**. The help for that metric displays.

Part I

Oracle Fusion Middleware Metrics

This part provides a compilation of the Oracle Fusion Middleware target metrics provided in Oracle Enterprise Manager.

Part I contains the following chapters:

- [Chapter 1, "WebLogic Server"](#)
- [Chapter 2, "Oracle Coherence Metrics"](#)
- [Chapter 3, "HTTP Server"](#)

WebLogic Server

This chapter describes Oracle WebLogic Server metrics.

1.1 Application Overview Metrics

This category provides information about application metrics.

1.1.1 Application - Cached Beans

For the selected application, the total number of EJBs currently in the EJB cache.

1.1.2 Application - Cache Accesses (per minute)

For the selected application, the number of cache access attempts per minute in the last 5 minutes.

1.1.3 Application - Bean Activations (per minute)

For the selected application, this metric shows the total number of EJBs that have been activated since the server started.

1.1.4 Application - Cache Hits (%)

For the selected application, the percentage of cache accesses that completed successfully in the last 5 minutes. The value of this metric can range between zero and 100.

1.1.5 Application - Cache Misses (per minute)

For the selected application, the number of cache misses per minute in the last 5 minutes.

1.1.6 Application - Bean Accesses (per minute)

For the selected application, the EJB pool accesses per minute in the last 5 minutes.

1.1.7 Application - Bean Access Successes (%)

For the selected application, the percentage of pool accesses that were successful in the last 5 minutes.

1.1.8 Application - Free Bean Instances

For the selected application, the current number of available bean instances in the free pool.

1.1.9 Application - Bean Destroys (per minute)

For the selected application, the bean destroys performed per minute in the last 5 minutes.

1.1.10 Application - Bean Access Failures (per minute)

For the selected application, failed EJB pool accesses per minute in the last 5 minutes.

1.1.11 Application - Beans in Use

For the selected application, the number of bean instances currently being used from the free pool.

1.1.12 Application - Bean Transaction Commits (%)

For the selected application, the percentage of transactions that were committed during the last 5 minutes.

1.1.13 Application - Bean Transaction Commits (per minute)

For the selected application, the transaction commits per minute for the last 5 minutes.

1.1.14 Application - Bean Transaction Rollbacks (per minute)

For the selected application, the total number of transactions that have been rolled back for this EJB since the server started.

1.1.15 Application - Bean Transaction Timeouts (per minute)

For the selected application, the transaction timeouts per minute for the last 5 minutes.

1.1.16 Application - MDB Messages (per minute)

For the selected application, the number of messages processed by message-driven beans (MDBs) per minute in the last 5 minutes.

1.1.17 Application - Requests (per minute)

For the selected application, this metric shows the average number of servlet and/or JSP invocations per minute, averaged over the past five minutes.

1.1.18 Application - Request Processing Time (ms)

For the selected application, this metric shows the average amount of time (in milliseconds) spent executing servlets and/or JSPs over the last five minutes.

1.1.19 Application - Active Sessions

This metric shows the number of active sessions for the selected application.

1.1.20 Application - Work Manager Pending Requests

For the selected application, the number of waiting requests in the work manager queue.

1.1.21 Application - Work Manager Requests (per minute)

For the selected application, the number of requests processed per minute in the last 5 minutes.

1.1.22 Application Work Manager - Work Manager Pending Requests

For the selected application, this metric shows the number of work manager requests waiting in the queue.

1.1.23 Application Work Manager - Work Manager Requests (per minute)

For the selected application, this metric shows the number of work manager requests processed per minute, averaged over the past five minutes.

1.1.24 Application Work Manager - Work Manager Stuck Threads

For the selected application, this metric shows the number of work manager threads that are considered stuck on the basis of any stuck thread constraints.

1.2 Application Metrics

This category provides information about application metrics.

1.2.1 Application - Status

This metric shows the current status of the application, indicated as an integer value:

- 1 - the application is considered up and running
- 0 - the application is considered down

On an Enterprise Manager page, the status is shown by an icon that graphically indicates whether the application is up and running or not.

1.3 Datasource Metrics

This category provides information about datasource metrics.

1.3.1 Datasource - Available Connections

For the selected data source, this metric shows the number of database connections currently available (not in use).

1.3.2 Datasource - Connections Created (per minute)

For the selected data source, this metric shows the number of database connections created per minute, averaged over the past five minutes.

1.3.3 Datasource - Connections in Use

For the selected data source, this metric shows the number of JDBC connections currently in use by applications.

1.3.4 Datasource - Connection Leaks (per minute)

For the selected data source, this metric shows the number of JDBC connection leaks per minute, averaged over the past five minutes.

1.3.5 Datasource - Connection Pool Size

For the current data source, this metric shows the current count of JDBC connections in the JDBC connection pool.

1.3.6 Datasource - Connection Refresh Failures (per minute)

For the selected data source, this metric shows the number of failed JDBC connection refreshes per minute, averaged over the past five minutes.

1.3.7 Datasource - Connection Request Failures (per minute)

For the selected data source, this metric shows the number of failed JDBC connection requests per minute, averaged over the past five minutes.

1.3.8 Datasource - Connection Requests (per minute)

For the selected data source, this metric shows the number of requests for a JDBC connection from this data source per minute, averaged over the past five minutes.

1.3.9 Datasource - Successful Connections (%)

For the selected data source, this metric shows the percentage of requests that successfully returned JDBC connections during the last five minutes.

1.3.10 Datasource - Unavailable Connections

For the selected data source, this metric shows the number of database connections that are currently unavailable (in use or being tested by the system).

1.3.11 Datasource - Connection Requests Waiting

For the selected data source, this metric shows the current number of JDBC connection requests waiting for a database connection.

1.3.12 Datasource - Failed Waiting Connection Requests (per minute)

For the selected data source, this metric shows the number of JDBC connection wait failures per minute, averaged over the past five minutes.

A wait failure is a request for a connection from this data source that had to wait before getting a connection and eventually failed to get a connection. Waiting connection requests can fail for a variety of reasons, including waiting for longer than the `ConnectionReserveTimeoutSecond`.

1.3.13 Datasource - Connection Requests that Waited (per minute)

For the selected data source, this metric shows the number of JDBC connection waits per minute, averaged over the past five minutes.

A JDBC connection wait is a request for a connection from this data source that had to wait before getting a connection, including those that eventually got a connection and those that did not get a connection

1.3.14 Datasource - Connection Wait Successes (%)

For the selected data source, this metric shows the percentage of requests waiting for a JDBC connection that successfully got a connection during the last 5 minutes.

A JDBC connection wait is a request for a connection from this data source that had to wait before getting a connection, including those that eventually got a connection and those that did not get a connection

1.3.15 Datasource - Connection Wait Successes (per minute)

For the selected data source, this metric shows the number of JDBC connection wait successes per minute, averaged over the past five minutes.

A wait success is a request for a connection from this data source that had to wait before getting a connection and eventually succeeded in getting a connection.

1.3.16 Datasource - State

For the selected data source, this metric shows the current state of the data source. Possible states are:

- Running - the data source is enabled (deployed and not suspended). This is the normal state of the data source.
- Suspended - the data source has been manually disabled.
- Shutdown - the data source is shutdown and all database connections have been closed.
- Overloaded - all resources in pool are in use.
- Unhealthy - all connections are unavailable (not because they are in use). This state occurs if the database server is unavailable when the data source is created (creation retry must be enabled) or if all connections have failed connection tests (on creation, on reserve, or periodic testing).
- Unknown - the data source state is unknown

1.3.17 Datasource - Statements Added to Cache (per minute)

For the selected data source, this metric shows the number of statements per minute added to the statement cache, averaged over the past five minutes.

1.3.18 Datasource - Statements Discarded from Cache (per minute)

For the current data source, this metric shows the number of statements per minute discarded from the statement cache, averaged over the past five minutes.

1.3.19 Datasource - Cached Statements not Used (per minute)

For the selected data source, this metric shows the number of statements per minute not satisfied by the statement cache, averaged over the past five minutes.

1.3.20 Datasource - Cached Statements Used (%)

For the selected data source, this metric shows the percentage of statements satisfied by the statement cache during the last 5 minutes.

1.3.21 Datasource - Cached Statements Used (per minute)

For the selected data source, this metric shows the number of statements per minute satisfied by the statement cache, averaged over the past five minutes.

1.3.22 Datasource - Statement Cache Size

For the current data source, this metric shows the number of prepared and callable JDBC statements currently cached in the connection pool.

Each JDBC connection in the connection pool has its own cache of statements. This number is the sum of the number of statements in the caches for all connections in the connection pool.

1.4 EJB Module Metrics

This category provides information about EJB module metrics.

1.4.1 EJB Module - Cached Beans

This metric shows the total number of beans currently in the EJB cache.

1.4.2 EJB Module - Cache Accesses (per minute)

For the current EJB module, this metric shows the number times per minute that an attempt was made to access the EJB cache. This value is averaged over the last 5 minutes.

1.4.3 EJB Module - EJB Activations (per minute)

For the current EJB module, this metric shows the total number of beans per minute that have been activated in the last 5 minutes.

1.4.4 EJB Module - EJB Cache Hits (%)

For the current EJB module, this metric shows the percentage of cache accesses that completed successfully in the last 5 minutes.

1.4.5 EJB Module - EJB Cache Misses (per minute)

For the current EJB module, this metric shows the number of cache misses per minute in the last 5 minutes.

1.4.6 EJB Module - EJB Passivations (per minute)

For the current EJB module, this metric shows the total number of beans per minute that have been passivated in the last 5 minutes.

1.4.7 EJB Module - Bean Accesses (per minute)

For the current EJB module, this metric shows how many times the EJB pool has been accessed per minute. This is averaged over the last 5 minutes.

1.4.8 EJB Module - Bean Successes (%)

For the current EJB module, this metric shows the percentage of pool accesses that were successful in the last 5 minutes.

1.4.9 EJB Module - Free Bean Instances

For the selected EJB, this metric shows the current number of available bean instances in the free pool.

1.4.10 EJB Module - Bean Destroys (per minute)

For the selected EJB Module, this metric shows the bean destroys per minute in the last 5 minutes.

1.4.11 EJB Module - Failed Bean Accesses (per minute)

For the selected EJB module, this metric shows the failed pool accesses per minute over the past five minutes.

1.4.12 EJB Module - Beans in Use

For the current EJB module, this metric shows the number of bean instances currently being used from the free pool.

1.4.13 EJB Module - Bean Transaction Commits (per minute)

For the selected EJB module, this metric shows the transaction commits per minute, averaged over the past five minutes.

1.4.14 EJB Module - Bean Transaction Rollbacks (per minute)

For the selected EJB module, this metric shows the number of transaction rollbacks per minute over the past five minutes

1.4.15 EJB Module - Bean Transaction Timeouts (per minute)

For the current EJB module, this metric shows the number of times a transaction timed out per minute, averaged over the last 5 minutes.

1.4.16 EJB Module - MDB Messages (per minute)

For the selected EJB module, this metric shows the number of messages processed by message-driven beans (MDBs) per minute, averaged over the past five minutes.

1.5 EJB Pool Metrics

This category provides information about EJB pool metrics.

1.5.1 EJB Pool - Bean Accesses (per minute)

This metric shows how many times per minute that the selected EJB pool has been accessed over the last 5 minutes.

1.5.2 EJB Pool - Successful Bean Accesses (%)

This metric shows the percentage of attempts to access the pool that were successful in the last 5 minutes.

1.5.3 EJB Pool - Available Beans

This metric shows the current number of available bean instances in the selected EJB pool.

1.5.4 EJB Pool - Bean Destroys (per minute)

For the selected EJB pool, this metric shows the bean destroys per minute over the last 5 minutes.

1.5.5 EJB Pool - Bean Access Failures (per minute)

This metric shows the number of times per minute that an attempted access to the EJB pool has failed, averaged over the past five minutes.

1.5.6 EJB Pool - Beans in Use

This metric shows the number of bean instances currently being used from the free pool.

1.6 EJB Transaction Metrics

This category provides information about EJB transaction metrics.

1.6.1 EJB Transaction - Bean Transaction Commits (%)

This metric shows the percentage of EJB transactions that were committed during the last five minutes.

1.6.2 EJB Transaction - Bean Transaction Commits (per minute)

This metric shows the EJB transaction commits per minute, averaged over the past five minutes.

1.6.3 EJB Transaction - Bean Transaction Rollbacks (per minute)

This metric shows the EJB transaction rollbacks per minute, averaged over the past five minutes.

1.6.4 EJB Transaction - Bean Transaction Timeouts (per minute)

This metric shows the EJB transaction timeouts per minute, averaged over the past five minutes.

1.7 JMS Metrics

This category provides information about JMS metrics.

1.7.1 JMS Connections

For the selected server, this metric shows the number of active JMS connections to the server.

1.7.2 JMS Servers

This metric shows the number of JMS servers that are deployed on the selected WebLogic Server instance.

1.8 JMS Server Metrics

This category provides information about JMS server metrics.

1.8.1 Bytes Pageable

For the current JMS server, this metric shows the number of bytes in all the messages that are currently available to be paged out, but which have not yet been paged out.

The JMS server attempts to keep this number smaller than the "MessageBufferSize" parameter.

1.8.2 Bytes Paged In (per minute)

For the selected JMS server, this metric shows the number of bytes per minute that were read from the paging directory, averaged over the past five minutes.

1.8.3 Bytes Paged Out (per minute)

For the current JMS server, this metric shows the number of bytes per minute that were written to the paging directory, averaged over the past five minutes.

1.8.4 Bytes Pending

This metric shows the current number of bytes pending (unacknowledged or uncommitted) stored on the selected JMS server. Pending bytes are over and above the current number of bytes.

1.8.5 Bytes Received (per minute)

This metric shows the number of bytes received by the selected JMS server in the past five minutes.

1.8.6 Bytes Stored

This metric shows the current number of bytes stored on this JMS server.

1.8.7 Destinations

This metric shows the current number of destinations defined for the selected JMS server.

1.8.8 Messages Paged In (per minute)

This metric shows the current number of messages stored on this JMS server. This number does not include the pending messages.

1.8.9 Messages Paged Out (per minute)

This metric shows the number of messages per minute that were written to the paging directory, averaged over the past five minutes.

1.8.10 Messages Pending

This metric shows the current number of pending messages (unacknowledged or uncommitted) stored on this JMS server. Pending messages are over and above the current number of messages.

1.8.11 Messages Received (per minute)

This metric shows the number of messages received by this JMS server in the past five minutes.

1.8.12 Messages Stored

This metric shows the current number of messages stored on the selected JMS server. This number does not include the pending messages.

1.8.13 Session Pools

This metric shows the current number of session pools instantiated on the selected JMS server.

1.9 JTA Metrics

This category provides information about JTA metrics.

1.9.1 Abandoned Transactions (per minute)

For the selected server, this metric shows the number of transactions abandoned per minute, averaged over the past five minutes.

1.9.2 Committed Transactions (%)

For the selected server, this metric shows the percentage of transactions committed over the last five minutes.

1.9.3 Committed Transactions (per minute)

For the selected server, this metric shows the total number of transactions committed per minute, averaged over the past five minutes.

1.9.4 Committed Transaction Time (seconds)

For the selected server, this metric shows the total number of seconds that JTA transactions were active for all committed transactions.

1.9.5 Heuristic Transactions (per minute)

For the selected server, this metric shows the number of transactions that completed with a heuristic status per minute, averaged over the past five minutes.

1.9.6 Transactions Processed (per minute)

For the selected server, this metric shows the number of transactions processed per minute, averaged over the past five minutes.

1.9.7 Transaction Rollbacks (per minute)

For the selected server, this metric shows the number of transactions rolled back per minute, averaged over the past five minutes.

1.9.8 Transaction Rollbacks due to Application Error (per minute)

For the selected server, this metric shows the number of transactions rolled back due to an application error per minute, averaged over the past five minutes.

1.9.9 Transaction Rollbacks due to Resource Error (per minute)

For the selected server, this metric shows the number of transactions rolled back due to a resource error per minute, averaged over the past five minutes.

1.9.10 Transaction Rollbacks due to System Error (per minute)

For the selected server, this metric shows the number of transactions rolled back due to an internal system error per minute, averaged over the past five minutes.

1.9.11 Transaction Rollbacks due to Timeout (per minute)

For the selected server, this metric shows the number of transactions rolled back due to a timeout expiration per minute, averaged over the past five minutes.

1.9.12 Active Transactions

This metric shows the number of currently active JTA transactions on the selected server.

1.10 JVM Metrics

This category provides information about JVM metrics.

1.10.1 JVM Class Loading

This metric provides information about JVM class loading metrics.

1.10.2 Classes Currently Loaded

This metric shows the number of Java classes currently loaded into the selected Java Virtual Machine (JVM).

1.10.3 Classes Loaded (since startup)

This metric shows the total number of Java classes that have been loaded into the Java Virtual Machine (JVM) since the JVM was started.

Compare this metric with the number of classes that have been unloaded from the JVM since it was started.

1.10.4 Classes Unloaded (since startup)

This metric shows the total number of Java classes that have been unloaded from the Java Virtual Machine (JVM) since the JVM was started.

Compare this metric with the number of classes that have been loaded into the JVM since it was started.

1.11 JVM Compilation Metrics

This category provides information about JVM compilation metrics.

1.11.1 Compilation Time (since startup) (ms)

For the selected Java Virtual Machine (JVM), this metric shows the total time spent by the JVM in compilation. The value of the metric is in milliseconds and shows the time spent in compilation since the JVM was last started.

1.11.2 CPU Usage (%)

This metric shows the percentage of the CPU currently in use by the Java Virtual Machine (JVM). This includes the load that the JVM is placing on all processors in the host computer.

For example, if the host uses multiple processors, the value represents a snapshot of the average load on all the processors.

1.11.3 Active Daemon Threads

This metric shows the number of daemon Java threads currently running in the Java Virtual Machine (JVM) across all processors.

1.11.4 Heap Usage (MB)

This metric shows the current size of the Java Virtual Machine (JVM) memory heap, in megabytes.

1.11.5 JDK Vendor

This metric shows the name vendor responsible for the current Java Development Kit (JDK) that is being used for the current Java Virtual Machine.

For example, when you install and configure an Oracle WebLogic Server domain, you can choose to use the standard Sun Microsystems JDK or the Oracle WebLogic Server JRockit JDK.

1.11.6 JDK Version

This metric shows the version of the Java Development Kit (JDK) that is currently in use by the selected server.

1.12 JVM Garbage Collectors

This category provides information about JVM garbage collectors metrics.

1.12.1 Garbage Collector - Garbage Collection Invocations (since startup)

This metric shows the number of times the Java Virtual Machine (JVM) garbage collector was invoked since the JVM was started.

1.12.2 Garbage Collector - Execution Time (since startup) (ms)

This metric shows the total time spent by the Java Virtual Machine (JVM) in garbage collection since the JVM was started.

1.13 JVM Memory Pools

This category provides information about JVM memory pools metrics.

1.13.1 Memory Pool - Current Usage (KB)

This metric shows the current memory usage (in kilobytes) of the selected Java Virtual Machine (JVM).

1.13.2 Memory Pool - Peak Usage (KB)

This metric shows the maximum (peak) memory used by the Java Virtual Machine (JVM) in kilobytes, since the JVM was started.

1.14 JVM Memory Usage

This category provides information about JVM memory usage metrics.

1.14.1 Heap Memory Usage (KB)

This metric shows the total heap memory (in kilobytes) currently in use by the selected Java Virtual Machine (JVM).

1.14.2 Non-Heap Memory Usage (KB)

This metric shows the total non-heap memory (in kilobytes) currently in use by the selected Java Virtual Machine (JVM).

1.14.3 Objects Pending Finalization

This metric shows the number of objects pending finalization in the selected Java Virtual Machine (JVM).

1.15 Nursery Size (MB)

This metric shows the amount of memory that is currently allocated to the nursery, in megabytes.

The nursery is the area of the Java heap that the virtual machine allocates to most objects. Instead of garbage collecting the entire heap, generational garbage collectors focus on the nursery.

Most objects do not last a long time in memory; as a result, in most cases, it is sufficient to perform garbage collect on only the nursery and not on the entire memory heap.

If you are not using a generational garbage collector, the nursery size is zero.

1.16 JVM Runtime Metrics

This metric shows how long the selected Java Virtual Machine (JVM) has been up in running (in milliseconds).

1.16.1 VM Name

This metric shows the name of the selected Java Virtual Machine (JVM).

1.16.2 VM Version

This metric shows the version number of the selected Java Virtual Machine (JVM).

1.17 JVM ThreadContention

This category provides information about JVM thread contention metrics.

1.17.1 Blocked Count

This metric shows the number of times the selected Java Virtual Machine (JVM) thread has been blocked since the thread was started.

1.17.2 Blocked Time (since startup) (seconds)

This metric shows how long (in seconds) this thread has been blocked since it was started or since thread contention monitoring was enabled.

1.17.3 CPU Usage (since startup) (seconds)

This metric shows the amount of CPU (in seconds) that is currently being used by the selected Java Virtual Machine (JVM) thread.

1.17.4 Lock Name

This metric shows the name of the lock the selected Java Virtual Machine (JVM) thread is blocked on (if applicable).

1.17.5 Thread State

This metric shows the state of the selected Java Virtual Machine (JVM) thread.

The state can be any of the states defined here on the Sun Java Web site.

1.17.6 Wait Count

This metric shows the number of times the selected Java Virtual Machine (JVM) thread has entered wait states since the thread was started.

1.17.7 Wait Time (since startup) (seconds)

This metric shows how long (in seconds) the selected Java Virtual Machine (JVM) thread has been in wait states since the thread was started.

1.18 Active Threads

This metric shows the number of Java threads (daemon and non-daemon) that are currently running in the virtual machine across all processors.

1.19 JVM Threads

This category provides information about JVM threads metrics.

1.19.1 Active Daemon Threads

This metric shows the current number of active daemon threads in the selected Java Virtual Machine (JVM).

1.19.2 Deadlocked Threads

This metric shows the current number of deadlocked threads in the selected Java Virtual Machine (JVM).

1.19.3 Peak Threads

This metric shows the maximum (peak) number of threads started in this Java Virtual Machine (JVM) since the JVM was started.

1.19.4 Active Threads

This metric shows the current number of active threads in the selected Java Virtual Machine (JVM).

1.19.5 Threads Started (since startup)

This metric shows the total number of threads started in this Java Virtual Machine (JVM) since the JVM was started.

1.20 Response

This category provides information about response metrics.

1.20.1 Status

This metric shows the status of the selected server. The value is an integer, where "1" means that the server is up and running and "0" means the server is down and unavailable.

1.20.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every 60 Samples	=	Not Defined	0	1	The J2EE Server instance is down

1.21 Server Metrics

This category provides information about server metrics.

1.21.1 Listen Port

This metric returns the port number on which the WebLogic server is listening for requests.

1.21.2 Active Sockets

This metric shows the current number of sockets listening on this server.

This value is equal to the number of sockets registered with the tuning muxer. For more information, see "Tuning Muxers" in *Oracle Fusion Middleware Performance and Tuning for Oracle WebLogic Server*.

1.21.3 Start Time

This metric shows the time when the server was started.

1.21.4 State

This metric shows the current state of the server as an integer. A value of "1" indicates that the server is up; a value of "2" indicates that the server is down.

This metric is used to determine the value of the Server Status metric, which is shown on the Server home page.

1.22 Server Datasource Metrics

This category provides information about server datasource metrics.

1.22.1 Server Datasource - Available Connections

The number of database connections currently available (not in use) in this data source.

1.22.2 Server Datasource - Connections Created (per minute)

For the selected server, the number of database connections created per minute in the last 5 minutes.

1.22.3 Server Datasource - Connections in Use

For the selected server, the number of JDBC connections currently in use by applications on the server.

1.22.4 Server Datasource - Connection Pool Size

For the current server, the current count of JDBC connections in the connection pool in the data source.

1.22.5 Server Datasource - Connection Requests (per minute)

The number of connection requests per minute for a connection from this data source in the last 5 minutes.

1.22.6 Server Datasource - Successful Connections (%)

The percentage of connection requests that successfully returned connections during the last 5 minutes.

1.22.7 Server Datasource - Connection Requests Waiting

The current number of connection requests waiting for a database connection.

1.22.8 Server Datasource - Connection Requests that Waited (per minute)

The number of connection waits per minute in the last 5 minutes.

1.22.9 Server Datasource - Connection Wait Successes (%)

The percentage of requests waiting for a connection that successfully got a connection during the last 5 minutes.

1.22.10 Server Datasource - Connection Wait Successes (per minute)

The number of connection wait successes per minute in the last 5 minutes.

1.22.11 Server Datasource - Statements Added to Cache (per minute)

The number of statements per minute added to the statement cache in the last 5 minutes.

1.22.12 Server Datasource - Cached Statements not Used (per minute)

The number of statements per minute not satisfied by the statement cache in the last 5 minutes.

1.22.13 Server Datasource - Cached Statements Used (%)

The percentage of statements satisfied by the statement cache during the last 5 minutes.

1.22.14 Server Datasource - Cached Statements Used (per minute)

The number of statements per minute satisfied by the statement cache in the last 5 minutes.

1.22.15 Server Datasource - Statement Cache Size

The number of prepared and callable statements currently cached in the statement.

Each connection in the connection pool has its own cache of statements. This number is the sum of the number of statements in the caches for all connections in the connection pool.

1.23 Server Overview Metrics

This category provides information about server overview metrics.

1.23.1 JDBC Connection Creates (per minute)

For the selected server, the number of database connections created per minute in the last 5 minutes.

1.23.2 Open JDBC Connections

For the selected server, the number of JDBC connections currently open.

1.23.3 Cached Beans

The total number of EJBs currently in the EJB cache.

1.23.4 Cache Accesses (per minute)

The number of EJB cache access attempts per minute in the last 5 minutes on the selected server.

1.23.5 Bean Activations (per minute)

The total number of Java beans per minute that have been activated in the last 5 minutes.

1.23.6 Cache Hits (%)

The percentage of EJB cache accesses that completed successfully in the last 5 minutes.

1.23.7 Cache Misses (per minute)

The number of EJB cache misses per minute in the last 5 minutes.

1.23.8 Bean Accesses (per minute)

EJB pool accesses per minute over the last 5 minutes.

1.23.9 Bean Successes (%)

For the selected server, this metric displays the percentage of EJB pool accesses that were successful in the last 5 minutes.

1.23.10 Free Bean Instances

For the selected server, this metric indicates the current number of available Java bean instances in the free pool.

1.23.11 Bean Destroys (per minute)

For the selected server, this metric shows the bean destroys per minute in the last 5 minutes.

1.23.12 Bean Access Failures (per minute)

For the selected server, this metric shows the percentage of pool accesses that were successful in the last 5 minutes.

1.23.13 Beans in Use

The number of bean instances currently being used from the EJB free pool.

1.23.14 Bean Transaction Commits (%)

For the selected server, this metric shows the percentage of EJB transactions that were committed during the last 5 minutes.

1.23.15 Bean Transaction Commits (per minute)

For the selected server, this metric identifies the EJB transaction commits per minute for the last 5 minutes.

1.23.16 Bean Transaction Rollbacks (per minute)

For the selected server, this metric specifies the percentage of transactions that were committed during the last 5 minutes.

1.23.17 Bean Transaction Timeouts (per minute)

This metric shows the EJB transaction timeouts per minute for the last 5 minutes on the selected server.

1.23.18 Committed JTA Transactions (per minute)

The total number of JTA transactions committed per minute on this server in the last 5 minutes.

1.23.19 JTA Transaction Rollbacks (per minute)

The number of JTA transactions rolled back per minute in the last 5 minutes on the selected server.

1.23.20 Active JTA Transactions

The number of active JTA transactions on the server.

1.23.21 MDB Messages (per minute)

For the selected server, this metric specifies the number of messages processed by message-driven beans (MDBs) per minute in the last 5 minutes.

1.23.22 Pending Messages

The current number of messages pending (unacknowledged or uncommitted) stored on the JMS servers deployed on this WebLogic Server instance. Pending messages are over and above the current number of messages.

1.23.23 Current Messages

The current number of JMS messages stored on the JMS servers deployed on this WebLogic Server instance. This number does not include the pending messages.

1.23.24 JMS Servers

The number of JMS servers that are deployed on this WebLogic Server instance.

1.23.25 Requests (per minute)

For the selected server, the average number of servlet and/or JSP invocations per minute in the last 5 minutes.

1.23.26 Request Processing Time (ms)

For the selected server, the average amount of time spent executing servlets and/or JSPs in the last 5 minutes (milliseconds)

1.23.27 Active Sessions

The number of active sessions for this server.

1.23.28 Work Manager Pending Requests

This metric shows the number of work manager requests waiting in the queue.

1.23.29 Work Manager Requests (per minute)

This metric shows the number of work manager requests processed per minute in the last 5 minutes on the selected server.

1.24 Server Work Manager Metrics

This category provides information about server work manager metrics.

1.24.1 Server Work Manager - Work Manager Pending Requests

For the selected server, this metric shows the number of work manager requests pending in the queue.

1.24.2 Server Work Manager - Work Manager Requests (per minute)

For the selected server, this metric shows the number of work manager requests processed per minute, averaged over the past five minutes.

1.24.3 Server Work Manager - Work Manager Stuck Threads

For the selected server, this metric shows the number of work manager threads that are considered stuck on the basis of any stuck thread constraints.

1.25 Servlet/JSP Metrics

This category provides information about server/JSP metrics.

1.25.1 Servlet/JSP - Reloads (per minute)

This metric shows the average number of reloads per minute of the selected servlet or JSP in the last 5 minutes.

1.25.2 Servlet/JSP - Requests (per minute)

This metric shows the average number of invocations per minute of the selected servlet or JSP in the last 5 minutes.

1.25.3 Servlet/JSP - Request Processing Time (ms)

This metric shows the average number of invocations per minute of the selected servlet or JSPs in the last 5 minutes.

1.26 Servlet Rollup

This category provides information about servlet rollup metrics.

1.26.1 Servlet - Active Invocations

This metric provides details about the servlet invocations that are active and are currently in the web container. Active invocation refers to the number of requests that are currently being processed by the server.

1.26.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

1.26.2 Servlet - Current Client Processing Time

This metric indicates the time taken to execute the servlet during the last interval.

1.26.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 15 Minutes	After Every Sample	>	Not Defined	Not Defined	1	The Current Client Processing time for servlet %Name% is %value%

1.26.2.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Name", "Application Name", and "Web Application Name" objects.

If warning or critical threshold values are currently set for any unique combination of "Name", "Application Name", and "Web Application 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 "Name", "Application Name", and "Web Application Name" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

1.26.3 Servlet - Average Execution Time

This metric indicates the average time taken by a servlet to respond.

1.26.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

1.26.4 Servlet - Total Execution Time

This metric indicates the total execution time for a request.

1.26.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

1.26.5 Servlet - Invocations Per Second

This metric indicates the number of number of requests to the server per second.

1.26.5.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 15 Minutes	After Every Sample	>	Not Defined	Not Defined	1	The invocations per second for Servlet %Name% is %value%

1.26.5.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Name", "Application Name", and "Web Application Name" objects.

If warning or critical threshold values are currently set for any unique combination of "Name", "Application Name", and "Web Application 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 "Name", "Application Name", and "Web Application Name" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

1.26.6 Servlet - Invocation Total Count

This metric indicates the total number of invocations for this servlet.

1.26.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

1.27 Web Module Metrics

This category provides information about Web module metrics.

1.27.1 Web Module - Requests (per minute)

For the selected Web module, this metric shows the average number of invocations per minute of servlets and/or JSPs in the last 5 minutes.

1.27.2 Web Module - Request Processing Time (ms)

For the selected Web module, this metric shows the average amount of time spent (in milliseconds) executing servlets and/or JSPs in the last 5 minutes.

1.27.3 Web Module - Active Sessions

This metric shows the number of active sessions for the selected web module.

1.28 Java JEE Web Service Endpoints

This category provides information about Java JEE service endpoints metrics.

1.28.1 Dispatch Time Total

This metric shows the total amount of dispatch time since the Web service was started.

1.28.2 Execution Time Total

This metrics represents the total time that the Web service port has spent executing since the Web service was started.

1.28.3 Invocation Count

This metric represents the total number of invocations for the selected Web services port since the Web service was started.

1.28.4 Response Count

This metric shows the total number of responses since the Web service was started.

1.28.5 Response Error Count

This number provides the total number of response errors since the Web service was started.

1.28.6 Response Time Total

This metric represents the total response time of this Web services port since the Web service was started.

1.29 Work Manager Metrics

This category provides information about work manager metrics.

1.29.1 Work Manager Pending Requests

This metric shows the number of work manager requests processed per minute, averaged over the past five minutes.

1.29.2 Work Manager Requests (per minute)

This metric shows the number of requests processed by the work manager per minute, averaged over the past five minutes.

1.29.3 Work Manager Stuck Threads

This metric shows the number of work manager threads that are considered stuck on the basis of any stuck thread constraints.

Oracle Coherence Metrics

The oracle_coherence metrics provide description, collection statistics, data source, multiple thresholds (where applicable), and user action information for each metric.

2.1 Cache Performance - Aggregated Over Nodes

Aggregated cache performance across all nodes on which a cache is running

2.1.1 Cache Hits Delta Sum

The rough number of cache hits during the last collection interval. A cache hit is a read operation invocation (i.e. get()) for which an entry exists in this map.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–1 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.1.2 Cache Hits Millis Delta Sum

The total number of milliseconds (during the last collection interval) for the get() operations for which an entry existed in this map.

Metric Summary

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

Table 2–2 Metric Summary Table

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.1.3 Cache Misses Delta Sum

The rough number of cache misses during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–3 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.1.4 CacheMissesMillisDeltaSum

The total number of milliseconds (during the last collection interval) for the get() operations for which no entry existed in this map.

Metric Summary

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

Table 2–4 Metric Summary Table

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.1.5 StoreReadMillisDeltaSum

The cumulative time (in millis) spent on load operations during the last collection interval; -1 if persistence type is NONE.

Metric Summary

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

Table 2–5 Metric Summary Table

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.1.6 StoreReadsDeltaSum

The total number of load operations during the last collection interval; -1 if persistence type is NONE.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–6 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.1.7 StoreWriteMillisDeltaSum

The cumulative time (in milliseconds) spent on store and erase operations during the last collection interval; -1 if persistence type is NONE or READ-ONLY.

Metric Summary

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

Table 2–7 Metric Summary Table

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.1.8 StoreWritesDeltaSum

The total number of store and erase operations during the last collection interval; -1 if persistence type is NONE or READ-ONLY.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–8 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.1.9 Total Gets Delta Sum

The total number of get() operations during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–9 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.1.10 TotalGetsMillisDeltaSum

The total number of milliseconds spent on get() operations during the last collection interval.

Metric Summary

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

Table 2–10 Metric Summary Table

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.1.11 TotalPutsDeltaSum

The total number of put() operations during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–11 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.1.12 TotalPutsMillisDeltaSum

The total number of milliseconds spent on put() operations during the last collection interval.

Metric Summary

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

Table 2–12 Metric Summary Table

Target Version	Evaluation and Collection Frequency
All Versions	Every 300 Hours

2.2 Cache Configuration

This metric addresses the cache configuration

2.2.1 Batch Factor

The BatchFactor attribute is used to calculate the `soft-ripe` time for write-behind queue entries. A queue entry is considered to be `ripe` for a write operation if it has been in the write-behind queue for no less than the QueueDelay interval. The `soft-ripe` time is the point in time prior to the actual `ripe` time after which an entry will be included in a batched asynchronous write operation to the CacheStore (along with all other `ripe` and `soft-ripe` entries). This attribute is only applicable if asynchronous writes are enabled (i.e. the value of the QueueDelay attribute is greater than zero) and the CacheStore implements the storeAll() method. The value of the element is expressed as a percentage of the QueueDelay interval. Valid values are doubles in the interval [0.0, 1.0].

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.2.2 Description

The cache description.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.2.3 Expiry Delay

The time-to-live for cache entries in milliseconds. Value of zero indicates that the automatic expiry is disabled. Change of this attribute will not affect already-scheduled expiry of existing entries.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.2.4 Flush Delay

The number of milliseconds between cache flushes. Value of zero indicates that the cache will never flush.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.2.5 High Units

The limit of the cache size measured in units. The cache will prune itself automatically once it reaches its maximum unit level. This is often referred to as the `high water mark` of the cache.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.2.6 Low Units

The number of units to which the cache will shrink when it prunes. This is often referred to as a `low water mark` of the cache.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.2.7 Persistence Type

The persistence type for this cache. Possible values include: NONE, READ-ONLY, WRITE-THROUGH, WRITE-BEHIND.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.2.8 Queue Delay

The number of seconds that an entry added to a write-behind queue will sit in the queue before being stored via a CacheStore. Applicable only for WRITE-BEHIND persistence type.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.2.9 Refresh Factor

The RefreshFactor attribute is used to calculate the `soft-expiration` time for cache entries. Soft-expiration is the point in time prior to the actual expiration after which any access request for an entry will schedule an asynchronous load request for the entry.

This attribute is only applicable for a ReadWriteBackingMap which has an internal LocalCache with scheduled automatic expiration. The value of this element is expressed as a percentage of the internal LocalCache expiration interval. Valid values are doubles in the interval[0.0, 1.0]. If zero, refresh-ahead scheduling will be disabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.2.10 Refresh Time

The timestamp when this model was last retrieved from a corresponding node. For local servers it is the local time.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.2.11 Requeue Threshold

The maximum size of the write-behind queue for which failed CacheStore write operations are requeued. If zero, the write-behind requeueing will be disabled. Applicable only for WRITE-BEHIND persistence type.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.3 Cache Performance

The cache performance metrics

2.3.1 Average Gets Millis

The average number of milliseconds per get() invocation since the cache statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.2 Average Get Millis Delta

The average number of milliseconds per get() invocation during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.3 Average Hit Millis

The average number of milliseconds per get() invocation that is a hit.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.4 Average Hit Millis Delta

The average number of milliseconds per get() invocation that is a hit during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.5 Average Miss Millis

The average number of milliseconds per get() invocation that is a miss.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.6 Average Miss Millis Delta

The average number of milliseconds per get() invocation that is a miss during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.7 Average Put Millis

The average number of milliseconds per put() invocation since the cache statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.8 Average Put Millis Delta

The average number of milliseconds per put() invocation since the cache statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.9 Cache Hits

The rough number of cache hits since the cache statistics were last reset. A cache hit is a read operation invocation (i.e. get()) for which an entry exists in this map.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.10 Cache Hits Delta

The rough number of cache hits during the last collection interval. A cache hit is a read operation invocation (i.e. get()) for which an entry exists in this map.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–13 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.3.11 Cache Hits Millis

The total number of milliseconds (since the last time statistics were reset) for the get() operations for which an entry existed in this map.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.12 Cache Hits Millis Delta

The total number of milliseconds (during the last collection interval) for the get() operations for which an entry existed in this map.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.13 Cache Misses

The rough number of cache misses since the cache statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.14 Cache Misses Delta

The rough number of cache misses during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–14 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.3.15 Cache Misses Millis

The total number of milliseconds (since the last time statistics were reset) for the get() operations for which no entry existed in this map.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.16 Cache Misses Millis Delta

The total number of milliseconds (during the last collection interval) for the get() operations for which no entry existed in this map.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.17 Hit Probability

The rough probability ($0 \leq p \leq 1$) that the next invocation will be a hit, based on the statistics collected since the last reset of the cache statistics.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.18 Queue Size

The size of the write-behind queue size. Applicable only for WRITE-BEHIND persistence type.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.19 Size

The number of entries in the cache.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.20 Store Average Batch Size

The average number of entries stored per CacheStore write operation. A call to the store() method is counted as a batch of one, whereas a call to the storeAll() method is counted as a batch of the passed Map size. The value of this attribute is -1 if the persistence type is NONE.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.21 Store Average Read Millis

The average time (in millis) spent per write operation; -1 if persistence type is NONE.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.22 Store Failures

The total number of CacheStore failures (load, store and erase operations); -1 if persistence type is NONE.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.23 Store Failures Delta

The total number of CacheStore failures (load, store and erase operations) during the last collection interval; -1 if persistence type is NONE.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.24 StoreRead Millis

The cumulative number of load operations per node; -1 if persistence type is NONE

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.25 Store Read Millis Delta

The cumulative time (in millis) spent on load operations during the last collection interval; -1 if persistence type is NONE.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.26 Store Reads

The total number of load operations; -1 if persistence type is NONE.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.27 Store Reads Delta

The total number of load operations during the last collection interval; -1 if persistence type is NONE.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–15 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.3.28 Store Write Millis

The cumulative time (in milliseconds) spent on store and erase operations; -1 if persistence type is NONE or READ-ONLY.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.29 Store Write Millis Delta

The cumulative time (in milliseconds) spent on store and erase operations during the last collection interval; -1 if persistence type is NONE or READ-ONLY.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.30 Store Writes

The total number of store and erase operations; -1 if persistence type is NONE or READ-ONLY.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.31 Store Writes Delta

The total number of store and erase operations during the last collection interval; -1 if persistence type is NONE or READ-ONLY.

Metric Summary

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

Table 2–16 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.3.32 Total Gets

The total number of get() operations since the cache statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.33 Total Gets Delta

The total number of get() operations during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding

Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–17 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.3.34 Total Gets Millis

The total number of milliseconds spent on get() operations since the cache statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.35 Total Gets Millis Delta

The total number of milliseconds spent on get() operations during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.36 Total Puts

The total number of put() operations since the cache statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.37 Total Puts Delta

The total number of put() operations during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–18 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "Service" 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 "Name" and "Service" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.3.38 Total Puts Millis

The total number of milliseconds spent on put() operations since the cache statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.39 Total Puts Millis Delta

The total number of milliseconds spent on put() operations during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.3.40 Units

The size of the cache measured in units.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.4 Cluster

This cluster metrics are described.

2.4.1 Cluster Name

The name of the cluster.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.4.2 Cluster Size

The total number of cluster nodes.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.4.3 License Mode

The license mode that this cluster is using. Possible values are Evaluation, Development or Production.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.4.4 Local Member Id

The member id for the cluster member that is co-located with the reporting MBeanServer; -1 if the cluster service is not running.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.4.5 Members

An array of all existing cluster members.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.4.6 Oldest Member ID

The senior cluster member id; -1 if the cluster service is not running.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.4.7 Running

Specifies whether the cluster is running.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.5 Cluster Aggregated

The aggregated coherence cluster metrics are described

2.5.1 Cache Size

The number of entries in the cache.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.5.2 Cache Size Delta

The number of entries present in the cache during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–19 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	<	Not Defined	Not Defined	1	Cache deleted or new cache added

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Cluster 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 "Cluster Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.5.3 Cluster Size

The total number of cluster nodes.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.5.4 Cluster Size Delta

The total number of cluster nodes present during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–20 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	<	Not Defined	Not Defined	1	Cache deleted or new cache added

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Cluster 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 "Cluster Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.5.5 Publisher Success Rate

The publisher success rate from the viewing node to the viewed node since the statistics were last reset.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–21 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	<	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Cluster 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 "Cluster Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.5.6 Receiver Success Rate

The receiver success rate from the viewing node to the viewed node since the statistics were last reset.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–22 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	<	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Cluster 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 "Cluster Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.6 Cluster Alerts

The coherence cluster alert metrics are described.

2.6.1 Cluster Name

The name of the cluster.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.6.2 Cluster Nodes

The total number of cluster nodes.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.6.3 Change in No. of Nodes Since Last Collection

The change in the number of nodes since the last collection.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–23 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 60 Seconds	After Every Sample	<	0	-5	1	Nodes departed the Cluster

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Cluster 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 "Cluster Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.6.4 Change in Number of Nodes

The change in the number of nodes since the last time statistics were collected.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–24 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 60 Seconds	After Every Sample	>	0	Not Defined	1	Nodes added to Cluster

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Cluster 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 "Cluster Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.7 Connection

The coherence connection metric s are described.

2.7.1 Connection Time Millis

The time duration (in milliseconds) that the client has been connected.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.7.2 Outgoing Byte Backlog

The backlog (in bytes) of the outgoing queue

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.7.3 Outgoing Message Backlog

The backlog of the outgoing message queue.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.7.4 Refresh Time

The timestamp when this model was last retrieved from a corresponding node. For local servers it is the local time.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.7.5 Remote Address

The IP address of the corresponding client.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.7.6 Remote Port

The port of the corresponding client.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.7.7 Timestamp

The date/time value (in local time) that the corresponding client connected to the Proxy.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.7.8 Total Bytes Received

The total number of bytes recieved since the last time the statistics were reset.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–25 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Name", "NodeID", and "UUID" objects.

If warning or critical threshold values are currently set for any unique combination of "Name", "NodeID", and "UUID" 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 "Name", "NodeID", and "UUID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.7.9 Total Bytes Sent

The total number of bytes sent since the statistics were last reset.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–26 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Name", "NodeID", and "UUID" objects.

If warning or critical threshold values are currently set for any unique combination of "Name", "NodeID", and "UUID" 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 "Name", "NodeID", and "UUID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.7.10 Total Messages Received

The total number of messages received since the statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.7.11 Total Messages Sent

The total number of messages sent since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.8 Connection Manager

The connection manager metrics are described.

2.8.1 Connection Count

The number of client connections.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.8.2 Host IP

The IP address of the host.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.8.3 Incoming Buffer Pool Capacity

The pool capacity (in bytes) of the incoming buffer.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.8.4 Incoming Buffer Pool Size

The number of buffers in the incoming pool.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.8.5 Outgoing Buffer Pool Capacity

The pool capacity (in bytes) of the outgoing buffer.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.8.6 Outgoing Buffer Pool Size

The number of buffers in the outgoing pool.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.8.7 Outgoing Byte Backlog

The backlog (in bytes) of the outgoing queue.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.8.8 Outgoing Message Backlog

The backlog of the outgoing message queue.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.8.9 RefreshTime

The timestamp when this model was last retrieved from a corresponding node. For local servers it is the local time.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.8.10 Total Bytes Received

The total number of bytes received by the Proxy host since the statistics were last reset.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–27 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.8.11 Total Bytes Sent

The total number of bytes sent by the Proxy host since the statistics were last reset.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–28 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.8.12 Total Messages Received

The total number of messages received by the Proxy host since the statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.8.13 Total Messages Sent

The total number of messages sent by the Proxy host since the statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.9 Garbage Collector after GC

Garbage Collector memory pool metrics after completion of garbage collection per node.

2.9.1 Name

Garbage Collector Name.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.9.2 Node ID

Coherence Node ID.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.9.3 GC Thread Count

GC Thread Count per garbage collector .

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.9.4 Start Time (ms)

GC Start Time for a garbage collector.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.9.5 End Time (ms)

GC End Time for a garbage collector.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.9.6 Key of Memory Usage after GC

Garbage Collector key.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.9.7 Committed of Memory Usage after GC

The amount of memory that is guaranteed to be available for use by the JVM after GC.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.9.8 Init of Memory Usage after GC

The initial amount of memory that the JVM requested for a memory pool from the operating system during startup.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.9.9 Max of Memory Usage after GC

The maximum amount of memory that can be used by the JVM.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.9.10 Used of Memory Usage after GC

The amount of memory currently in use.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.10 Garbage Collector Before GC

Garbage Collector memory pool metrics before completion of garbage collection per node.

2.10.1 Name

Garbage Collector Name.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.10.2 Node ID

Coherence Node ID.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.10.3 GC Thread Count

GC Thread Count per garbage collector

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding

Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–29 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.10.4 Start Time (ms)

GC Start Time for a garbage collector.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.10.5 End Time (ms)

GC End Time for a garbage collector.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.10.6 Key of Memory Usage Before GC

Garbage Collector key.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.10.7 Committed of Memory Usage Before GC

The amount of memory that is guaranteed to be available for use by the JVM after GC.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–30 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.10.8 Init of Memory Usage Before GC

The initial amount of memory that the JVM requested for a memory pool from the operating system during startup.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–31 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.10.9 Max of Memory Usage Before GC

The maximum amount of memory that can be used by the JVM.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–32 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.10.10 Used of Memory Usage Before GC

The amount of memory currently in use.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–33 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.11 Garbage Collector Collection

Metrics from a garbage collection for each garbage collector per node.

2.11.1 Name

Garbage Collector Name.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.11.2 Node ID

Coherence Node ID

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.11.3 Collection Count

Number of items garbage collected

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–34 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.11.4 Garbage Collection Rate

Number of items garbage collected in a sample interval

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–35 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.11.5 Collection Time

Time to garbage collect items

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12 HTTP Session

Metrics indicating session usage by a web application using Coherence.

2.12.1 Application ID

Coherence Web Application ID.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.2 Node ID

Coherence Node ID.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.3 Collection Class Name

The fully qualified class name of the HttpSessionCollection implementation in use.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.4 Factory Class Name

The fully qualified class name of the Factory implementation in use.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.5 Local Attribute Cache Name

The name of the local cache that stores non-distributed session attributes.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.6 Number of Local Attributes in Local Attribute Cache

The number of non-distributed session attributes stored in the local session attribute cache.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.7 Local Session Cache Name

The name of the local cache that stores non-distributed sessions.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.8 Number of Local Attributes in Local Session Cache

The number of non-distributed sessions stored in the local session cache

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.9 Overflow Average Size

The average size (in bytes) of the session attributes stored in the "overflow" clustered cache.

2.12.10 Overflow Cache Name

The name of the clustered cache that stores the "large attributes" that exceed a certain size.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.11 Overflow Max Size

The maximum size (in bytes) of a session attribute stored in the "overflow" clustered cache.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.12 Overflow Threshold

The minimum length (in bytes) that the serialized form of an attribute value must be for that attribute value to be stored in the separate "overflow" cache.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.13 Updates to Attributes Stored in Overflow Cache

The number of updates to session attributes stored in the "overflow" clustered cache.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.14 Updates to Attributes Stored in Overflow Cache Per Min -Per Min

The number of updates to session attributes stored in the "overflow" clustered cache per minute.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.15 Servlet Context Cache Name

The name of the clustered cache that stores javax.servlet.ServletContext attributes.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.16 Servlet Context Name

The name of the web application Servlet Context.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.17 Session Average Lifetime

The average lifetime (in seconds) of session objects invalidated.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.18 Session Cache Name

The name of the clustered cache that stores serialized session objects.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.19 Session ID Length

The length (in characters) of generated session IDs.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.20 Session Object Maximum Size (bytes)

The maximum size (in bytes) of a session object placed in the session storage clustered cache.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.21 Number of Http Session Objects

The number of session objects that are pinned to this instance of the web application.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.22 Session Timeout

The session expiration time (in seconds).

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.23 Updates to Object Store in Session Cache -Per Node

The number of updates of session object stored in the session storage clustered cache.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.12.24 Updates to Object Store in Session Cache Per Min -Per Node

The number of updates of session object stored in the session storage clustered cache per minute.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.13 HTTP Session - Aggregated Over Nodes

Metrics indicating session usage by a web application using Coherence aggregated over all nodes where application session information is stored.

This metric is a subset of Http Session Metrics aggregated over nodes where application session data is stored. Therefore, metric column descriptions are the same as those contained in the Http Session metric, but Application ID is the key and aggregation of metrics happens over all nodes where application id session information is stored.

2.13.1 Application ID

The application id number.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–36 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.13.2 No. of Attributes in Local Attribute Cache

The number of attributes in the local attribute cache

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–37 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.13.3 No. of Sessions in Local Session Cache

The number of updates of session object stored in the session storage clustered cache per minute.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–38 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.13.4 Updates to Attributes Stored in Overflow Cache

The updates to attributes stored in the overflow cache.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–39 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.13.5 Updates to Attributes Stored in Overflow Cache Per Min

The number of attributes stored in the overflow cache per minute.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–40 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.13.6 Updates to Object Store in Session Cache

The updates to the object store in the session cache.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–41 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.13.7 Updates to Object Store in Session Cache Per Min

The number of updates to the object store in the session cache per minute.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–42 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.14 Memory Performance

Memory usage metrics for heap and non-heap memory for each Coherence node.

2.14.1 Node ID

Coherence Node ID.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.14.2 Heap Memory Committed for JVM to use (bytes)

The amount of heap memory that is guaranteed to be available for use by the JVM.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–43 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.14.3 Heap Memory that JVM initially Requests from OS (bytes)

The initial amount of heap memory that the JVM requested for a memory pool from the operating system during startup.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.14.4 Maximum Heap Memory That Can Be Used (bytes)

The maximum amount of heap memory that can be used by the JVM.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–44 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.14.5 Heap Memory Used (bytes)

The amount of heap memory currently in use.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–45 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.14.6 Non Heap Memory Committed for JVM to use (bytes)

The amount of non-heap memory that is guaranteed to be available for use by the JVM.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–46 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.14.7 Non Heap Memory that JVM initially Requests from OS (bytes)

The initial amount of non-heap memory that the JVM requested for a memory pool from the operating system during startup.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–47 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.14.8 Non Heap Memory Used (bytes)

The amount of non-heap memory currently in use.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–48 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.14.9 Object Pending Finalization Count

The approximate number of objects for which finalization is pending.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–49 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.14.10 Verbose

Boolean value indicating whether verbose logging is enabled or not.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.15 Node Performance - Aggregate Over Caches

The node performance metrics aggregated over caches are described.

2.15.1 Total Gets Sum

The total number of get() operations since the cache statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.15.2 Total Gets Sum Delta

The total number of get() operations during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–50 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.15.3 Total Puts Sum

The total number of put() operations since the cache statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.15.4 Total Puts Sum Delta

The total number of put() operations during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–51 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.16 Node Configuration

The node configuration metrics are described.

2.16.1 Buffer Publish Size

The buffer size of the unicast datagram socket used by the Publisher, measured in the number of packets. Changing this value at runtime is an inherently unsafe operation

that will pause all network communications and may result in the termination of all cluster services.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.2 Buffer Receive Size

The buffer size of the unicast datagram socket used by the Receiver, measured in the number of packets. Changing this value at runtime is an inherently unsafe operation that will pause all network communications and may result in the termination of all cluster services.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.3 Burst Count

The maximum number of packets to send without pausing. Anything less than one (e.g. zero) means no limit.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.4 Burst Delay

The number of milliseconds to pause between bursts. Anything less than one (e.g. zero) is treated as one millisecond.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.5 Flow Control Enabled

Indicates whether flow control is enabled for this node.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.6 Logging Format

Specifies how messages will be formatted before being passed to the log destination

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.7 Logging Level

Specifies which logged messages will be output to the log destination. Valid values are non-negative integers or -1 to disable all logger output.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.8 Logging Limit

The maximum number of characters that the logger daemon will process from the message queue before discarding all remaining messages in the queue. Valid values are integers in the range [0...]. Zero implies no limit.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.9 Machine Id

The Member's machine Id.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.10 Machine Name

A configured name that should be the same for all Members that are on the same physical machine, and different for Members that are on different physical machines.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.11 Member Name

A configured name that must be unique for every Member.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.12 Memory Max (MB)

The maximum amount of memory that the JVM will attempt to use in MB.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.13 Multicast Address

The IP address of the Member's MulticastSocket for group communication.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.14 Multicast Enabled

Specifies whether or not this Member uses multicast for group communication. If false, this Member will use the WellKnownAddresses to join the cluster and point-to-point unicast to communicate with other Members of the cluster.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.15 Multicast Port

The port of the member's MulticastSocket for group communication.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.16 Multicast Threshold

The percentage (0 to 100) of the servers in the cluster that a packet will be sent to, above which the packet will be multicasted and below which it will be unicasted.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.17 Multicast TTL

The time-to-live for multicast packets sent out on this Member's MulticastSocket.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.18 Nack Enabled

Indicates whether or not the early packet loss detection protocol is enabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.19 Process Name

A configured name that should be the same for Members that are in the same process (JVM), and different for Members that are in different processes. If not explicitly provided, for processes running with JRE 1.5 or higher the name will be calculated internally as the Name attribute of the system RuntimeMXBean, which normally represents the process identifier (PID).

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.20 Product Edition

The product edition this Member is running. Possible values are: Standard Edition (SE), Enterprise Edition (EE), Grid Edition (GE).

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.21 Rack Name

A configured name that should be the same for Members that are on the same physical "rack" (or frame or cage), and different for Members that are on different physical "racks".

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.22 RefreshTime

The timestamp when this model was last retrieved from a corresponding node. For local servers it is the local time.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.23 Resend Delay

The minimum number of milliseconds that a packet will remain queued in the Publisher's re-send queue before it is resent to the recipient(s) if the packet has not been acknowledged. Setting this value too low can overflow the network with unnecessary repetitions. Setting the value too high can increase the overall latency by delaying the re-sends of dropped packets. Additionally, change of this value may need to be accompanied by a change in SendAckDelay value.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.24 Role Name

A configured name that can be used to indicate the role of a Member to the application. While managed by Coherence, this property is used only by the application

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.25 Site Name

A configured name that should be the same for Members that are on the same physical site (e.g. data center), and different for Members that are on different physical sites.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.26 Timestamp

The date/time value (in cluster time) that this Member joined the cluster.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.27 Traffic Jam Count

The maximum total number of packets in the send and resend queues that forces the publisher to pause client threads. Zero means no limit.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.28 Traffic Jam Delay

The number of milliseconds to pause client threads when a traffic jam condition has been reached. Anything less than one (e.g. zero) is treated as one millisecond.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.29 Unicast Address

The IP address of the Member's DatagramSocket for point-to-point communication.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.16.30 Unicast Port

The port of the Member's DatagramSocket for point-to-point communication.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.17 Node Performance

The node performance metrics are described.

2.17.1 Cpu Count

Number of CPU cores for the machine this Member is running on.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.2 Machine Id

The Member's machine Id.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.3 Machine Name

A configured name that should be the same for all Members that are on the same physical machine, and different for Members that are on different physical machines.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.4 Member Name

A configured name that must be unique for every Member.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.5 Memory Available (MB)

The total amount of memory in the JVM available for new objects in MB.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–52 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	<	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "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 "Id" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.17.6 Packet Delivery Efficiency

The efficiency of packet loss detection and retransmission. A low efficiency is an indication that there is a high rate of unnecessary packet retransmissions.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.7 Packets Bundled

The total number of packets which were bundled prior to transmission. The total number of network transmissions is equal to (PacketsSent - PacketsBundled).

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.8 Packets Bundled (Delta Time)

The total number of packets which were bundled prior to transmission during the last collection interval. The total number of network transmissions is equal to (PacketsSent - PacketsBundled).

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.9 Packets Received

The number of packets received since the node statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.10 Packets Received (Delta Time)

The number of packets received during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.11 Packets Repeated

The number of duplicate packets received since the node statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.12 Packets Repeated (Delta Time)

The number of duplicate packets received since the node statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.13 Packets Resent

The number of packets resent since the node statistics were last reset. A packet is resent when there is no ACK received within a timeout period.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.14 Packets Resent (Delta Time)

The number of packets resent during the last collection interval. A packet is resent when there is no ACK received within a timeout period.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.15 Packets Resent Early

The total number of packets resent ahead of schedule. A packet is resent ahead of schedule when there is a NACK indicating that the packet has not been received.

Metric Summary

Multiple Thresholds

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.16 Packets Resent Early (Delta Time)

The total number of packets resent ahead of schedule during the last collection interval. A packet is resent ahead of schedule when there is a NACK indicating that the packet has not been received. .

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.17 Packets Resent Excess

The total number of packet retransmissions which were later proven unnecessary.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.18 Packets Resent Excess (Delta Time)

The total number of packet retransmissions sent during the last collection interval which were later proven unnecessary.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.19 Packets Sent

The number of packets sent since the node statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.20 Packets Sent (Delta Time)

The number of packets sent during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.21 Process Name

A configured name that should be the same for Members that are in the same process (JVM), and different for Members that are in different processes. If not explicitly provided, for processes running with JRE 1.5 or higher the name will be calculated internally as the Name attribute of the system RuntimeMXBean, which normally represents the process identifier (PID).

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.22 Publisher Success Rate

The publisher success rate for this cluster node since the node statistics were last reset. Publisher success rate is a ratio of the number of packets successfully delivered in a first attempt to the total number of sent packets. A failure count is incremented when there is no ACK received within a timeout period. It could be caused by either very high network latency or a high packet drop rate.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.23 Publisher Success Rate (Delta Time)

The publisher success rate for this cluster node during the last collection interval. Publisher success rate is a ratio of the number of packets successfully delivered in a first attempt to the total number of sent packets. A failure count is incremented when there is no ACK received within a timeout period. It could be caused by either very high network latency or a high packet drop rate.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–53 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "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 "Id" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.17.24 Rack Name

A configured name that should be the same for Members that are on the same physical "rack" (or frame or cage), and different for Members that are on different physical "racks".

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.25 Receiver Success Rate (Delta Time)

The receiver success rate from the viewing node to the viewed node during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.26 Receiver Success Rate Delta

The receiver success rate from the viewing node to the viewed node during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding

Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–54 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	<	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "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 "Id" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.17.27 RefreshTime

The timestamp when this model was last retrieved from a corresponding node. For local servers it is the local time.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.28 Role Name

A configured name that can be used to indicate the role of a Member to the application. While managed by Coherence, this property is used only by the application.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.29 Send Queue Size

The number of packets currently scheduled for delivery. This number includes both packets that are to be sent immediately and packets that have already been sent and awaiting for acknowledgment. Packets that do not receive an acknowledgment within ResendDelay interval will be automatically resent.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–55 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "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 "Id" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.17.30 Site Name

A configured name that should be the same for Members that are on the same physical site (e.g. data center), and different for Members that are on different physical sites.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.31 Tcp Ring Failures

The number of recovered TcpRing disconnects since the node statistics were last reset. A recoverable disconnect is an abnormal event that is registered when the TcpRing peer drops the TCP connection, but recovers after no more than maximum configured number of attempts. This value will be -1 if the TcpRing is disabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.32 Tcp Ring Failures (Delta Time)

The number of recovered TcpRing disconnects during the last collection interval. A recoverable disconnect is an abnormal event that is registered when the TcpRing peer drops the TCP connection, but recovers after no more than maximum configured number of attempts. This value will be -1 if the TcpRing is disabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.33 Tcp Ring Timeouts

The number of TcpRing timeouts since the node statistics were last reset. A timeout is a normal, but relatively rare event that is registered when the TcpRing peer did not ping this node within a heartbeat interval. This value will be -1 if the TcpRing is disabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.34 Tcp Ring TimeOuts (Delta Time)

The number of TcpRing timeouts during the last collection interval. A timeout is a normal, but relatively rare event that is registered when the TcpRing peer did not ping this node within a heartbeat interval. This value will be -1 if the TcpRing is disabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.35 Up Since

The date/time value (in cluster time) that this Member joined the cluster.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.17.36 Weakest Channel

The id of the cluster node to which this node is having the most difficulty communicating, or -1 if none is found. A channel is considered to be weak if either the point-to-point publisher or receiver success rates are below 1.0.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.18 Point to Point

Point to point metrics.

2.18.1 Deferred Packets

The number of packets addressed to the viewed member that the viewing member is currently deferring to send. The viewing member will delay sending these packets until the number of outstanding packets falls below the value of the Threshold attribute. The value of this attribute is only meaningful if the viewing member has FlowControl enabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.2 Deferring

Indicates whether or not the viewing member is currently deferring packets to the viewed member. The value of this attribute is only meaningful if the viewing member has FlowControl enabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.18.3 LastIn

The number of milliseconds that have elapsed since the viewing member last received an acknowledgment from the viewed member.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.4 LastOut

The number of milliseconds that have elapsed since the viewing member last sent a packet to the viewed member.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.5 Last Slow

The number of milliseconds that have elapsed since the viewing member declared the viewed member as slow, or -1 if the viewed member has never been declared slow.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.6 Outstanding Packets

The number of packets that the viewing member has sent to the viewed member which have yet to be acknowledged. The value of this attribute is only meaningful if the viewing member has FlowControl enabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.7 Paused

Indicates whether or not the viewing member currently considers the viewed member to be unresponsive. The value of this attribute is only meaningful if the viewing member has FlowControl enabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.8 Pause Rate

The percentage of time since the last time statistics were reset in which the viewing member considered the viewed member to be unresponsive. Under normal conditions this value should be very close to 0.0. Values near 1.0 would indicate that the viewed node is nearly inoperable, likely due to extremely long GC pauses. The value of this attribute is only meaningful if the viewing member has FlowControl enabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.9 Publisher Success Rate

The publisher success rate from the viewing node to the viewed node since the statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.10 Publisher Success Rate Delta

The publisher success rate for this cluster node during the last collection interval. Publisher success rate is a ratio of the number of packets successfully delivered in a first attempt to the total number of sent packets. A failure count is incremented when there is no ACK received within a timeout period. It could be caused by either very high network latency or a high packet drop rate.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.11 ReceiverSuccessRate

The receiver success rate from the viewing node to the viewed node since the statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.12 ReceiverSuccessRateDelta

The receiver success rate from the viewing node to the viewed node during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.13 Threshold

The maximum number of outstanding packets for the viewed member that the viewing member is allowed to accumulate before initiating the deferral algorithm. The value of this attribute is only meaningful if the viewing member has FlowControl enabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.18.14 Viewed Member Id

The ID of the member being viewed.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.19 Operating System

System properties and resources about the operating system on which the Coherence node JVM is running.

2.19.1 Node ID

Coherence Node ID.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.19.2 Architecture

Operating system architecture.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.19.3 Available Processors

Number of processors available to the Coherence node JVM.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.19.4 Committed Virtual Memory Size (bytes)

Amount of virtual memory that is guaranteed to be available to running Coherence node JVM in bytes.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–56 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.19.5 Free Physical Memory Size (bytes)

The amount of free physical memory.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–57 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.19.6 Free Swap Space Size (bytes)

The amount of free swap space.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–58 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.19.7 Max File Descriptor Count

Maximum permissible open file descriptor.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.19.8 Name

Operating system name.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.19.9 Open File Descriptor Count

Current count of open file descriptors.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–59 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.19.10 Process Cpu Time

The CPU time used by the process on which the Java virtual machine is running.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.19.11 Total Physical Memory Size (bytes)

The amount of total physical memory.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.19.12 Total Swap Space Size (bytes)

The amount of total swap space.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.19.13 Version

Operating system version.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.20 Resource Usage

Memory resource usage aggregated across all nodes.

2.20.1 Memory Usage (MB)

Memory usage is the difference between Maximum memory and Available memory aggregated across all nodes.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.21 Response

Response metrics.

2.21.1 Status

This metric shows the response status.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–60 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every Sample	=	Not Defined	0	1	The Coherence Cluster instance is down

2.22 Runtime

Coherence Node Runtime metrics

2.22.1 Row ID

Internal key metric

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.22.2 Node ID

Coherence Node ID

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.22.3 Input Arguments

Start or input parameters a Coherence Node JVM process is started with

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.22.4 VM Name

Virtual Machine name

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.22.5 VM Vendor

Virtual Machine vendor

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.22.6 VM Version

Virtual Machine version

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Seconds

2.23 Service Configuration

The service configuration metrics are described.

2.23.1 Partitions All

The total number of partitions that each cache storage is divided into.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.23.2 Requests Timeout Millis

The default timeout value in milliseconds for requests that can be timed-out (e.g. implement the `com.tangosol.net.PriorityTask` interface), but do not explicitly specify the request timeout value.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.23.3 Running

Specifies whether or not the service is running.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.23.4 Statistics

Statistics for this service in human readable format.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.23.5 High Availability Status

The High Availability status for this service. The value of MACHINE-SAFE means that all the cluster nodes running on any given machine could be stopped at once without data loss. The value of NODE-SAFE means that any cluster node could be stopped without data loss. The value of ENDANGERED indicates that abnormal termination of any cluster node that runs this service may cause data loss.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.23.6 Storage Enabled

Specifies whether local storage is enabled for this cluster Member.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.23.7 Task Hung Threshold Millis

The amount of time in milliseconds that a task can execute before it is considered hung. Note that a posted task that has not yet started is never considered as hung.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.23.8 Task Timeout Millis

The default timeout value in milliseconds for tasks that can be timed-out (e.g. implement the `com.tangosol.net.PriorityTask` interface), but do not explicitly specify the task execution timeout value.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.23.9 Thread Count

The number of threads in the service thread pool.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.23.10 Type

The type identifier of the service.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.24 Service Performance

The service performance metrics are described.

2.24.1 Backup Count

The number of backups for every cache storage.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.2 Owned Partitions Backup

The number of partitions that this Member backs up (responsible for the backup storage).

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.3 Owned Partitions Primary

The number of partitions that this Member owns (responsible for the primary storage).

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.4 Partitions Endangered

The total number of partitions that are not currently backed up.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.5 Partitions Unbalanced

The total number of primary and backup partitions which remain to be transferred until the partition distribution across the storage enabled service members is fully balanced.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.6 Partitions Vulnerable

The total number of partitions that are backed up on the same machine where the primary partition owner resides.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.7 Refresh Time

The timestamp when this model was last retrieved from a corresponding node. For local servers it is the local time.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.8 Request Average Duration

The average duration (in milliseconds) of an individual synchronous request issued by the service since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.9 Request Max Duration

The maximum duration (in milliseconds) of a synchronous request issued by the service since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.10 Request Pending Count

The number of pending synchronous requests issued by the service.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.11 Request Pending Duration

The duration (in milliseconds) of the oldest pending synchronous request issued by the service.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.12 Request Timedout Count

The total number of timed-out requests since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.13 Request Timeout Count Delta

The number of timed out requests per node since the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–61 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.24.14 Request Total Count

The total number of synchronous requests issued by the service since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.15 Request Total Count Delta

The total number of synchronous requests issued by the service during the last collection interval.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–62 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.24.16 Status HA

The High Availability status for this service. The value of MACHINE-SAFE means that all the cluster nodes running on any given machine could be stopped at once without data loss. The value of NODE-SAFE means that any cluster node could be stopped without data loss. The value of ENDANGERED indicates that abnormal termination of any cluster node that runs this service may cause data loss.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–63 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	MATCH	Not Defined	ENDANGERED	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.24.17 Storage Enabled Count

Specifies the total number of cluster nodes running this Service for which local storage is enabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.18 Task Average Duration

The average duration (in milliseconds) of an individual task execution.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.19 Task Backlog

The size of the backlog queue that holds tasks scheduled to be executed by one of the service threads.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.20 Task Count

The total number of executed tasks since the last time the statistics were reset

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.21 Task Count Delta

The total number of executed tasks during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.22 Task Hung Count

The total number of currently executing hung tasks.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.23 Task Hung Duration

The longest currently executing hung task duration in milliseconds.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.24 Task Hung Task Id

The ID of the longest currently executing hung task.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.25 Task Max Backlog

The maximum size of the backlog queue since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.26 Task Timeout Count

The total number of timed-out tasks since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.27 Task Timeout Count Delta

The total number of timed out tasks since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.28 Thread Abandoned Count

The number of abandoned threads from the service thread pool. A thread is abandoned and replaced with a new thread if it executes a task for a period of time longer than execution timeout and all attempts to interrupt it fail.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.29 Thread Abandoned Count Delta

The number of abandoned threads from the service thread pool during the last collection interval. A thread is abandoned and replaced with a new thread if it executes a task for a period of time longer than execution timeout and all attempts to interrupt it fail.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–64 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.24.30 Thread Average Active Count

The average number of active (not idle) threads in the service thread pool since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.31 Thread Count

The number of threads in the service thread pool.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–65 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.24.32 Thread Idle Count

The number of currently idle threads in the service thread pool.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.33 Thread Idle Count Delta

The number of threads that became idle in the service thread pool during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.24.34 Eviction Count

The number of evictions from the backing map managed by this StorageManager caused by entries expiry or insert operations that would make the underlying backing map to reach its configured size limit.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.24.35 Insert Count

The number of inserts into the backing map managed by this StorageManager. In addition to standard inserts caused by put and invoke operations or synthetic inserts caused by get operations with read-through backing map topology, this counter is incremented when distribution transfers move resources 'into' the underlying backing map and is decremented when distribution transfers move data 'out'.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.24.36 Listener Filter Count

The number of filter-based listeners currently registered with the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.24.37 Listener Key Count

The number of filter-based listeners currently registered with the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.24.38 Listener Registrations

The total number of listener registration requests processed by the StorageManager since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.24.39 Locks Granted

The number of locks currently granted for the portion of the partitioned cache managed by the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.24.40 Locks Pending

The number of pending lock requests for the portion of the partitioned cache managed by the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.24.41 Remove Count

The number of removes from the backing map managed by this StorageManager caused by operations such as clear, remove or invoke.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.25 Service Performance - Aggregated Over Nodes

The service performance metrics aggregated over nodes are described.

2.25.1 Backup Count

The number of backups for every cache storage.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.2 Owned Partitions Backup

The number of partitions that this Member backs up (responsible for the backup storage).

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.3 Owned Partitions Primary

The number of partitions that this Member owns (responsible for the primary storage).

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.4 Partitions Endangered

The total number of partitions that are not currently backed up.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.5 Partitions Unbalanced

The total number of primary and backup partitions which remain to be transferred until the partition distribution across the storage enabled service members is fully balanced.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.6 Partitions Vulnerable

The total number of partitions that are backed up on the same machine where the primary partition owner resides.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.7 Request Average Duration

The average duration (in milliseconds) of an individual synchronous request issued by the service since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.8 Request Max Duration

The maximum duration (in milliseconds) of a synchronous request issued by the service since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.9 Request Pending Count

The number of pending synchronous requests issued by the service.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.10 Request Pending Duration

The duration (in milliseconds) of the oldest pending synchronous request issued by the service.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.11 Request Timeout Count

The total number of timed-out requests since the last time the statistics were reset.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–66 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "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 "Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.25.12 Request Timeout Count Delta

The total number of synchronous requests issued by the service during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.13 Request Total Count

The total number of synchronous requests issued by the service since the last time the statistics were reset.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–67 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "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 "Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.25.14 Request Total Count Delta

The total number of synchronous requests issued by the service during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.15 Storage Enabled Count

Specifies the total number of cluster nodes running this service for which local storage is enabled.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.16 Task Average Duration

The average duration (in milliseconds) of an individual task execution.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.17 Task Backlog

The size of the backlog queue that holds tasks scheduled to be executed by one of the service pool threads.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.18 Task Count

The total number of executed tasks since the last time the statistics were reset

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.19 Task Count Delta

The total number of executed tasks during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.20 Task Hung Count

The total number of currently executing hung tasks.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.21 Task Hung Duration

The longest currently executing hung task duration in milliseconds.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.22 Task Max Backlog

The maximum size of the backlog queue since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.23 Task Timeout Count

The total number of timed-out tasks since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.24 Task Timeout Count Delta

The total number of timed-out tasks during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.25 Thread Abandoned Count

The number of abandoned threads from the service thread pool. A thread is abandoned and replaced with a new thread if it executes a task for a period of time longer than execution timeout and all attempts to interrupt it fail.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–68 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "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 "Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.25.26 Thread Abandoned Count Delta

The number of abandoned threads from the service thread pool during the last collection interval. A thread is abandoned and replaced with a new thread if it executes a task for a period of time longer than execution timeout and all attempts to interrupt it fail.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.27 Thread Average Active Count

The average number of active (not idle) threads in the service thread pool since the the statistics were last reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.28 Thread Count

The number of threads in the service thread pool.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–69 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Hours	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any "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 "Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.25.29 Thread Idle Count

The number of currently idle threads in the service thread pool.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.25.30 Thread Idle Count Delta

The number of threads that became idle in the service thread pool during the last collection interval.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

2.26 Storage Manager

The store manager metrics are described.

2.26.1 Events Dispatched - Per Node

The total number of events dispatched by the StorageManager since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.26.2 Eviction Count - Per Node

The number of evictions from the backing map managed by this StorageManager caused by entries expiry or insert operations that would make the underlying backing map to reach its configured size limit. The eviction count is used to audit the cache size in a static system. Cache Size = Insert Count - Remove Count - Eviction count. Therefore the eviction count is not reset by the reset statistics method.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.26.3 Insert Count - Per Node

The number of inserts into the backing map managed by this StorageManager. In addition to standard inserts caused by put and invoke operations or synthetic inserts caused by get operations with read-through backing map topology, this counter is incremented when distribution transfers move resources `into` the underlying backing map and is decremented when distribution transfers move data `out`. The insert count is used to audit the cache size in a static system. Cache Size = Insert Count - Remove Count - Eviction count. Therefore the insert count is not reset by the reset statistics method.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.26.4 Listener Filter Count - Per Node

The number of filter-based listeners currently registered with the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.26.5 Listener Key Count - Per Node

The number of key-based listeners currently registered with the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.26.6 Listener Registrations - Per Node

The total number of listener registration requests processed by the StorageManager since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.26.7 Locks Granted - Per Node

The number of locks currently granted for the portion of the partitioned cache managed by the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.26.8 Locks Pending - Per Node

The number of pending lock requests for the portion of the partitioned cache managed by the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.26.9 Remove Count - Per Node

The number of removes from the backing map managed by this StorageManager caused by operations such as clear, remove or invoke. The remove count is used to audit the cache size in a static system. Cache Size = Insert Count - Remove Count - Eviction count. Therefore the remove count is not reset by the reset statistics method.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.27 Store Manager - Aggregated Over Nodes

The storage manager aggregate metrics are described.

2.27.1 Events Dispatched

The total number of events dispatched by the StorageManager since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.27.2 Eviction Count Sum

The number of evictions from the backing map managed by this StorageManager caused by entries expiry or insert operations that would make the underlying backing map to reach its configured size limit.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.27.3 Insert Count

The number of inserts into the backing map managed by this StorageManager. In addition to standard inserts caused by put and invoke operations or synthetic inserts caused by get operations with read-through backing map topology, this counter is incremented when distribution transfers move resources `into` the underlying backing map and is decremented when distribution transfers move data `out`.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.27.4 Listener Filter Count Sum

The number of filter-based listeners currently registered with the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.27.5 Listener Key Count Sum

The number of key-based listeners currently registered with the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.27.6 Listener Registrations Sum

The total number of listener registration requests processed by the StorageManager since the last time the statistics were reset.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.27.7 Locks Granted Sum

The number of locks currently granted for the portion of the partitioned cache managed by the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.27.8 Locks Pending Sum

The number of pending lock requests for the portion of the partitioned cache managed by the StorageManager.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.27.9 Remove Count Sum

The number of removes from the backing map managed by this StorageManager caused by operations such as clear, remove, or invoke.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.27.10 Resource Usage

This metric shows the memory usage for the Coherence target.

2.27.11 Memory Usage (MB)

This metric shows the total memory used by all nodes in the cluster.

2.28 Threading

The threading metrics are described.

2.28.1 Node ID

Coherence Node ID.

Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Hour

2.28.2 Thread Count

Current live thread count for a Coherence node JVM.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–70 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

2.28.3 Peak Thread Count

Peak thread count for a Coherence node JVM.

Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Table 2–71 Metric Summary Table

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 300 Seconds	After Every Sample	>	Not Defined	Not Defined	1	Metric Value is %value%

Multiple Thresholds

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

If warning or critical threshold values are currently set for any unique combination of "Name" and "NodeID" 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 "Name" and "NodeID" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

HTTP Server

This chapter describes Oracle HTTP Server (OHS) metrics.

3.1 OHS Process Metrics

Provides metrics about Oracle HTTP Server processes.

3.1.1 Heap Usage (MB)

The amount of heap space (total program memory) used by Apache's parent process.

3.1.1.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes	After Every 12 Samples	>	Not Defined	Not Defined	2	Heap Usage is %value%MB

3.2 Oracle HTTP Server General Metrics

Provides general Oracle HTTP Server metrics.

3.2.1 Start Time (ms since Epoch)

This metric shows the start time of Oracle HTTP Server.

3.2.1.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.3 OHS Module Metrics

Contains metrics about Oracle HTTP Server modules.

3.3.1 Active Requests for a Module

This metric specifies the number of requests currently being handled by this module.

3.3.1.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

3.3.2 Request Handling Time, for a Module (seconds)

This metric specifies the average time (in seconds) it took the module to handle a request during the last interval. The interval is the period of time specified as the collection frequency of this module.

3.3.2.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

3.3.3 Request Handling Throughput, for a Module (requests per second)

This rate metric specifies the number of requests handled per second by this module during the last interval. The interval is the period of time specified as the collection frequency for this metric. This metric helps you determine the frequency of requests currently being served by the module. If the request handling throughput is very low, either there is no activity on the Web server for the module, or there is something else going on that is preventing the module from processing requests.

3.3.3.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

3.4 OHS Response Code Metrics

Includes metrics about the amount of Oracle HTTP Server response codes since the Oracle HTTP Server was last restarted.

3.4.1 HTTP 4xx errors

The number of HTTP 4xx errors generated since the Oracle HTTP Server was last restarted.

3.4.1.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 15 Minutes

3.4.2 HTTP 5xx errors

This metric specifies the number of HTTP 5xx errors generated since the Oracle HTTP Server was last restarted.

3.4.2.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 15 Minutes

3.5 OHS Server Metrics

Displays information about Oracle HTTP Server.

3.5.1 Busy Processes

This metric specifies the number of child servers that are busy.

3.5.1.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.5.2 Percentage of Busy Processes

This metric shows the percentage of Oracle HTTP Server child processes that are currently servicing requests.

3.5.2.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes	After Every 12 Samples	>	85	90	2	The percentage of processes that are busy handling requests is %value%%

3.5.2.2 User Action

To tune the percentage of Oracle HTTP Server child processes that are currently servicing requests, you can change the values for the MaxClients, MinSpareServers, and MaxSpareServers directives. You can change the values of the MaxClients, MinSpareServers and MaxSpareServers directives on the Oracle HTTP Server Performance Directives Configuration page in the Fusion Middleware Console. For more information about the MaxClients, MinSpareServers, and MaxSpareServers directives, go to the Apache HTTP Server Documentation section of the <http://www.apache.org> Web site, then navigate to the Directives information.

3.5.3 Active HTTP Connections

This metric shows the number of open HTTP connections.

3.5.3.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 12 Samples	>	135	140	2	The number of active connections is %value%

3.5.4 Connection Duration (seconds)

This metric shows the average time (in seconds) a connection was open during the last interval. The interval is the period of time specified as the collection frequency for this metric. A connection remains open until all of the requests that it is handling have been completed.

3.5.4.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

3.5.5 Error Rate (%)

This metric specifies the percentage of requests that encountered errors during the last interval. The interval is the period of time specified as the collection frequency for this metric.

3.5.5.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes	After Every 12 Samples	>	1	1.5	2	The percentage of requests that resulted in errors is %value%%%

3.5.5.2 User Action

Use the Fusion Middleware Control Console to examine the errors in this Oracle HTTP Server's error log file. For information about viewing Oracle HTTP Server error logs in the Fusion Middleware Control, see the "Viewing Oracle HTTP Server Logs" topic in the Fusion Middleware Administrator's Guide.

3.5.6 Idle Processes

This metric specifies the number of child servers that are available.

3.5.6.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.5.7 Active HTTP Requests

This metric specifies the number of requests currently being serviced.

3.5.7.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 12 Samples	>	135	140	2	The number of active requests is %value%

3.5.8 Request Processing Time (seconds)

This metric shows the average time it took to process a request during the last interval. The interval is the period of time specified as the collection frequency for this metric.

3.5.8.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 12 Samples	>	Not Defined	Not Defined	2	The average processing time for a request is %value% seconds

3.5.9 Request Processing Time (milliseconds)

This metric shows the average time (in milliseconds) it took to process a request during the last interval. The interval is the period of time specified as the collection frequency for this metric.

3.5.10 Request Throughput (requests per second)

This rate metric specifies the number of requests per second serviced during the last interval. The interval is the period of time specified as the collection frequency for this metric. This metric specifies how much of a load this Oracle HTTP Server is handling.

3.5.10.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

3.5.11 Response Data Processed (KB per response)

This metric specifies the average size of a response (in KB) during the last interval. The interval is the period of time specified as the collection frequency for this metric.

3.5.11.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes	After Every 12 Samples	>	Not Defined	Not Defined	2	The average size of a request is %value%KB

3.5.12 Response Data Throughput (KB per second)

This metric specifies the amount of response data (in KB/seconds) processed by Oracle HTTP Server during the last interval. The interval is the period of time specified as the collection frequency for this metric.

3.5.12.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.6 OHS Virtual Host Metrics

Provides virtual host information.

3.6.1 Active Requests for a Virtual Host

This metric specifies the total number of active requests currently being processed by the virtual host.

3.6.1.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes	After Every 12 Samples	>	135	140	2	The number of active requests for virtual host %name% is %value%

3.6.1.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "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 "Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

3.6.2 Request Processing Time for a Virtual Host (seconds)

This metric shows the average amount of time it took the virtual host to process one request during the last interval. The interval is the period of time specified as the collection frequency for this metric.

3.6.2.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes	After Every 12 Samples	>	Not Defined	Not Defined	2	The average processing time for a request for virtual host %name% is %value% seconds

3.6.2.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "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 "Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

3.6.3 Request Throughput for a Virtual Host (requests per second)

This rate metric specifies the number of requests per second serviced by the virtual host during the last interval. The interval is the period of time specified as the collection frequency for this metric. This metric specifies how much of a load this virtual host is handling.

3.6.3.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.6.4 Response Data Processed For a Virtual Host (KB per response)

This metric specifies the average size of a response (in KB) during the last interval. The interval is the period of time specified as the collection frequency for this metric.

3.6.4.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes	After Every 12 Samples	>	Not Defined	Not Defined	2	The average size of a request for virtual host %name% is %value%KB

3.6.4.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "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 "Name" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

3.6.5 Response Data Throughput For a Virtual Host (KB per second)

This metric specifies the amount of response data (in KB/second) processed by the virtual host during the last interval. The interval is the period of time specified as the collection frequency for this metric.

3.6.5.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.6.6 Virtual Host Type

This metrics provides the type of virtual host, which can be IP_DEFAULT, NAME_DEFAULT, IP_BASED, or NAME_BASED.

3.6.6.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.4.x and 10.1.2.x	Every 5 Minutes

3.7 Resource Usage

Provides information about CPU and memory usage.

3.7.1 CPU Usage (%)

Shows the percentage of CPU used by the Oracle HTTP server.

3.7.1.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes	After Every 12 Samples	>	Not Defined	Not Defined	2	CPU Utilization is %value%%%

3.7.2 CPU Idle Time (%)

Shows the percentage of time the CPU has been idle on the host.

3.7.2.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.7.3 Other CPU Usage (%)

Shows the percentage of CPU used by host processes other than Oracle HTTP Server. If you are having problems with CPU usage on the Oracle HTTP server host computer, this metric can help you determine whether or not Oracle HTTP server is causing the problem.

3.7.3.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.7.3.2 User Action

Navigate to the host Home page in the Fusion Middleware Control and review the top processes to see if you can identify which processes are consuming the most CPU. Or, navigate to the host Performance page in the Grid Control and view the processes that are consuming the most CPU.

3.7.4 Memory Usage (MB)

This metric shows the current physical memory used by Oracle HTTP Server.

3.7.4.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.7.5 Free Memory (MB)

This metric shows current free memory on the host that has Oracle HTTP Server.

3.7.5.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.7.6 Other Memory Usage (MB)

This metric shows current host memory used by processes other than Oracle HTTP Server. If you are having problems with CPU usage on the Oracle HTTP Server host computer, this metric can help you determine whether or not Oracle HTTP Server is causing this problem.

3.7.6.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.7.6.2 User Action

Navigate to the host Home page in the Fusion Middleware Control and review the top processes to see if you can identify which processes are consuming the most memory. Or, navigate to the host Performance page in the Grid Control and view the processes that are consuming the most memory.

3.7.7 Total Memory (MB)

Amount of physical memory (in MB) for the host where Oracle HTTP Server is running.

3.7.7.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.7.8 Memory Usage (%)

Shows the current percentage of physical memory used by the component.

3.7.8.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes	After Every 12 Samples	>	80	90	2	Memory Utilization is %value%%%

3.7.9 Free Memory (%)

This metric shows the percentage of current free memory on the host that has Oracle HTTP Server.

3.7.9.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.7.10 Other Memory Usage (%)

This metric specifies the percentage of other memory usage on the computer. It includes all the memory except for Oracle HTTP Server.

3.7.10.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x; 9.0.4.x and 10.1.2.x	Every 5 Minutes

3.7.11 Start Time (ms since Epoch)

This metric shows the start time of Oracle HTTP Server.

3.7.11.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

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

Target Version	Collection Frequency
9.0.4.x and 10.1.2.x	Every 5 Minutes

3.7.12 Up Time (ms since Epoch)

This metric shows the total time the Oracle HTTP Server has been running.

3.8 Response

Contains metrics that give availability information about Oracle HTTP Server.

3.8.1 Response - Status

This metric indicates whether or not Oracle HTTP Server is up or down. A status of 1 means the server is up, and a status of 2 means the server is down.

3.8.1.1 Metric Summary

The rest of the information in this section is only valid for this metric when it appears in Enterprise Manager Grid Control (if applicable).

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every 60 Samples	=	Not Defined	0	1	The Oracle HTTP Server instance is down

3.8.1.2 User Action

If the Oracle HTTP Server instance is down, start it using the Home page for the Oracle HTTP Server in the Fusion Middleware Control.

Part II

Oracle Fusion Middleware Metrics

This part provides a compilation of the Oracle Fusion Middleware target metrics provided in Oracle Enterprise Manager.

Part II contains the following chapters:

- [Chapter 4, "Java EE Application"](#)
- [Chapter 5, "Oracle Service Bus"](#)
- [Chapter 6, "Oracle SOA Infra"](#)
- [Chapter 7, "SOA Composite"](#)
- [Chapter 8, "Forms"](#)
- [Chapter 9, "Portal"](#)
- [Chapter 10, "Discoverer"](#)
- [Chapter 11, "Reports Bridge"](#)
- [Chapter 12, "Virtual Directory"](#)
- [Chapter 13, "Internet Directory"](#)
- [Chapter 14, "Directory Integration Platform"](#)
- [Chapter 15, "WebCenter"](#)
- [Chapter 16, "Web Cache"](#)

Java EE Application

This chapter describes metrics for Java EE applications deployed to Oracle WebLogic Server.

4.1 ADFC Taskflow Metrics

Metrics concerning ADFc (ADF Controller)

4.1.1 Total Active Taskflows

The count of active task flows in an ADF application since the application was run. This metric records the number of times that a task flow has been the active task flow for a user.

For example, if the user goes from task flow A to task flow B and then exits back to A, then task flow A will have been active two times and task flow B active once. Therefore the total active count would be three.

4.1.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.1.2 User Action

No user action.

4.1.2 Active Taskflows

This metric represents the number of currently active ADF task flows.

4.1.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.2.2 User Action

No user action.

4.1.3 Active throughput

The rate at which task flows are processed since the data was last collected.

4.1.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.3.2 User Action

No user action.

4.1.4 Active Taskflows

The number of active task flows processed since the last time metric data was collected.

4.1.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.4.2 User Action

No user action.

4.1.5 Total Entered Taskflows

This metric displays the number of task flows entered at the current time.

4.1.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.5.2 User Action

No user action.

4.1.6 Entered Taskflows

This metric records the number of times a task flow has been entered. Exiting back into a task flow from a called task flow is not the same as entering it. For example, if the user goes from task flow A to task flow B and then exits back to A, the total would be two, one entry for each into A and B.

4.1.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.6.2 User Action

No user action.

4.1.7 Invoke Taskflows

This metric records the number of times the task flow has been called by a task flow call activity.

Not every task flow is initiated by a task flow call activity. In most cases, this is probably going to be the same as number of entered task flows, but not all the time.

4.1.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.7.2 User Action

No user action.

4.1.8 Invoked Taskflows completed

The number of invoked task flows that have been completed at the current time.

4.1.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.8.2 User Action

No user action.

4.1.9 Taskflow Invoke time

The average time taken over the past five minutes to invoke ADF taskflows.

4.1.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.9.2 User Action

No user action.

4.1.10 Average Time

The average time to process task flows since the data was last collected.

4.1.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.10.2 User Action

No user action.

4.1.11 Maximum Time

The maximum time taken to process task flows since the data was last collected.

4.1.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.11.2 User Action

No user action.

4.1.12 Minimum Time

The minimum time taken to process the task flow since the data was last collected.

4.1.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.1.12.2 User Action

No user action.

4.2 Module Removals

The number of Application Module instances removed at the current time.

4.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.3 Pool Checkouts

The number of free Application Module instances checked out at the current time.

4.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.4 Average Creation time(ms)

The average creation time for an ApplicationModule instance since the data was last collected.

4.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.4.2 User Action

No user action.

4.5 Free Instances

The number of free Application Module instances at the current time.

4.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.6 Maximum Creation Time (ms)

The maximum creation time for an Application Module instance since the data was last collected.

4.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.7 Request Count

The number of main Application Module instances since the last request.

4.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.8 Available Instances in last 5 Minutes

For the selected ApplicationModule, the count of available ApplicationModule instances in the pool over the past five minutes.

4.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.8.2 User Action

No user action.

4.9 Available Instances in last 1 min

For the selected ApplicationModule, the count of available ApplicationModule instances in the pool over the past one minute.

4.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.9.2 User Action

No user action.

4.10 Unavailable Instances in last 5 Minutes

For the selected ApplicationModule, the number of unavailable ApplicationModule instances in the pool over the past five minutes.

4.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.10.2 User Action

No user action.

4.11 Unavailable Instances in last 1 Minute

For the selected ApplicationModule, the count of unavailable ApplicationModule instances in the pool over the past one minute.

4.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.11.2 User Action

No user action.

4.12 Activations

The name of the ApplicationModule to which the following metrics apply.

4.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.12.2 User Action

No user action.

4.13 Pool check Ins

The number of ApplicationModule instances checked into the ApplicationModule pool since the data was last collected.

4.13.1 Metric Summary

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

Target Version	Collection Frequency
----------------	----------------------

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.13.2 User Action

No user action.

4.14 Pool Check Outs

The number of ApplicationModule instances checked out of the ApplicationModule pool since the data was last collected.

4.14.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.14.2 User Action

No user action.

4.15 Pool Check Out Failures

4.15.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.16 Referenced Modules Reused

4.16.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.17 Reference Modules Reused

4.17.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.18 Unreference Modules Recycled

4.18.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.19 Datasource Metrics

This category provides information about datasource metrics.

4.19.1 Datasource - Available Connections

This metric shows the number of database connections currently available (not in use) for this data source.

4.19.2 Datasource - Connections Created (per minute)

This metric shows the average number of database connections created for this data source per minute in the last 5 minutes.

4.19.3 Datasource - Connections in Use

This metric shows the number of data source connections that are currently in use by applications.

4.19.4 Datasource - Connection Leaks (per minute)

For the selected data source, this metric shows the average number of JDBC connection leaks per minute over the last 5 minutes. A leaked connection is a connection that was reserved from the data source but was not returned to the data source when the connect was closed.

4.19.5 Datasource - Connection Pool Size

This metric shows the current number of JDBC connections in the connection pool in the selected data source.

4.19.6 Datasource - Connection Refresh Failures (per minute)

This metric shows the average number of times per minute that the data source attempted to refresh a database connection and failed over the past five minutes.

4.19.7 Datasource - Connection Request Failures (per minute)

This metric shows the average number of failed connection requests per minute in the last 5 minutes.

4.19.8 Datasource - Connection Requests (per minute)

This metric shows the average number of requests for a connection from this data source per minute over the last 5 minutes.

4.19.9 Datasource - Successful Connections (%)

This metric shows the percentage of requests that successfully returned connections for this data source over the last 5 minutes.

4.19.10 Datasource - Unavailable Connections

This metric shows the number of database connections that are currently unavailable (in use or being tested by the system) in this instance of the data source.

4.19.11 Datasource - Connection Requests Waiting

This metric shows the current number of connection wait requests for this data source. A connection wait request is a request that had to wait for a database connection.

4.19.12 Datasource - Failed Waiting Connection Requests (per minute)

For the current data source, this metric shows the average number of connection wait requests that failed per minute over the past five minutes.

A connection wait request is a request for a database connection that had to wait before getting a connection.

Connection wait requests can fail for a variety of reasons, including waiting for longer than the value of the `ConnectionReserveTimeoutSeconds` property.

4.19.13 Datasource - Connection Requests that Waited (per minute)

For the selected data source, this metric shows the average number of connection wait requests per minute in the last 5 minutes.

A connection wait request is a request for a connection that had to wait before getting a connection. This metric includes those that eventually got a connection and those that did not get a connection.

4.19.14 Datasource - Connection Wait Successes (%)

For the current data source, this metric shows the percentage of connection wait requests that successfully got a database connection during the last 5 minutes.

A connection wait request is a request for a connection that had to wait before getting a connection. This metric includes those that eventually got a connection and those that did not get a connection.

4.19.15 Datasource - Connection Wait Successes (per minute)

For this data source, this metric shows the average number of connection wait requests that succeeded per minute over the last 5 minutes.

A connection wait request succeeds when it eventually gets a database connection.

4.19.16 Datasource - State

The current state of the data source. Possible states are:

- Running - the data source is enabled (deployed and not suspended). This is the normal state of the data source.
- Suspended - the data source has been manually disabled.
- Shutdown - the data source is shutdown and all database connections have been closed.
- Overloaded - all resources in pool are in use.
- Unhealthy - all connections are unavailable (not because they are in use). This state occurs if the database server is unavailable when the data source is created (creation retry must be enabled) or if all connections have failed connection tests (on creation, on reserve, or periodic testing).
- Unknown - the data source state is unknown

4.19.17 Datasource - Statements Added to Cache (per minute)

This metric shows the number of statements per minute that were added to the statement cache for this data source in the last 5 minutes.

4.19.18 Datasource - Statements Discarded from Cache (per minute)

This metric shows the average number of statements per minute that were discarded from the cache over the past five minutes.

Each connection in the connection pool has its own cache of statements. This number is the sum of the number of statements that were discarded from the caches for all connections in the connection pool.

4.19.19 Datasource - Cached Statements not Used (per minute)

This metric shows the average number of statements per minute not satisfied by the statement cache in the last 5 minutes.

Each connection in the connection pool has its own cache of statements.

4.19.20 Datasource - Cached Statements Used (%)

This metric shows the percentage of statements satisfied by the statement cache during the last 5 minutes.

Each connection in the connection pool has its own cache of statements.

4.19.21 Datasource - Cached Statements Used (per minute)

This metric shows the average number of statements per minute satisfied by the data source statement cache in the last 5 minutes.

Each connection in the connection pool has its own cache of statements.

4.19.22 Datasource - Statement Cache Size

This metric shows the number of prepared and callable statements currently cached in the statement.

Each connection in the connection pool has its own cache of statements. This number is the sum of the number of statements in the caches for all connections in the connection pool.

4.20 EJB Module Metrics

This category provides information about EJB module metrics.

4.20.1 EJB Module - Cached Beans

This metric shows the total number of beans from this EJB Home that are currently in the EJB cache.

4.20.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.2 EJB Module - Bean Activations (per minute)

This metric shows the average number of beans per minute that have been activated from the EJB home over the last 5 minutes.

4.20.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.3 EJB Module - Cache Hits (%)

This metric shows the percentage of cache accesses that completed successfully in the last 5 minutes.

4.20.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.4 EJB Module - Cache Misses (per minute)

This metric shows the average number of cache misses per minute for this EJB module in the last 5 minutes.

4.20.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.5 EJB Module - Bean Accesses (per minute)

This metric shows the average EJB pool accesses per minute over the last 5 minutes.

4.20.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.6 EJB Module - Bean Access Successes (%)

This metric shows the percentage of EJB pool accesses that were successful in the last 5 minutes.

4.20.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.7 EJB Module - Free Bean Instances

This metric shows the current number of available EJB instances in the free pool.

4.20.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.8 EJB Module - Bean Destroys (per minute)

This metric shows the average number of EJB destroys per minute for this EJB module over the last 5 minutes.

4.20.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.9 EJB Module - Bean Access Failures (per minute)

This metric shows the average number of times per minute that a failed attempt was made to get an instance from the free pool. This value is averaged over the past five minutes.

An Attempt to get a bean from the pool will fail if there are no available instances in the pool.

4.20.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.10 EJB Module - Beans In Use

This metric shows the number of EJB instances currently being used from the free pool.

4.20.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.11 EJB Module - Bean Transaction Commits (per minute)

This metric shows the average number of EJB transaction commits per minute for this EJB Module over the last 5 minutes.

4.20.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.12 EJB Module - Bean Transaction Rollbacks (per minute)

This metric shows the average number of EJB transaction rollbacks per minute for this EJB module over the last 5 minutes.

4.20.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.20.13 EJB Module - Bean Transaction Timeouts (per minute)

This metric shows the average number of EJB transaction timeouts per minute for this EJB module over the last 5 minutes.

4.20.13.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.21 EJB Transaction Metrics

This category provides information about EJB transaction metrics.

4.21.1 EJB Transaction - Bean Transaction Commits (%)

This metric shows the percentage of EJB transactions that were committed for the select EJB over the last 5 minutes.

4.21.2 EJB Transaction - Bean Transaction Commits (per minute)

This metric shows the average number of EJB transaction commits per minute for this EJB over the last 5 minutes.

4.21.3 EJB Transaction - Bean Transaction Rollbacks (per minute)

This metric shows the average number of transaction rollbacks per minute for this EJB over the last 5 minutes.

4.21.4 EJB Transaction - Bean Transaction Timeouts (per minute)

This metrics shows the average number of transaction timeouts per minute for this EJB over the last 5 minutes.

4.22 Overview Metrics

This category provides information about the EJB overview metrics.

4.22.1 Cached Beans

This metric shows the total number of EJBs currently active in the EJB cache.

4.22.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.2 Cache Accesses (per minute)

This metric shows the average number of EJB cache access attempts per minute for this application in the last 5 minutes.

4.22.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.3 Bean Activations (per minute)

This metric shows the average number of Enterprise Java Beans that have been activated per minute over the last 5 minutes.

4.22.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.4 Cache Hits (%)

This metrics indicates the percentage of cache accesses that completed successfully in the last 5 minutes.

4.22.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.5 Cache Misses (per minute)

This metric shows the average number of EJB cache misses per minute in this application in the last 5 minutes.

4.22.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.6 Bean Accesses (per minute)

This metrics shows the average number of Pool accesses per minute for this application in the last 5 minutes.

4.22.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.7 Bean Successes (%)

This metric shows the percentage of EJB pool accesses that were successful for this application in the last 5 minutes.

4.22.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.8 Free Bean Instances

This metric shows the current number of available EJB instances in the free pool.

4.22.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.9 Bean Destroys (per minute)

This metric shows the average number of EJB destroys per minute for this application in the last 5 minutes.

4.22.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.10 Bean Access Failures (per minute)

This metric shows the average number of failed pool accesses per minute for this application in the last 5 minutes.

4.22.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.11 Beans In Use

This metric shows the number of EJB instances currently being used from the free pool.

4.22.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.12 Bean Transaction Commits (%)

This metric shows the percentage of EJB transactions that were committed for this application during the last 5 minutes.

4.22.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.13 Bean Transaction Commits (per minute)

This metric shows the average number of EJB transaction commits per minute for this application in the last 5 minutes.

4.22.13.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.14 Bean Transaction Rollbacks (per minute)

This metric shows the average number of EJB transaction rollbacks per minute for this application over the last 5 minutes.

4.22.14.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.15 Bean Transaction Timeouts (per minute)

This metric shows the average number of EJB transaction timeouts per minute for this application over the past 5 minutes.

4.22.15.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.16 MDB Messages (per minute)

This metric shows the average number of messages processed by Message Driven Beans (MDBs) per minute for this application in the last 5 minutes.

4.22.16.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.17 Requests (per minute)

This metrics shows the average number of servlet and/or JSP invocations per minute in this application in the last 5 minutes.

4.22.17.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.18 Request Processing Time (ms)

This metric displays the average amount of time (in milliseconds) spent executing servlets and/or JSPs in this application in the last 5 minutes.

4.22.18.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.19 Active Sessions

This metric shows the number of active sessions for this application.

4.22.19.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.20 Work Manager Pending Requests

This metric shows the number of Work Manager requests waiting in the queue.

4.22.20.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.22.21 Work Manager Requests (per minute)

This metric shows the average number of Work Manager requests processed per minute for this application over the last 5 minutes.

4.22.21.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.23 Servlet/JSP Metrics

This category provides information about servlet/JSP metrics.

4.23.1 Servlet/JSP - Reloads (per minute)

This metric shows the average number of reloads of the selected servlet or JSP per minute over the last 5 minutes.

4.23.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.23.2 Servlet/JSP - Requests (per minute)

This metric shows the average number of invocations of the selected servlet or JSP per minute over the last 5 minutes.

4.23.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.23.3 Servlet/JSP - Request Processing Time (ms)

This metric shows the average amount of time (in milliseconds) that was spent executing the selected servlet or JSP over the last 5 minutes.

4.23.3.1 Metric Summary

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

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 5 Minutes
--------------	-----------------

4.24 Status Metrics

This category provides information about status metrics.

4.24.1 Server Name

This metric shows the name of the current WebLogic server.

4.24.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.24.2 State

This metric shows the current state of the application.

This metric specifies whether this application version is the currently active version. An application can be the only version currently deployed, or it can have more than one version currently deployed, using the side-by-side deployment feature.

If more than one version is deployed, only one version can be active.

Values for this metric can be:

- INACTIVE - it has not been activated yet, or that there is more than one version of the application deployed (using side-by-side deployment) and this one is retiring.
- ACTIVE_ADMIN - the currently active version for administrative channel requests.
- ACTIVE - the currently active version for normal (non-administrative) channel requests.

4.24.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.25 Web Module Metrics

This category provides information about Web module metrics.

4.25.1 Web Module - Requests (per minute)

This metric shows the average number of servlet and JSP invocations per minute for this Web module in the last 5 minutes.

4.25.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.25.2 Web Module - Request Processing Time (ms)

This metric shows the average amount of time (in milliseconds) spent executing servlets and/or JSPs in this Web module for the last 5 minutes.

4.25.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.25.3 Web Module - Active Sessions

This metric shows the number of active sessions for this web module.

4.25.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.26 MDS Metrics

Metrics concerning MDS (Metadata Services)

4.26.1 Metadata Object Gets Per Second

The number of metadata object gets per second.

4.26.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.26.1.2 User Action

High values (100s or 1000s or more) may occur for a heavily used and/or complex application. No action needs to be taken unless application performance is perceived to be too slow, in which case it might be advisable to increase the hardware resources available to the application.

4.26.2 Metadata Object Get Processing Time (seconds)

The average elapsed time in seconds to get a metadata object.

4.26.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 3 Samples	>	Not Defined	Not Defined	2	The average processing time of each Metadata Object Get is %value% seconds.

4.26.2.2 User Action

This value should be considerably less than 1 second. If the value is greater than about 0.01, check the cache efficiency metrics (IOs Per Metadata Object Get and IOs Per MO Content Get), and increase the MDS cache size if those metrics indicate a problem. If those metrics are normal, the CPU may be overloaded, meaning the application may need more hardware resources, especially if performance is perceived to be slow.

4.26.3 IOs Per Document Get

The average number of physical IOs per persistence-level document get. This is an indication of the efficiency of the document cache portion of the metadata cache for the application. Ideally it should be much less than 1.

4.26.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 3 Samples	>	Not Defined	Not Defined	2	The average number of physical IOs per Document Get is %value%.

4.26.3.2 User Action

If the value is close to or more than 1, this could indicate that the MDS cache for the application is too small. Try increasing the size of the MDS cache to see if it reduces the value of this statistic.

4.26.4 IOs Per MO Content Get

The average number of physical IOs per metadata object XML content retrieval. This is an indication of the efficiency of the metadata cache for the application. Ideally it should be much less than 1.

4.26.4.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 3 Samples	>	Not Defined	Not Defined	2	The average number of physical IOs per MO Content Get is %value%.

4.26.4.2 User Action

If the value is close to or more than 1, this could indicate that the MDS cache for the application is too small. Try increasing the size of the MDS cache to see if it reduces the value of this statistic.

4.26.5 IOs Per Metadata Object Get

The average number of physical IOs per metadata object get. This is an indication of the efficiency of the metadata cache for the application. Ideally it should be much less than one.

4.26.5.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 3 Samples	>	Not Defined	Not Defined	2	The average number of physical IOs per Metadata Object Get is %value%.

4.26.5.2 User Action

If the value is close to or more than 1, this could indicate that the metadata cache for the application is too small. Try increasing the size of the MDS cache to see if it reduces the value of this statistic.

4.26.6 Metadata Updates Per Second

The average number of metadata objects updated per second.

4.26.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

4.26.6.2 User Action

If the value of this metric is significant compared to the number of HTTP requests serviced per second, it means many users are creating or updating customizations or personalizations using the page composer or, more likely, via inherent changes. In this case, if application performance is perceived to be slow, it might be wise to reduce the number of these updates by configuring the application to use a different ADF change manager (so the changes are saved in memory only, not to disk).

4.26.7 Metadata Update Processing Time (seconds)

The average elapsed time in seconds taken to update a metadata object.

4.26.7.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 3 Samples	>	Not Defined	Not Defined	2	The average processing time of each Metadata Update is %value% seconds.

4.26.7.2 User Action

If the value of this metric exceeds about 1 second the MDS repository database may need tuning. If the "Metadata Updates Per Second" metric is also high, steps may need to be taken to reduce the number of updates (see user action for that metric).

4.26.8 UpDown Status

The up or down status of the application. A value of one (1) indicates that the application is up and running; a value of zero (0) indicates that the application is down.

4.26.8.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every 60 Samples	=	Not Defined	0	1	The J2EE Application is down

Oracle Service Bus

This chapter describes Oracle Service Bus (OSB) metrics.

5.1 Oracle Service Bus

Oracle Service Bus is a proven market-leading Enterprise Service Bus (ESB) built from the ground up for Service-Oriented Architecture (SOA) lifecycle management that provides foundation capabilities for service discovery and intermediation, rapid service provisioning and deployment, and governance. This service-infrastructure software adheres to the SOA principles of building coarse grained, loosely coupled, and standards-based services, creating neutral container in which business functions may connect service consumers and back-end business services, regardless of underlying infrastructure.

Built to meet exacting standards for reliability, availability, scalability, and performance, Oracle Service Bus uniquely combines the integration capabilities of an ESB with operational service management, into a single enterprise-class software product, with a layered functional architecture.

You can use the All Metrics page of this target to view the metrics that have been collected for this target by Oracle Management Agent.

5.2 Flow Component Resource Details Metrics

This category provides information about the resource type FLOW_COMPONENT.

5.2.1 Average Response Time (Flow Component)

This metric shows the average response time of the resource type (FLOW_COMPONENT) since last collection (75 minutes).

5.2.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Average response time since last collection is %value%

5.2.1.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Service ID", "Service Type", "Resource Name", and "Resource Type" objects.

If warning or critical threshold values are currently set for any unique combination of "Service ID", "Service Type", "Resource Name", and "Resource 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 "Service ID", "Service Type", "Resource Name", and "Resource Type" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.2.2 Open Instance Count (FLOW_COMPONENT)

This metric shows the open instance count of the resource type (FLOW_COMPONENT) since last collection (75 minutes).

5.2.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of opened instances since last collection is %value%

5.2.2.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Service ID", "Service Type", "Resource Name", and "Resource Type" objects.

If warning or critical threshold values are currently set for any unique combination of "Service ID", "Service Type", "Resource Name", and "Resource 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 "Service ID", "Service Type", "Resource Name", and "Resource Type" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.2.3 Error Instance Count (Flow Component)

This metric shows the error instance count of the resource type (FLOW_COMPONENT) since last collection (75 minutes).

5.2.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of faulted instances since last collection is %value%

5.2.3.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Service ID", "Service Type", "Resource Name", and "Resource Type" objects.

If warning or critical threshold values are currently set for any unique combination of "Service ID", "Service Type", "Resource Name", and "Resource 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 "Service ID", "Service Type", "Resource Name", and "Resource Type" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.3 End Point URI Resource Details Metrics

This category provides information about the resource type URI.

5.3.1 Average Response Time (End Point URI)

This metric shows the average response time of the resource type (URI) since last collection (75 minutes).

5.3.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

This table lists the metric's collection and threshold information.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Average response time since last collection is %value%

5.3.1.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Service ID", "Service Type", "Resource Name", and "Resource Type" objects.

If warning or critical threshold values are currently set for any unique combination of "Service ID", "Service Type", "Resource Name", and "Resource 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 "Service ID", "Service Type", "Resource Name", and "Resource Type" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.3.2 Open Instance Count (End Point URI)

This metric shows the open instance count of the resource type (URI) since last collection (75 minutes).

5.3.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of opened instances since last collection is %value%

5.3.2.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Service ID", "Service Type", "Resource Name", and "Resource Type" objects.

If warning or critical threshold values are currently set for any unique combination of "Service ID", "Service Type", "Resource Name", and "Resource 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 "Service ID", "Service Type", "Resource Name", and "Resource Type" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.3.3 Error Instance Count (End Point URI)

This metric shows the error instance count of the resource type (URI) since last collection (75 minutes).

5.3.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of faulted instances since last collection is %value%

5.3.3.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Service ID", "Service Type", "Resource Name", and "Resource Type" objects.

If warning or critical threshold values are currently set for any unique combination of "Service ID", "Service Type", "Resource Name", and "Resource 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 "Service ID", "Service Type", "Resource Name", and "Resource Type" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.4 End Point Operation Resource Details Metrics

This category provides information about the resource type WEBSERVICE_OPERATION.

5.4.1 Average Response Time (End Point Operation)

This metric shows the average response time of the resource type (WEBSERVICE_OPERATION) since last collection (75 minutes).

5.4.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Average response time since last collection is %value%

5.4.1.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Service ID", "Service Type", "Resource Name", and "Resource Type" objects.

If warning or critical threshold values are currently set for any unique combination of "Service ID", "Service Type", "Resource Name", and "Resource 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 "Service ID", "Service Type", "Resource Name", and "Resource Type" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.4.2 Open Instance Count (End Point Operation)

This metric shows the open instance count of the resource type (WEBSERVICE_OPERATION) since last collection (75 minutes).

5.4.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of opened instances since last collection is %value%

5.4.2.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Service ID", "Service Type", "Resource Name", and "Resource Type" objects.

If warning or critical threshold values are currently set for any unique combination of "Service ID", "Service Type", "Resource Name", and "Resource 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 "Service ID", "Service Type", "Resource Name", and "Resource Type" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.4.3 Error Instance Count (End Point Operation)

This metric shows the error instance count of the resource type (WEBSERVICE_OPERATION) since last collection (75 minutes).

5.4.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of faulted instances since last collection is %value%

5.4.3.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Service ID", "Service Type", "Resource Name", and "Resource Type" objects.

If warning or critical threshold values are currently set for any unique combination of "Service ID", "Service Type", "Resource Name", and "Resource 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 "Service ID", "Service Type", "Resource Name", and "Resource Type" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.5 OSB Service Details Metrics

This category provides metrics related to Oracle Service Bus services.

5.5.1 Contains Sensitive Data

This metric indicates whether or not the service handles sensitive data.

5.5.2 Created User

This metric shows the name of the user who created the service.

5.5.3 Created Time

This metric shows the time when the service was created.

5.5.4 End Point URI

This metric shows the end point URI of the service.

5.5.5 Last Modified User

This metric shows the name of the user who last modified the service.

5.5.6 Last Modified Time

This metric shows the time when the service was last modified time.

5.5.7 Protocol

This metric shows the protocol of the service it operates on.

5.6 Status

This metric shows the status of the service. The status can be either Up or Down.

5.7 Version

This metric shows the version of Oracle Service Bus.

5.8 OSB Service Metrics

This category provides metrics related to proxy services and business services running on Oracle Service Bus.

5.8.1 Average Response Time

This metric shows the average time taken by the service to respond since last collection. The information is collected every 75 minutes.

5.8.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Average response time since last collection is %value%

5.8.1.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.8.2 Total Average Response Time

This metric shows the total average time taken by the service to respond since the time the server was restarted.

5.8.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total average response time is %value%

5.8.2.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.8.3 Open Instance Count

This metric shows the number of open instances of the service since last collection. The information is collected every 75 minutes.

5.8.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of opened instances since last collection is %value%

5.8.3.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.8.4 Total Open Instance Count

This metric shows the total number of messages handled by this service since the time the server was restarted.

5.8.4.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total number of opened instances are %value%

5.8.4.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.8.5 Error Instance Count

This metric shows the number of errored instances for the service since last collection. The information is collected every 75 minutes.

5.8.5.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of faulted instances since last collection is %value%

5.8.5.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.8.6 Total Error Instance Count

This metric shows the total number of error instances occurred on this service since the time the server was restarted.

5.8.6.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total number of faulted instances are %value%

5.8.6.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.8.7 Failure Rate

This metric shows the failure rate of messages since last collection.

5.8.7.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Failure Rate since last collection is %value%

5.8.7.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.8.8 Maximum Response Time

This metric shows the maximum time taken by this service to respond since the time the server was restarted.

5.8.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.8.9 Minimum Response Time

This metric shows the minimum time taken by this service to respond since the time the server was restarted.

5.8.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.8.10 Service Name

This metric shows the name of the proxy or business service running on Oracle Service Bus.

5.8.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.8.11 Path

This metric shows the current path to the project and resource directory hierarchy in the configuration.

5.8.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.8.12 Project Name

This metric shows the name of the project associated with Oracle Service Bus.

5.8.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.8.13 Service Type

This metric shows the type of service running on Oracle Service Bus. The service type can be Proxy or Business.

5.8.13.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.8.14 WebService Security Violation Instance Count

This metric shows the number of web service security violation caused by the service since last collection. The information is collected every 75 minutes.

5.8.14.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of webservice security violation instances since last collection is %value%

5.8.14.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.8.15 Total WebService Security Violation Instance Count

This metric shows the total number of web service security violations caused by the service since the time the server was restarted.

5.8.15.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding

Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total number of webservice security violation instances are %value%

5.8.15.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.9 OSB Service Dependency Metrics

This category provide metrics related to the services that are dependent on a particular service.

5.9.1 Dependent Service ID

This metric shows the ID of the service that is dependent on this particular service.

5.10 Response

This category provides status metrics for Oracle Service Bus.

5.10.1 Status

This metric shows the current status of Oracle Service Bus. The status can be either Up or Down.

5.10.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text

All Versions	Every Minute	After Every Sample =	Not Defined	0	1	ALSB IS DOWN.....
--------------	--------------	----------------------	-------------	---	---	-------------------

5.11 SLA Alert Metrics

This category provides information about the SLA alerts generated in the OSB console.

5.11.1 Service Name

This metric shows the name of the service (Proxy/Business).

5.11.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.11.2 Project Name

This metric shows the name of the project.

5.11.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.11.3 SLA Severity Critical Alerts

This metric shows the number of SLA severity alerts since last collection (Severity Level - Critical, interval - 75 minutes).

5.11.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of OSB Severity critical alerts since last collection is %value%

5.11.3.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.11.4 Total SLA Severity Critical Alerts

This metric shows the total number of SLA severity alerts since the time the server restarted(Severity Level - critical).

5.11.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.11.5 SLA Severity Fatal Alerts

This metric shows the number of SLA severity alerts since last collection (Severity Level - Fatal, interval - 75 minutes).

5.11.5.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of OSB Severity fatal alerts since last collection is %value%

5.11.5.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.11.6 Total SLA Severity Fatal Alerts

This metric shows the total number of SLA severity alerts since the time the server was restarted (Severity Level - fatal).

5.11.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.11.7 SLA Severity Major Alerts

This metric shows the number of SLA severity alerts since last collection (Severity Level - Major, interval - 75 minutes).

5.11.7.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of OSB Severity minor alerts since last collection is %value%

5.11.7.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.11.8 Total SLA Severity Major Alerts

This metric shows the total number of SLA severity alerts since the time the server was restarted (Severity Level - Major).

5.11.8.1 Metric Summary

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

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 75 Minutes
--------------	------------------

5.11.9 SLA Severity Minor Alerts

This metric shows the number of SLA severity alerts since last collection (Severity Level - Minor, interval - 75 minutes).

5.11.9.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of OSB Severity major alerts since last collection is %value%

5.11.9.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.11.10 Total SLA Severity Minor Alerts

This metric shows the total number of SLA severity alerts since the time the server was restarted(Severity Level - Minor).

5.11.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.11.11 SLA Severity Warning Alerts

This metric shows the number of SLA severity alerts since last collection (Severity Level - Warning, interval - 75 minutes).

5.11.11.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 75 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Number of OSB Severity warning alerts since last collection is %value%

5.11.11.2 Multiple Thresholds

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

If warning or critical threshold values are currently set for any "Service 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 "Service ID" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

5.11.12 Total SLA Severity Warning Alerts

This metric shows the total number of SLA severity alerts since the time the server was restarted(Severity Level - Warnings).

5.11.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

5.11.13 Service Type

This metric shows the type of service either Proxy or Business.

5.11.13.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 75 Minutes

This chapter describes Oracle SOA Infra metrics.

6.1 B2B Active Entities Metrics

This category provides information about B2B active entities metrics.

6.1.1 Number of Active Agreements

This metric shows the number of active agreements.

6.1.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.1.2 Number of Active Documents

This metric shows the number of active documents.

6.1.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.1.3 Number of Active Trading Partners

This metric shows the number of active trading partners.

6.1.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.2 B2B Document Metrics Table

This category provides information about B2B Document Metrics Table metrics.

6.2.1 Number of Inbound Messages

This metric shows the number of inbound messages.

6.2.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.2.2 Number of inbound message faults

This metric shows the number of inbound message faults.

6.2.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.2.3 Inbound Message Size (KB)

This metric shows the inbound message size.

6.2.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.2.4 Inbound Messages processing time

This metric shows the processing time for inbound messages.

6.2.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.2.5 B2B Document Name

This metric shows the B2B document name.

6.2.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.2.6 Number of Outbound Messages

This metric shows the number of outbound messages.

6.2.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.2.7 Number of outbound message faults

This metric shows the number of outbound message faults.

6.2.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.2.8 Outbound Message Size (KB)

This metric shows the outbound message size.

6.2.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.2.9 Outbound Messages processing time

This metric shows the processing time for outbound messages.

6.2.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.3 B2B Inbound Endpoint Metrics

This category provides information about B2B Inbound Endpoint metrics.

6.3.1 B2B Inbound Endpoint Status

This metric shows the status of the B2B inbound endpoint.

6.3.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.4 B2B Outbound Endpoint Metrics

This category provides information about B2B Outbound Endpoint metrics.

6.4.1 B2B Outbound Endpoint Status

This metric shows the status of the B2B outbound endpoint.

6.4.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.5 B2B Trading Partner Metrics

This category provides information about B2B Trading Partner metrics.

6.5.1 Number of inbound messages

This metric shows the number of inbound messages for B2B trading partners.

6.5.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.5.2 Number of inbound message faults

This metric shows the number of inbound message faults for B2B trading partners.

6.5.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.5.3 Inbound Message Size (KB)

This metric shows the inbound size of messages for B2B trading partners.

6.5.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.5.4 Processing time of Inbound Messages

This metric shows the processing time of inbound messages for B2B trading partners.

6.5.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.5.5 Trading partner name

This metric shows the name of the trading partner.

6.5.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.5.6 Number of outbound messages

This metric shows the number of outbound messages for B2B trading partners.

6.5.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.5.7 Number of outbound message faults

This metric shows the number of outbound message faults for B2B trading partners.

6.5.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.5.8 Outbound Message Size (KB)

This metric shows the outbound size of messages for B2B trading partners.

6.5.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.5.9 Processing time of outbound messages

This metric shows the processing time of outbound messages for B2B trading partners.

6.5.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.6 BPEL Engine Requests Rate Metric Table

Displays the BPEL Engine request related metrics

6.6.1 Active request throughput in the last 5 minutes

This metric shows the active request throughput in the last five minutes in the BPEL service engine.

6.6.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.6.2 Last active request value

This metric shows the value for the last active request in the BPEL service engine.

6.6.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.6.3 Scheduled request throughput in the last 5 minutes

This metric shows the scheduled request throughput for the last five minutes in the BPEL service engine.

6.6.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.6.4 Last scheduled request value

This metric shows the value for the last scheduled request in the BPEL service engine.

6.6.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.7 Business Rules Engine Cache Metrics Table

Displays the business rules service engine cache metrics.

6.7.1 Total Access Count since server startup

This metric shows the total access count since server startup.

6.7.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.7.2 Last value of access count

This metric shows the last value of the access count in the business rules service engine cache.

6.7.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.7.3 Total Count since server startup

This metric shows the total count of objects in the cache since server startup.

6.7.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

All Versions	Every 5 Minutes
--------------	-----------------

6.7.4 Last value of total count

This metric shows the last value of the total count.

6.7.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.8 Business Rule Engine Request Breakdown Rate Metric Table

Displays the path of a message/request within the service engine.

6.8.1 Request count since server startup

This metric shows the count of objects in the cache since server startup.

6.8.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.8.2 Throughput of requests in the last 5 minutes

This metric shows the request message throughput in the last five minutes.

6.8.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.8.3 Last value of request

This metric shows the value of the last request message.

6.8.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.8.4 Average request processing time

This metric shows the average request processing time for messages.

6.8.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.8.5 Number of completed requests

This metric shows the number of completed instances.

6.8.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.8.6 Response

Shows if the SOA server is available or not.

6.8.7 UpDown Status

This metric shows the current status of the SOA server. A value of 1 indicates the server is running. A value of 0 indicates the server is down.

6.8.7.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every 60 Samples	=	Not Defined	0	1	The SOA Infrastructure instance is down

6.9 SOA Infrastructure Bindings Rate Metric Table

Shows metrics of the SOA Bindings.

6.9.1 Active request throughput in the last 5 minutes

This metric shows the active request throughput in the last five minutes in the binding component.

6.9.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.9.2 Number of faults since server startup

This metric shows the number of faults in the binding component since server startup.

6.9.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.9.3 Average request processing time

This metric shows the average request processing time for the binding component.

6.9.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.9.4 Total number of processed requests since server startup

This metric shows the total number of processed requests since server startup in the binding component.

6.9.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10 SOA Composite Bindings/References Rate Metric

This category provides information about SOA composite bindings/references rate metrics.

6.10.1 Total number of faults since server start

This metric shows the total number of faults since server startup.

6.10.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.2 Throughput of faults in the last 5 minutes

This metric shows the throughput for faults in the last five minutes.

6.10.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.3 Average Incoming Messages Processing Time

This metric shows the average processing time for incoming messages.

6.10.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.4 Total completed incoming messages since server start

This metric shows total number of completed incoming messages since server startup.

6.10.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.5 Incoming messages throughput in the last 5 minutes

This metric shows the throughput for incoming messages in the last five minutes.

6.10.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.6 Total Incoming Messages Processing Time

This metric shows the total processing time for incoming messages.

6.10.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.7 Total Incoming Message processing time

This metric shows the total processing time for incoming message.

6.10.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.8 Total number of incoming messages since server start

This metric shows the total number of incoming messages since server startup.

6.10.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.9 Incoming message throughput in the last 5 minutes

This metric shows the throughput for incoming messages in the last five minutes.

6.10.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.10 Average Outbound Messages Processing Time

This metric shows the average processing time for outbound messages.

6.10.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.11 Total completed outgoing messages since server start

This metric shows the total number of completed, outgoing messages since server startup.

6.10.11.1 Metric Summary

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

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 5 Minutes
--------------	-----------------

6.10.12 Outbound messages throughput in the last 5 minutes

This metric shows the throughput for outbound messages in the last five minutes.

6.10.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.13 Total Outbound Messages Processing Time since server start

This metric shows the total processing time for outbound messages since server startup.

6.10.13.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.14 Total Outbound Message processing time

This metric shows the total processing time for outbound messages.

6.10.14.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.15 Total number of outgoing messages since server start

This metric shows the total number of outgoing messages since server startup.

6.10.15.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.10.16 Outbound message events throughput in the last 5 minutes

This metric shows the throughput for outbound message events in the last five minutes.

6.10.16.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11 BPEL Dispatcher Rate Metric Table

Displays throughput metrics related to the BPEL dispatcher.

6.11.1 Total number of messages processed per thread since server startup

This metric shows the total number of messages processed per thread in the BPEL dispatcher since server startup.

6.11.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.2 Throughput of total messages in the last 5 minutes

This metric shows the total message throughput in the BPEL dispatcher.

6.11.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.3 Last value of total message

This metric shows the last value of all messages in the BPEL dispatcher.

6.11.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.4 Throughput of average thread allocation time in the last 5 minutes

This metric shows the throughput for the average thread allocation time in the last five minutes in the BPEL dispatcher.

6.11.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.5 Last value of average thread allocation time

This metric shows the last value of average thread allocation time in the BPEL dispatcher.

6.11.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.6 Throughput of average thread lifetime in the last 5 minutes

This metric shows the throughput for the lifetime of the average thread in the last five minutes in the BPEL dispatcher.

6.11.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.7 Last value of average thread lifetime

This metric shows the last value for the lifetime of the average thread in the BPEL dispatcher.

6.11.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.8 Throughput of maximum threads in the last 5 minutes

This metric shows the throughput for maximum threads in the last five minutes.

6.11.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.9 Last value of maximum thread

This metric shows the last value for the maximum thread in the BPEL dispatcher.

6.11.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.10 Throughput of minimum threads in the last 5 minutes

This metric shows the throughput for minimum threads in the last five minutes.

6.11.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.11 Last value of minimum thread

This metric shows the last value for the minimum thread in the BPEL dispatcher.

6.11.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.12 Number of active threads since server startup

This metric shows the number of active threads since server startup in the BPEL dispatcher.

6.11.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.13 Throughput of thread count in the last 5 minutes

This metric shows the thread count throughput in the last five minutes in the BPEL dispatcher.

6.11.13.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.14 Last value of thread count

This metric shows the last thread count value in the BPEL dispatcher.

6.11.14.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.15 Number of thread highs since server startup

This metric shows the number of high thread levels in the BPEL dispatcher since server startup.

6.11.15.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.16 Throughput of thread high in the last 5 minutes

This metric shows high levels of thread throughput in the last five minutes in the BPEL dispatcher.

6.11.16.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.17 Last value of thread high

This metric shows the last high value of the thread in the BPEL dispatcher.

6.11.17.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.18 Number of pending threads since server startup

This metric shows the number of pending threads in the BPEL dispatcher since server startup.

6.11.18.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.19 Throughput of pending threads in the last 5 minutes

This metric shows the throughput for pending threads in the last five minutes in the BPEL dispatcher.

6.11.19.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.20 Last value of pending thread

This metric shows the last value of the pending thread in the BPEL dispatcher.

6.11.20.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.21 Total number of threads since server startup

This metric shows the total number of threads in the BPEL dispatcher since server startup.

6.11.21.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.22 Throughput of total threads in the last 5 minutes

This metric shows the throughput for all threads in the last five minutes in the BPEL dispatcher.

6.11.22.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.11.23 Last value of total thread metric

This metric shows the last value for the total thread metric in the BPEL dispatcher.

6.11.23.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.12 BPMN Dispatcher Rate Metric Table

Throughput of total messages in the last 5 minutes.

6.12.1 Last value of maximum thread

This metric shows the last value for the maximum thread in the BPMN dispatcher

6.12.1.1 Metric Summary

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

6.12.2 Number of active threads since server startup

This metric shows the number of active threads since server startup in the BPMN dispatcher.

6.12.2.1 Metric Summary

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

6.12.3 Throughput of thread count in the last 5 minutes

This metric shows the thread count throughput in the last five minutes in the BPMN dispatcher.

6.12.3.1 Metric Summary

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

6.13 SOA Mediator Engine Cache

This category provides information about SOA Mediator Engine Cache metrics.

6.13.1 Last value of metadata cache

This metric shows the last value of the metadata cache in the mediator service engine.

6.13.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.13.2 Last value of transformation cache

This metric shows the last value of transformation cache in the mediator service engine.

6.13.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.13.3 SOA Mediator Engine Requests

6.13.4 Callback Requests

This metric shows the callback requests in the mediator service engine.

6.13.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.13.5 Delay Requests

This metric shows the delayed requests in the mediator service engine.

6.13.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.13.6 Dequeue Requests

This metric shows the dequeued requests in the mediator service engine.

6.13.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.13.7 Timeout Requests

This metric shows the timed out requests in the mediator service engine.

6.13.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14 SOA Mediator Engine Request Breakdown Rate Metric Table

Displays the path taken by a request in the Mediator Service Engine and the time spent

6.14.1 enqueue-phase.active

This metric shows the active enqueued phases in the mediator service engine.

6.14.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.2 Avg Enqueue Time (ms)

This metric shows the average time spent on enqueueing in the mediator service engine.

6.14.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.3 Enqueued

This metric shows the completed enqueueing phases in the mediator service engine.

6.14.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.4 Throughput of Enqueue phase

This metric shows the throughput for enqueueing in the mediator service engine.

6.14.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.5 Total Enqueue Time (ms)

This metric shows the total enqueueing time in the mediator service engine.

6.14.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.6 invoke-phase.active

This metric shows the active invocations in the mediator service engine.

6.14.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.7 Avg Invoke Time (ms)

This metric shows the average invocation time in the mediator service engine.

6.14.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.8 Invoked

This metric shows the completed invocations in the mediator service engine.

6.14.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.9 Throughput of Invoke phase

This metric shows the throughput for invocations in the mediator service engine.

6.14.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.10 Total Invoke Time (ms)

This metric shows the total invocation time in the mediator service engine.

6.14.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.11 invokeOneway-phase.active

This metric shows the active one-way invocations in the mediator service engine.

6.14.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.12 Avg Oneway Invoke Processing Time (ms)

This metric shows the average processing time for one-way invocations in milliseconds in the mediator service engine.

6.14.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.13 Oneway Invocations

This metric shows the completed one-way invocations in the mediator service engine.

6.14.13.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.14 Throughput of Oneway Invocations

This metric shows the throughput for one-way invocations in the mediator service component.

6.14.14.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.15 Total Oneway Invocation Time (ms)

This metric shows the total time for one-way invocations in the mediator service engine.

6.14.15.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.16 publish-phase.active

This metric shows the active publishing phases in the mediator service engine.

6.14.16.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.17 Avg Publication Time (ms)

This metric shows the average publication times in the mediator service engine.

6.14.17.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.18 Publications

This metric shows the completed publishing phases in the mediator service engine.

6.14.18.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.19 Throughput of Publication phase

This metric shows the throughput for publications in the mediator service engine.

6.14.19.1 Metric Summary

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

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 5 Minutes
--------------	-----------------

6.14.20 Total Publication Time (ms)

This metric shows the total publication times in the mediator service engine.

6.14.20.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.21 transformation-phase.active

This metric shows the active transformation phases in the mediator service engine.

6.14.21.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.22 Avg Transformation Time (ms)

This metric shows the average time for transformations in the mediator service engine.

6.14.22.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.23 Transformations

This metric shows the completed transformations in the mediator service engine.

6.14.23.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.24 Throughput of Transformations

This metric shows the throughput for transformations in the mediator service engine.

6.14.24.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.14.25 Total Transformation Time (ms)

This metric shows the total time for transformations in the mediator service engine.

6.14.25.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15 SOA Infra Mesh

Displays the SOA server mesh metrics

6.15.1 Number of active messages in the mesh

This metric displays the number of active messages in the SOA server.

6.15.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.2 Total number of messages in the mesh since server startup

This metric shows the total number of messages in the SOA server since startup.

6.15.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.3 Number of faulted messages in the mesh

This metric displays the number of faulted messages in the SOA server.

6.15.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.4 Throughput of messages in the mesh in the last 5 minutes

This metric displays the throughput for messages in the SOA server in the last 5 minutes.

6.15.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.5 SOA Infra Mesh Message Count Table

6.15.6 Total number of messages in the mesh since server startup

This metric shows the total number of messages in the service infrastructure since server startup.

6.15.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.7 Mesh Name

This metric shows service infrastructure performance.

6.15.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.8 Throughput of faulted asynchronous messages in the last 5 minutes

This metric shows the throughput for faulted asynchronous messages in the last five minutes in the SOA server.

6.15.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.9 Number of active asynchronous messages

This metric shows the number of active, asynchronous messages in the SOA server at this time.

6.15.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.10 Average processing time of asynchronous messages

This metric shows the average processing time for asynchronous messages in the SOA server.

6.15.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.11 Number of completed asynchronous messages

This metric shows the number of completed asynchronous messages in the SOA server.

6.15.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.12 Throughput of asynchronous messages in the last 5 minutes

This metric shows the throughput for asynchronous messages in the last five minutes in the SOA server.

6.15.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.13 Total processing time of asynchronous messages

This metric shows the total processing time for asynchronous messages in the SOA server.

6.15.13.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.14 Throughput of faulted synchronous messages in the last 5 minutes

This metric shows the throughput for faulted, synchronous messages in the last five minutes in the SOA server.

6.15.14.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.15 Number of active synchronous messages

This metric shows the number of active, synchronous messages in the SOA server.

6.15.15.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.16 Average processing time of synchronous messages

This metric shows the average processing time for synchronous messages in the SOA server.

6.15.16.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.17 Number of completed synchronous messages

This metric displays the number of completed, synchronous messages in the SOA server.

6.15.17.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.18 Throughput of synchronous messages in the last 5 minutes

This metric displays the throughput for synchronous messages in the last five minutes in the SOA server.

6.15.18.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.15.19 Total processing time of synchronous messages

This metric shows the total processing time for synchronous messages in the SOA server.

6.15.19.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16 SOA Message Processing Rate Metric Table

Displays the metrics related to message processing in the service engines

6.16.1 Total number of active requests since server startup

This metric shows the total number of active requests since server startup.

6.16.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.2 Throughput of the active requests in the last 5 minutes

This metric shows message processing details in the service engines.

6.16.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.3 Last value of active request

This metric shows the last value of the active request.

6.16.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.4 Total number of BPEL messages since startup

This metric shows the total number of BPEL messages since startup.

6.16.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.5 Engine Name

This metric shows the BPEL service engine name.

6.16.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.6 Decision Service Engine

6.16.7 Total number of decision service messages since startup

This metric shows the total number of decision service messages since server startup.

6.16.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.8 Engine Name

This metric shows the decision service engine name.

6.16.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.9 Number of faults since server startup

This metric shows the number of faults occurring since server startup.

6.16.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.10 Throughput of faulted requests in the last 5 minutes

This metric shows the throughput of faulted requests in the last five minutes.

6.16.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.11 Rate of processing faulted requests

This metric shows the processing rate for faulted requests.

6.16.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.12 Total number of mediator messages since startup

This metric shows the total number of mediator messages since server startup.

6.16.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.13 Engine Name

This metric shows the mediator service engine name.

6.16.13.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.14 Total number of messages since server startup

This metric shows the total number of messages since server startup.

6.16.14.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.15 Total number of asynchronous requests since server startup

This metric shows the total number of asynchronous requests since server startup.

6.16.15.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.16 Throughput of asynchronous invocations in the last 5 minutes

This metric shows the throughput of asynchronous invocations in the last five minutes.

6.16.16.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.17 Rate of processing asynchronous invocations

This metric shows processing rate for asynchronous invocations.

6.16.17.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.18 Average asynchronous message processing time since server startup

This metric shows the average processing time for asynchronous messages since server startup.

6.16.18.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.19 Total number of synchronous requests since server startup

This metric shows the total number of synchronous requests since server startup.

6.16.19.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.20 Synchronous invocations throughput in the last 5 minutes

This metric shows the throughput for synchronous invocations in the last five minutes.

6.16.20.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.21 Rate of processing synchronous invocations

This metric shows the processing rate for synchronous invocations.

6.16.21.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.22 Average synchronous message processing time since server startup

This metric shows the average processing time for synchronous messages since server startup.

6.16.22.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.16.23 Workflow Service Engine**6.16.24 Total number of workflow messages since startup**

This metric shows the total number of human workflow messages since server startup.

6.16.24.1 Metric Summary

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

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 5 Minutes
--------------	-----------------

6.16.25 Engine Name

This metric shows the human workflow service engine name.

6.16.25.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.17 Total Composite Instances

Shows the instance metrics of composites deployed on the SOA server

6.17.1 Total number of instances since server startup

This metric shows the total number of composite instances created since server startup. This count spans across composites deployed on the SOA server.

6.17.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.17.2 Domain Name

This metric shows the instance metrics of composites deployed on the SOA server.

6.17.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.18 Total Composite Faults

Displays the metrics of the composite faults

6.18.1 Total number of faults since server startup

This metric shows the total number of faults across all composite instances since server startup.

6.18.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 15 Minutes

6.19 Workflow Request Breakdown Rate Metrics

Shows the various services used by the Workflow Service Engine and their usage metrics.

6.19.1 Number of invocations since server startup

This metric shows the total number of invocations in the human workflow service engine since server startup.

6.19.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.19.2 Service Name

This metric shows the name of the internal service used by the human workflow service engine.

6.19.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.19.3 Number of active requests

This metric shows the number of active requests in the human workflow service engine.

6.19.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.19.4 Average request processing time

This metric shows the average request processing time in the human workflow service engine.

6.19.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

6.19.5 Number of completed requests

This metric shows the total number of completed requests in the human workflow service engine.

6.19.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 5 Minutes

This chapter describes Oracle SOA Composite metrics.

7.1 SOA Composite B2B Inbound Metrics

Displays the metrics for the Composite B2B Inbound messages.

7.1.1 InBound Message Count

This metric displays the number of inbound messages in B2B trading partners.

7.1.2 Inbound Messages Faults

This metric displays the number of faults in inbound messages in the B2B binding.

7.1.3 Inbound Messages Size

This metric displays the inbound message size in the B2B binding.

7.1.4 Inbound Message Processing Time

This metric displays the time taken to process inbound messages by the B2B binding.

7.1.5 ServerName

This metric shows the name of the server for inbound messages in B2B trading partners.

7.1.6 Total Outbound Messages Count

This metric displays the total number of outbound messages in the B2B binding.

7.1.7 SOA Composite B2B Outbound Metrics

Displays the B2B outbound message metrics.

7.1.8 Name

7.1.9 OutBound Message Count

This metric displays the number of outbound messages in the B2B binding.

7.1.10 Outbound Messages Faults

This metric displays the number of faults in outbound messages in the B2B binding.

7.1.11 Outbound Messages Size

This metric displays the size of outbound messages in the B2B binding.

7.1.12 Outbound Message Processing Time

This metric displays the processing time for outbound messages in the B2B binding.

7.1.13 ServerName

This metric shows the name of the server for B2B outbound messages.

7.1.14 Total Outbound Messages Count

This metric displays the total number of outbound messages in the B2B binding.

7.2 BPEL Activity Time Distribution

Shows the information related to the BPEL activity-time distribution.

7.2.1 Average Execution Time (ms)

This metric shows the average time it took for this BPEL activity to execute.

7.2.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.2.2 Activity Count

This metric shows the number of times this BPEL activity has been invoked.

7.2.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.2.3 Activity Name

This metric shows the name of the BPEL activity whose data is being shown.

7.2.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.2.4 Server Name

This metric shows the name of the J2EE container in which the SOA composite application is deployed.

7.2.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.3 BPMN Activity Time Distribution

Displays the metrics related to the BPMN Activity-Time distribution

7.3.1 Average Execution Time (ms)

The average time that this BPMN activity took to execute.

7.3.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.3.2 Activity Count

Number of times this BPMN activity has been invoked.

7.3.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.3.3 Activity Name

Name of the BPMN Activity whose data are being shown.

7.3.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.3.4 Server Name

Name of the J2EE container where the SOA Composite application is deployed

7.3.4.1 Metric Summary

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

7.4 SOA Composite - Response Metrics

Indicates whether the SOA composite application is available to accept requests.

7.4.1 ServerName

This metric shows the name of the server on which the SOA composite application is deployed.

7.4.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.4.2 Composite Status

This metric shows the current state of the SOA composite application. A value of 1 indicates the application is available and a value of 0 indicates the application is not available.

7.4.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every 60 Samples	=	Not Defined	0	1	The J2EE Application is down

7.5 SOA Composite Bindings/References Rate Metric

Displays metrics about the SOA bindings.

7.5.1 Total number of faults since server start

This metric shows the total number of faulted messages in the binding.

7.5.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.5.2 Fault throughput in last 5 minutes

This metric shows the throughput for faulted messages in the binding.

7.5.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.5.3 Average Incoming Messages Processing Time

This metric shows the average processing time of incoming messages in the binding.

7.5.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.5.4 Total Incoming Messages Processing Time

This metric shows the total processing time of incoming messages in the binding.

7.5.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.5.5 Total Incoming Message processing time

This metric shows the total processing time of incoming messages in the binding.

7.5.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.5.6 Total number of incoming messages since server start

This metric shows the total number of incoming messages in the binding.

7.5.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.5.7 Incoming message throughput in last 5 minutes

This metric shows the throughput for inbound messages in the binding.

7.5.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.5.8 Average Outbound Messages Processing Time

This metric shows the average processing time of outbound messages in the binding.

7.5.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.5.9 Total Outbound Messages Processing Time since server start

This metric shows the total processing time of outbound messages in the binding.

7.5.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.5.10 Total Outbound Message processing time

This metric shows the total processing time of outbound messages in the binding.

7.5.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.5.11 Total number of outgoing messages since server start

This metric shows the total number of outbound messages in the binding.

7.5.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.5.12 Outbound message throughput in last 5 minutes

This metric shows the throughput for outbound messages in the binding.

7.5.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.6 SOA Component References

Shows the metrics of the SOA component references.

7.6.1 Average Message Processing Time (secs) since server start up

This metric shows the average time taken by the reference to respond to invocations.

7.6.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.6.2 Number of Faults

This metric shows the number of faults encountered when this reference was invoked.

7.6.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.6.3 Number of Messages

This metric shows the number of times that this reference has been invoked by the SOA component.

7.6.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

All Versions	Every Minute
--------------	--------------

7.6.4 Average Message Processing Time (secs) in last 5 mins

This metric shows the total time taken by the reference to respond to an invocation.

7.6.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.6.5 Reference Name

This metric shows the SOA component whose reference is being shown.

7.6.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.6.6 Component References

This metric shows the SOA component reference name.

7.6.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.6.7 Component Type

This metric indicates that the current type is a component whose references are being listed.

7.6.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7 SOA Component Rollup Metric

Shows all SOA component throughput metrics.

7.7.1 Business faults throughput in the last 5 minutes

This metric shows the throughput for messages with business faults in this SOA component.

7.7.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.2 Number of active faulted instances

This metric shows the number of active, faulted messages in this SOA component.

7.7.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.3 Average processing time for faulted instances

This metric shows the average time spent processing faulted messages in this SOA component.

7.7.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.4 Number of faulted instances

This metric shows the number of faulted messages in this SOA component.

7.7.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.5 Throughput of faulted instances in the last 5 minutes

This metric shows the throughput for faulted messages in this SOA component.

7.7.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.6 Total time spent on processing faulted instances

This metric shows the time spent processing faulted messages in this SOA component.

7.7.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.7 Total average faulted processing time

This metric shows the total time spent processing faulted messages in this SOA component.

7.7.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.8 Instance throughput in the last 5 minutes

This metric shows the throughput for incoming messages in the SOA component.

7.7.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.9 Server Name

This metric shows the SOA component throughput for the server.

7.7.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.10 Number of active successful instances

This metric shows the number of successful, active messages in this SOA component.

7.7.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.11 Average processing time for successful instances

This metric shows the average time spent processing successful messages in this SOA component.

7.7.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.12 Number of successful instances

This metric shows the number of successfully completed messages in this SOA component.

7.7.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.13 Throughput of successful instances in the last 5 minutes

This metric shows the throughput for successful messages in this SOA component.

7.7.13.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.14 Total time spent on processing successful instances

This metric shows the time spent processing successful messages in this SOA component.

7.7.14.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.15 Total average successful processing time

This metric shows the total time spent processing successful messages.

7.7.15.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.16 System faults throughput in the last 5 minutes

This metric shows the throughput for messages with system faults in this SOA component.

7.7.16.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.7.17 Total faults throughput in the last 5 minutes

This metric shows the throughput for all faulted messages in this SOA component.

7.7.17.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.8 SOA Composite - Rate Metrics

Displays the throughput metrics for a SOA composite.

7.8.1 Instance throughput in the last 5 minutes

This metric shows the total throughput for the composite instances.

7.8.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.8.2 Asynchronous message event throughput in the last 5 minutes

This metric shows the throughput for asynchronous messages.

7.8.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.8.3 Average asynchronous message Processing time

This metric shows the average processing time of asynchronous messages in the SOA composite.

7.8.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.8.4 Asynchronous messages throughput in the last 5 minutes

This metric shows the throughput of asynchronous messages in this SOA composite.

7.8.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.8.5 Total asynchronous message processing time

This metric shows the total processing time of asynchronous messages in the SOA composite.

7.8.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.8.6 Synchronous message event throughput in the last 5 minutes

This metric displays the throughput for requests to the SOA composite.

7.8.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.8.7 Average synchronous message processing time

This metric shows the average processing time of synchronous messages in this SOA composite.

7.8.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.8.8 Synchronous messages throughput in the last 5 minutes

This metric shows the throughput for all synchronous messages to this composite.

7.8.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.8.9 Total synchronous message processing time

This metric shows the total processing time of synchronous messages for this SOA composite.

7.8.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.8.10 Server Name

This metric shows the name of the server on which the SOA composite application revision is deployed.

7.8.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9 Mediator Case

Displays metrics about the mediator case.

7.9.1 Average processing time of invocations

This metric shows the average processing time of invocations in the mediator component.

7.9.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.2 Invocation throughput in the last 5 minutes

This metric shows the throughput for messages in the mediator case.

7.9.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.3 Total processing time for invocations

This metric shows the total processing time of invocations in the mediator component.

7.9.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.4 Failed filtered messages throughput in the last 5 minutes

This metric shows the throughput for the failed, filtered message in the mediator case.

7.9.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.5 Average processing time of faulted cases

This metric shows the average processing time of faulted cases in the mediator component.

7.9.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.6 Number of faulted cases

This metric shows the number of faulted cases in this mediator component.

7.9.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.7 Faulted case throughput in the last 5 minutes

This metric shows the throughput of faulted cases in this mediator component.

7.9.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.8 Total processing time for faulted cases

This metric shows the total processing time for faulted cases in the mediator component.

7.9.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.9 Filtered messages throughput in the last 5 minutes

This metric shows the throughput for the filtered messages of the mediator case.

7.9.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.10 Invocation Count throughput in the last 5 minutes

This metric shows the throughput for the invocation count of the mediator case.

7.9.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.11 Component Type

This metric shows the component type, which in this case is always mediator.

7.9.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.12 Composite Label

This metric shows the label of the SOA composite.

7.9.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.13 Case Target

This metric shows the case target of the mediator.

7.9.13.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.14 Mediator Flow

This metric shows the state of the mediator flow.

7.9.14.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.15 Flow Type

This metric shows the flow type of the mediator component case.

7.9.15.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.16 Average processing time of successful cases

This metric shows the average processing time of successful cases.

7.9.16.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.17 Number of successful cases

This metric shows the number of successful cases in this mediator component.

7.9.17.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.18 Successful case throughput in the last 5 minutes

This metric shows the throughput for successful cases in this mediator component.

7.9.18.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.9.19 Total processing time for successful cases

This metric shows the total processing time for successful cases in this mediator component.

7.9.19.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10 Mediator Flow Metric

Displays metrics about the Mediator component's flow.

7.10.1 Deferred message count throughput in last 5 minutes

This metric shows the throughput for the deferred message count in the mediator component.

7.10.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10.2 Failed filter case throughput in last 5 minutes

This metric shows the throughput for the failed filter case in the mediator component.

7.10.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10.3 Average faulted instance processing time

This metric shows the average time spent processing faulted messages.

7.10.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10.4 Faulted instances throughput in last 5 minutes

This metric shows the throughput for faulted messages.

7.10.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10.5 Total faulted instance processing time

This metric shows the total time spent processing faulted messages.

7.10.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10.6 Instance throughput in last 5 minutes

This metric shows the throughput for the mediator instances.

7.10.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10.7 Component Type

This metric shows the component type, which in this case is always mediator.

7.10.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10.8 Composite Label

This metric shows the SOA composite label in which the mediator component is packaged.

7.10.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10.9 Flow Type

This metric shows the flow type for the mediator.

7.10.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10.10 Average successful instance processing time

This metric shows the average time spent processing successful messages.

7.10.10.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10.11 Successful instances throughput in last 5 minutes

This metric shows the successful instance processing time.

7.10.11.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.10.12 Total successful instance processing time

This metric shows the total time spent processing successful messages.

7.10.12.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every Minute

7.11 Total Composite Instances

Displays metrics about the total composite instances.

7.12 Total Composite Faults

Displays metrics about total composite faults.

This chapter describes Oracle Forms metrics.

8.1 Forms Resource Usage

This metric measures the resources consumed by all the Forms sessions.

8.1.1 CPU Usage (%)

This metric specifies the percentage CPU usage of all the Forms sessions.

8.1.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 60 Hours	After Every Sample	>=	80	90	2	cpu.component is %value% and has crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold.

8.1.1.2 User Action

Navigate to the host Home page in the Fusion Middleware Control and review the top processes to see if you can identify which processes are consuming the most CPU. Or, navigate to the host Performance page in the Fusion Middleware Control and view the processes that are consuming the most CPU and take appropriate action.

8.1.2 Total CPU Time (millisec)

This metric specifies the total CPU Time (in millisec) consumed by all the Forms sessions.

8.1.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 60 Hours

8.1.2.2 User Action

No user action required.

8.1.3 instname.value

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

Target Version	Collection Frequency
All Versions	Every 60 Hours

8.1.4 Total Private Memory (KB)

This metric measures the total memory (in KB) consumed by all the Forms sessions. This is not the actual private memory on Linux but the Resident Set Size (RSS). RSS is the number of pages the process has in physical memory, minus three for administrative purposes.

8.1.4.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 60 Hours	After Every Sample	>=	80	90	2	totalprivateMem is %value% and has crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold.

8.1.4.2 User Action

Navigate to the host Home page in the Fusion Middleware Control and review the top processes to see if you can identify which processes are consuming the most Memory.

Or, navigate to the host Performance page in the Fusion Middleware Control and view the processes that are consuming the most Memory and take appropriate action.

8.2 Forms Application Deployments

Shows information about the forms application deployments deployed in various WebLogic Server instances.

8.2.1 Forms Application Context Root

This metric specifies the Context Root of the Forms application deployment.

8.2.1.1 User Action

No action required.

8.2.2 instname.value

8.2.3 Forms Application Internal Name

This metric provides the Forms application name including the version information.

8.2.3.1 User Action

No user action required.

8.2.4 Number of User Sessions

This metric indicates the number of forms sessions running per Forms application deployment.

8.2.4.1 User Action

No user action required.

8.2.5 Application State

This metric describes the current state of an application. An application could be in various states such as Active, Prepared and so on.

8.2.5.1 User Action

No user action required.

8.2.6 Application status

This metric specifies the status of Forms application deployment.

8.2.6.1 User Action

Check whether the WebLogic Server instance has been started and if so whether the application formsapp.ear has been deployed successfully.

8.2.7 Forms Application Version

This metric specifies the Forms Application Version.

8.2.7.1 User Action

No user action required.

8.3 FormsUserSessions

This metric provides information such as resource usage, connection time, and so on of individual Forms sessions.

8.3.1 appname.value

This metric specifies the name of the forms application that is deployed in the WebLogic Server instance. This is the deployment name in whose servlet the forms session is running.

8.3.1.1 User Action

No user action required.

8.3.2 config

This metric specifies the Configuration section the Forms Session is using. The configuration parameters are picked up either from the default section or the selected configuration section of formsweb.cfg.

8.3.2.1 User Action

No user action required.

8.3.3 connect.value

This metric provide information on the client connection time. This is the time at which the client connected to the Forms Runtime process.

8.3.3.1 User Action

No user action required.

8.3.4 CPU Usage (%)

This metric specifies the percentage of CPU Usage consumed by individual Forms sessions.

8.3.4.1 User Action

Navigate to the host Home page in the Fusion Middleware Control and review the top processes to see if you can identify which processes are consuming the most CPU. Or, navigate to the host Performance page in the Fusion Middleware Control and view the processes that are consuming the most CPU and take appropriate action.

8.3.5 CPU Time (millisec)

This metric specifies the CPU Time (in millisec) consumed by individual Forms session.

8.3.5.1 User Action

No user action required.

8.3.6 dbname.value

This metric provides the Database name to which the Forms session is connected.

8.3.6.1 User Action

No user action required.

8.3.7 executable.value

This metric provides the Forms Runtime Engine executable name.

8.3.7.1 User Action

No user action required.

8.3.8 form.value

This metric specifies the name of the form module that is active.

8.3.8.1 User Action

No user action required.

8.3.9 instname.value

This metric provides the name of the application server instance where Forms is configured.

8.3.9.1 User Action

No user action required.

8.3.10 ip.value

This metric specifies the IP Address of the client machine.

8.3.10.1 User Action

No user action required.

8.3.11 jsessionid.value

This metric provides the Jsession ID that is used by the Forms Servlet for session management.

8.3.11.1 User Action

No user action required.

8.3.12 Private Memory (KB)

This metric specifies the Private memory consumed by the Forms session. This is not the actual private memory on Linux but the Resident Set Size (RSS). RSS is the number of pages the process has in physical memory, minus three for administrative purposes.

8.3.12.1 User Action

Navigate to the host Home page in the Fusion Middleware Control and review the top processes to see if you can identify which processes are consuming the most Memory. Or, navigate to the host Performance page in the Fusion Middleware Control and view the processes that are consuming the most Memory and take appropriate action.

8.3.13 servername.value

This metric provides the name of the WebLogic Server where Forms servlet is deployed.

8.3.13.1 User Action

No user action required.

8.3.14 threadid.value

This metric provides the Thread ID of the Forms process that is used for tracing on Windows platforms.

8.3.14.1 User Action

No user action required.

8.3.15 tnshentry.value

This metric provides the TNS alias that is used by a running form to connect to Database. This is used to get Database specific information to be displayed on the Database Sessions page.

8.3.15.1 User Action

No user action required.

8.3.16 tracefile.value

This metric provides the name of the Trace file in which the trace statements are written.

8.3.16.1 User Action

No user action required.

8.3.17 tracing.value

This metric provides information on the trace group that contains the trace events that are being traced. This may not reflect the accurate trace group if dynamic tracing built-ins have been used to enable or disable selective events.

8.3.17.1 User Action

No user action required.

8.3.18 user.value

This metric specifies the Database user that is connected to the Forms session.

8.3.18.1 User Action

No user action required.

8.4 JVM Controllers

The JVM Controllers category provides information about all the JVM controllers.

8.4.1 CPU Usage (%)

This metric specifies the percentage of CPU Usage consumed by the JVM Controller including child JVM processes.

8.4.1.1 User Action

Navigate to the host Home page in the Fusion Middleware Control and review the top processes to see if you can identify which processes are consuming the most CPU. Or, navigate to the host Performance page in the Fusion Middleware Control and view the processes that are consuming the most CPU.

8.4.2 CPU Time (millisec)

This metric specifies the CPU Time (in millisec) consumed by the JVM Controller process including child JVM processes.

8.4.3 Number of JVMs running for this JVM Controller

This metric specifies the number of JVMs running for this JVM Controller.

8.4.3.1 User Action

No user action required.

8.4.4 Private Memory (KB)

This metric specifies the Private Memory (in KB) consumed by the JVM Controller including child JVM processes. This is not the actual private memory on Linux but the Resident Set Size (RSS). RSS is the number of pages the process has in physical memory, minus three for administrative purposes.

8.4.5 UP Time (millisec)

This metric specifies the UP Time (in millisec) of the JVM Controller.

8.4.5.1 User Action

No user action required.

8.5 Number Of Sessions

The number of sessions category provides information about all the sessions.

8.5.1 Number Of Sessions

This metric measures the total number of Forms sessions of all Forms application deployments deployed in various WebLogic Server instances.

8.5.1.1 User Action

No user action required.

8.6 Response

The Response category provides information about the default Forms application deployment 'formsapp' in WebLogic Server WLS_FORMS

8.6.1 instname.value

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

Target Version	Collection Frequency
All Versions	Every 60 Hours

8.6.2 Status

This metric shows the status of the Forms Application 'formsapp' running inside WebLogic Server WLS_FORMS. By default, this metric has a critical threshold of 0 indicating that the Forms Servlet is unreachable and therefore Forms instance is not accessible. A critical alert is generated when the metric value reaches the critical threshold value once. It is recommended that this threshold is set at the default value.

8.6.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 60 Hours	After Every Sample	=	Not Defined	0	1	The Forms instance is not accessible.

The value of this metric is obtained by measuring the availability of the Forms application.

The current value of this metric can also be seen from the Deployments table on the Forms Home Page.

8.6.2.2 User Action

Check whether the WLS_FORMS instance has been started and if so that the application formsapp.ear has been deployed successfully.

8.7 Servlet ResponseTime

Average response time (in millisec) of Forms Servlet.

8.7.1 instname.value

This metric provides the name of the instance where Forms Servlet is configured.

8.7.2 Average Response Time (millisec)

This metric measure the average response time.

8.7.3 instname.value

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

Target Version	Collection Frequency
All Versions	Every 60 Hours

8.7.4 Servlet Name

This metric specifies the name of the servlet.

8.7.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 60 Hours

8.7.4.2 User Action

No user action required.

8.7.5 Response Time (millisec)

This metric measures the average time per task taken by the servlet to serve requests. By default, this metric has a warning threshold of 500 millisec and critical threshold of 1000 millisec indicating that the Forms Listener Response time is unacceptable. A critical alert is generated when the metric value reaches the critical threshold value once. It is recommended that this threshold is left at the default value

8.7.5.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 60 Hours	After Every Sample	>	500	1000	1	The Forms Listener response time is unacceptable.

8.7.5.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Application Deployment Name" and "Server Name" objects.

If warning or critical threshold values are currently set for any unique combination of "Application Deployment Name" and "Server 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 "Application Deployment Name" and "Server Name" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

8.7.5.3 User Action

No user action required.

8.7.6 Number Of Tasks

This metric specifies the total number of requests served by Forms Servlet.

8.7.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 60 Hours

8.7.6.2 User Action

No user action required.

8.7.7 Total Time

This metric specifies the total time spent (in millisecond) on serving completed requests by Forms Servlet.

8.7.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 60 Hours

8.7.7.2 User Action

No user action required.

8.7.8 Web Module Name

This metric specifies the name of the Web Module.

8.7.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 60 Hours

8.7.8.2 User Action

No user action required.

This chapter describes Oracle Portal metrics.

9.1 Portlets (All)

This category provides information about portlets metrics.

9.1.1 Cache Hit (%)

This metric specifies the rate of cache hits.

9.1.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.1.2 Average Time (ms)

This metric specifies the average time.

9.1.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.1.3 Requests (per minute)

This metric specifies the rate per minute of requests to the Portal.

9.1.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.1.4 Availability (%)

9.1.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.1.5 Producer Type

9.1.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.2 Portlets (Top 5)

This category provides information about portlets (Top 5) metrics.

9.2.1 Cache Hit (%)

9.2.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.2.2 Average Time (ms)

9.2.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.2.3 Requests (per minute)

9.2.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.2.4 Availability (%)

9.2.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.2.5 Producer Type

9.2.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.3 Producers (All)

This category provides information about producers metrics.

9.3.1 Cache Hit (%)

This metric specifies the rate of cache hits from a producer.

9.3.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.3.1.2 User Action

Use this metric to assess the cache hit ratio for a producer.

9.3.2 Average Time (ms)

This metric specifies the average time in executing portlet calls for a producer.

9.3.2.1 Metric Summary

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

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 300 Hours
--------------	-----------------

9.3.2.2 User Action

Use this metric to assess the response times of a producer. If response time is out-of-bound, then assess average response times of individual portlets. Consider caching options and scaling out to optimize performance.

9.3.3 Requests (per minute)

This metric specifies the rate of requests per minute that are getting serviced from a producer.

9.3.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.3.3.2 User Action

Use this metric to assess the load on the producer.

9.3.4 Availability (%)

This metric measures the availability of a producer, and measures the percent of requests that got an HTTP-2xx response from a producer.

9.3.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.3.4.2 User Action

Use this metric to assess the availability of a producer. If this number is out-of-bounds, check the producer instance for availability and check for network connectivity from the Portal middle-tier to the producer.

9.3.5 Producer Type

9.3.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.4 Producers (Top 5)

This category provides information about producers (Top 5) metrics.

9.4.1 Cache Hit (%)

9.4.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.4.2 Average Time (ms)

9.4.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.4.3 Requests (per minute)

9.4.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.4.4 Availability (%)

9.4.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.4.5 Producer Type

9.4.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.5 Portal Cache

This category provides information about portal cache metrics.

9.5.1 Average Expire Hit Time (ms)

This metric specifies the average time for cache hits in the Portal cache for content that uses "Expires-based caching".

9.5.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.5.1.2 User Action

Use this metric to assess the caching statistics of the pages/content. See "Using Caching with PL/SQL Based Web Applications" section in the Oracle® Fusion Middleware User's Guide for mod_plsql in OTN.

9.5.2 Expire Hit (%)

This metric specifies the percentage of requests that resulted in successful cache hits, for content that does "Expiry-based caching".

9.5.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.5.2.2 User Action

Use this metric to assess the caching statistics of your pages/content. See "Using Caching with PL/SQL Based Web Applications" section in the Oracle® Fusion Middleware User's Guide for mod_plsql in OTN.

9.5.3 Average New Time (ms)

This metric specifies the average time for cache misses in the Portal cache for content that is newly created in the cache.

9.5.3.1 Metric Summary

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

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 300 Hours
--------------	-----------------

9.5.3.2 User Action

Use this metric to assess the caching statistics of your pages/content. See "Using Caching with PL/SQL Based Web Applications" section in the Oracle® Fusion Middleware User's Guide for mod_plsql in OTN.

9.5.4 New (%)

This metric specifies the percentage of requests for which new content was generated and cached. New content was generated because there were cache misses due to content not being in the Portal Cache.

9.5.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.5.4.2 User Action

Use this metric to assess the caching statistics of your pages/content.

9.5.5 Average Ping Hit Time (ms)

This metric specifies the average time for cache hits in the Portal cache for content that uses "Ping-based caching".

9.5.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.5.5.2 User Action

Use this metric to assess the caching statistics of the pages/content. See "Using Caching with PL/SQL Based Web Applications" section in the Oracle® Fusion Middleware User's Guide for mod_plsql in OTN.

9.5.6 Ping Hit (%)

This metric specifies the percentage of requests that resulted in successful cache hits for content that requires ping checks.

9.5.6.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.5.6.2 User Action

Use this metric to assess the caching statistics of your pages/content.

9.5.7 Requests (per minute)

This metric specifies the rate per minute of requests to the Portal Cache.

9.5.7.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.5.7.2 User Action

Use this metric to assess the rate of requests to the Portal Cache.

9.5.8 Average Stale Time (ms)

This metric specifies the average time for cache misses in the Portal cache for content that is stale and requires re-generation.

9.5.8.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.5.8.2 User Action

Use this metric to assess the caching statistics of the pages/content. See "Using Caching with PL/SQL Based Web Applications" section in the Oracle® Fusion Middleware User's Guide for mod_plsql in OTN.

9.5.9 Stale (%)

This metric specifies the percentage of requests that resulted in cache misses because the content cached in the Portal Cache was not the latest.

9.5.9.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.5.9.2 User Action

Use this metric to assess the caching statistics of your pages or content and to understand how frequently cached content is becoming stale due to content editing. See "Using Caching with PL/SQL Based Web Applications" section in the Oracle® Fusion Middleware User's Guide for mod_plsql in OTN.

9.6 Database Connection Pool Statistics

This category provides information about database connection pool statistics.

9.6.1 Average Fetch Time (ms)

This metric provides the average time spent by Portal in acquiring a database connection - either by creating a fresh connection to the database or by acquiring a connection from the connection pool.

9.6.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.6.1.2 User Action

If the average time is high:

- Look at the connection pool hit ratio and make sure it is not out-of-bounds.
- Look at the time taken to create new connections to the database, and at your database settings.
- Check the network and database connection.

9.6.2 Requests (per minute)

This metric provides the rate of incoming requests that require a database connection

9.6.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.6.2.2 User Action

Tune the database sessions based on the number of concurrent requests that require a database connection.

9.6.3 Hit (%)

This metric indicates the effectiveness of the Portal connection pool by measuring the percentage of Portal database connection pool hit rate.

9.6.3.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.6.3.2 User Action

If this value is low, check the:

- setting for the `PlsqlMaxRequestsPerSession` connection pooling parameter. If the setting for this parameter is too low, the pooled connection may be discarded based on the property setting.
- "Database Repository Response Code Statistics" metrics to see if there are too many error responses. Some database error responses may cause a pooled connection to be discarded.
- database logs to find out if there are any server side crashes.

9.6.4 New (%)

This metric indicates the effectiveness of the Portal connection pool by measuring the percentage rate of creating new database connections in Portal.

9.6.4.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.6.4.2 User Action

If this value is low, check the:

- setting for the `PlsqlMaxRequestsPerSession` connection pooling parameter. If the setting for this parameter is too low, the pooled connection may be discarded based on the property setting.
- "Database Repository Response Code Statistics" metrics to see if there are too many error responses. Some database error responses may cause a pooled connection to be discarded.
- database logs to find out if there are any server side crashes.

9.6.5 Stale (%)

This metric indicates the effectiveness of the Portal connection pool by providing the percentage rate of discarding pooled database connections.

9.6.5.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.6.5.2 User Action

If this value is low, check the:

- setting for the `PlsqlMaxRequestsPerSession` connection pooling parameter. If the setting for this parameter is too low, the pooled connection may be discarded based on the property setting.

- "Database Repository Response Code Statistics" metrics to see if there are too many error responses. Some database error responses may cause a pooled connection to be discarded.
- database logs to find out if there are any server side crashes.

9.7 Database Repository Response Code Statistics

This category provides information about database repository response code statistic metrics.

9.7.1 Responses (per minute)

This metric provides the rate per minute for generating a HTTP response code, and is a measure of how frequently a particular HTTP response code is generated.

9.7.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.7.1.2 User Action

Use this value along with the "Average Response Time" value to assess the number of requests that are generating a particular HTTP response code and to assess the total time spent on it. For example, charting some of the typical failure codes like HTTP-404, HTTP-500, HTTP-503 and so on enables the assessment of the failure rate.

9.7.2 Average Processing Time (ms)

This metric provides the average response time for generating a response code.

9.7.2.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.7.2.2 User Action

Use this value along with the "Responses (per minute)" value to assess the time taken in generating a response. If the average response time is not within bounds, or if the total time taken in generating a response code is high, investigate further to find out why. For example, if the time taken to get a response from the database is consistently high, then it is possible that the database is under too much load or it is not performing too well because of some SQL*queries. SQL*Profiling and SQL*Trace maybe enabled to help identify the issue.

9.8 Page Engine Statistics

This category provides information about page engine statistic metrics.

9.8.1 Average Queue Wait Time (ms)

This metric provides the average time all internal PPE requests spent in the PPE internal request queue.

9.8.1.1 Metric Summary

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

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.1.2 User Action

If the average time is high, analyze why existing PPE threads are not being released to take on work from the PPE internal queue. Assess the page caching options and analyze the performance of system and network resources. Consider tweaking the number of the PPE fetcher threads or having a new container. Make sure that increasing the number of threads does not impact performance.

9.8.2 Average Page Processing Time (ms)

This metric specifies the average time taken to generate the pages and also retrieve the page metadata.

9.8.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.2.2 User Action

If the average time is high, use the Portal diagnostic logs to find out the cause of any failures. Average time may be high due to any of the following reasons:

- Poorly performing portlet
- Sudden increase in the load
- Database performance issues
- Network or database connectivity issues

9.8.3 Pages (per minute)

This metric specifies the rate of pages per minute generated by the Parallel Page Engine

9.8.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.3.2 User Action

If this value is out-of-bounds, identify the key pages in the site and enable better caching options. See "Improving Page Performance" chapter in "User's Guide for Oracle Portal" document in OTN for performance tuning information.

9.8.4 Average Page Metadata Fetch Time (ms)

This metric specifies the average time taken to retrieve the page metadata.

9.8.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.4.2 User Action

If the average time is high, review the page metadata caching options for the pages and enable more caching at various levels.

9.8.5 Page Metadata Fetches (per minute)

This metric specifies the page metadata throughput per-minute.

9.8.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.5.2 User Action

If this value is out-of-bounds, identify the key pages for the site and enable better caching. See "Improving Page Performance" chapter in "User's Guide for Oracle Portal" document in OTN for performance tuning information.

9.8.6 Average Page Metadata Wait Time (ms)

This metric specifies the average time spent in the PPE internal request queue waiting for page metadata.

9.8.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.6.2 User Action

This value maybe high because the system is under heavy load. The value may also be high because the requests are waiting for the PPE request threads to become available.

In this case, use the "poolSize" configuration setting to increase the size of the thread pool. Default pool size is 25.

9.8.7 Queue Timeouts (per minute)

This metric specifies the rate per minute of the number of requests for Portal data that have timed out in the PPE internal request queue.

9.8.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.7.2 User Action

If this value is out-of-bounds, use the diagnostic logs to get details on specific page or content requests that are timing out. Analyze the log details to identify any pattern for the timeout. Some common reasons for timeout are listed below.

- Timeout is encountered around peak load times. This may be because the system is not setup to take the load, the pages/content are not responsive enough to meet the performance criteria, or the pages/contents are not optimized for best performance by enabling caching.
- Timeout is always for a particular portlet or set of portlets. Evaluate the backend producer and its load and response characteristics.
- Same page times out most of the times. This typically is indicative of some attributes of the page. Analyze the page and its contents, and optimize caching. See "Improving Page Performance" chapter in "User's Guide for Oracle Portal" document in OTN for performance tuning information.
- Timeouts may occur because of performance blocks in resources like the CPU/memory/resources of the instances such as the middle-tier, database, producer nodes and so on. Monitor these resources for such blocks.
- Timeout may occur while connecting to the remote nodes or there maybe delays due to network connectivity issues.
- Timeout may occur due to improper system and network configuration.

If all aspects of the system and network are setup properly, consider increasing the number of fetcher threads in the PPE or adding another node in the cluster. However, this option should be done only if other components can take on the load added by the additional fetcher threads. If the backend producer cannot handle the existing load, then adding more fetcher threads or cluster members will negatively impact performance. In such cases, consider adding more resources for the producers.

9.8.8 HTTP200-HTTP299 (%)

This metric provides the percentage rate of HTTP-2xx responses generated by the Portal Parallel Page Engine (PPE).

9.8.8.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.8.2 User Action

If this number is low, use the diagnostic logs to investigate.

9.8.9 HTTP300-HTTP399 (%)

This metric provides the percentage rate of HTTP-3xx responses generated by the Portal Parallel Page Engine (PPE).

9.8.9.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.9.2 User Action

If the number of redirect responses in the system (HTTP 302) is high, review the cause for these responses and see if something can be optimized.

9.8.10 HTTP400-HTTP499 (%)

This metric provides the percentage rate of HTTP-4xx responses generated by the Portal Parallel Page Engine (PPE).

HTTP-499 is a special code which means that the requested resource is protected and a "Login" is needed to access it. Similarly, "HTTP-470" means that a "Logout" operation was performed.

9.8.10.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.10.2 User Action

If the number of HTTP-4xx responses that are generated by the PPE is high, use the Portal diagnostic logs to investigate.

9.8.11 HTTP500-HTTP599 (%)

This metric provides the percentage rate of HTTP-5xx responses generated by the Portal Parallel Page Engine (PPE).

9.8.11.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 300 Hours
--------------	-----------------

9.8.11.2 User Action

If the number of HTTP-5xx responses that are generated by the PPE is high, use the Portal diagnostic logs to investigate.

9.8.12 Timeout (%)

This metric provides the percentage rate of timeout responses generated by the Portal Parallel Page Engine (PPE).

9.8.12.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.12.2 User Action

If the number of timeout responses is high, use the Portal diagnostic logs to find out the cause of any failures. Timeouts may occur because of any of the following reasons:

- Poorly performing portlet
- Sudden increase in the load
- Database performance issues
- Network or database connectivity issues
- Timeout value set to an improper value

9.8.13 WebCache Page Metadata Hits (%)

This metric specifies the percentage of requests that are able to retrieve the cached page metadata content from WebCache.

9.8.13.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.13.2 User Action

Use this value to understand how effectively WebCache is being used for caching the page metadata. If this value is low, review the caching options for the pages.

9.8.14 WebCache Page Metadata Misses (%)

This metric specifies the percentage of requests that are not able to retrieve hit the cached page metadata content from WebCache.

9.8.14.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.14.2 User Action

Use this metric to understand how effectively WebCache is being used for caching the page metadata for your pages.

9.8.15 WebCache Page Metadata Non-Cacheables (%)

This metric specifies the percentage of requests that cannot use WebCache to cache content.

9.8.15.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

9.8.15.2 User Action

If this value is high, use this value to analyze how much of the pages are defined to not use WebCache for page metadata caching. This is not a common option for most pages and has performance implications. Unless there are specific reasons, review the caching options for such pages. See "Improving Page Performance" chapter in "User's Guide for Oracle Portal" document in OTN for performance tuning information.

9.9 Response

This category provides information about response metrics.

9.9.1 UpDown Status

This metric indicates the status of Portal. A green up arrow indicates that Portal is up and running. A red down arrow indicates that Portal is down. A "timer-clock" state icon indicates that Enterprise Manager is unable to query and retrieve the Portal status information.

9.9.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every 60 Samples	=	Not Defined	0	1	The Portal Application is down

9.9.1.2 User Action

If the status of Portal is down, check if the:

- Portal WebLogic container (WLS_PORTAL) is running. If required, start it.
- Portal application within the Portal WebLogic container has been started. If required, start the application.

If the container and the application are running and the status is still down, use the Portal diagnostic logs to find out why.

This chapter describes Oracle Discoverer metrics.

10.1 Discoverer

Oracle Business Intelligence Discoverer is a business intelligence tool for analyzing data. Discoverer provides an integrated business intelligence solution comprising a set of intuitive ad-hoc query, reporting, analysis, and Web-publishing tools. These tools enable nontechnical users to gain immediate access to information from data marts, data warehouses, and online transaction processing systems. Discoverer integrates seamlessly with Oracle Portal, enabling rapid deployment of Discoverer workbooks and worksheets to Web portals.

10.2 Discoverer Components

Provides information about the metrics aggregated at the level of each Discoverer component.

10.2.1 Total CPU Time (millisec)

This metric specifies the total CPU usage time (in milliseconds) for all the Discoverer server sessions of the selected component on the host machine.

10.2.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.2.2 CPU Usage (%)

This metric specifies the CPU usage (in percentage) for the Discoverer server sessions of the selected component on the host machine.

10.2.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 10 Hours
--------------	----------------

10.2.3 Memory Usage (MB)

This metric specifies the memory (in megabytes) used by the Discoverer server sessions of the selected component on the host machine.

10.2.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.2.4 Total Private Memory (MB)

This metric specifies the total private memory (in megabytes) used by the Discoverer server sessions of the selected component on the host machine.

10.2.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.2.5 Number of Sessions

This metric specifies the total number of Discoverer server sessions that are active for the selected component on the host machine.

10.2.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.3 Discoverer Sessions

This metric displays information about the Discoverer server sessions that are currently active.

10.3.1 Component Name

This metric indicates the name of the Discoverer component for which the metrics are displayed.

10.3.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 10 Hours
--------------	----------------

10.3.2 Connection Details

This metric specifies the database user, the database, and the EUL to which the session is connected. The format of this metric value is databaseuser@database - EUL.

10.3.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.3.3 Total CPU Time (millisec)

This metric specifies the total CPU usage time (in milliseconds) for the Discoverer server session on the host machine.

10.3.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.3.4 CPU Usage (%)

This metric specifies the CPU usage (in percentage) for the Discoverer server session on the host machine.

10.3.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.3.5 Memory Usage (MB)

This metric specifies the amount of memory (in megabytes) used by the Discoverer server session on the host machine.

10.3.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.3.6 Port Number

10.3.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.3.7 Total Private Memory (MB)

This metric specifies the amount of total private memory (in megabytes) used by the Discoverer server session on the host machine.

10.3.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.3.8 Session ID

This metric indicates the discoverer client session ID.

10.3.8.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.3.9 SSO User Name

If Single Sign-On (SSO) is enabled for Discoverer, this column identifies the SSO user that is using the current session. Information about the SSO user is meaningful only for Discoverer Plus and Discoverer Viewer because SSO users work in Plus and Viewer sessions interactively. For Discoverer Portlet Provider, information about the SSO user might not be meaningful because Discoverer Portlet Provider runs queries in the background, and a query could run as a given SSO user even when that SSO user is not currently logged in.

10.3.9.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.4 Discoverer Application

Provides information about metrics at the Discoverer application level, aggregated for all the server sessions.

10.4.1 CPU Idle (%)

This metric specifies the CPU idle time (in percentage) on the host machine.

10.4.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.4.2 Total CPU Time (millisec)

This metric specifies the total CPU usage time (in milliseconds) for all Discoverer server sessions on the host machine.

10.4.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.4.3 CPU Usage (%)

This metric specifies the CPU usage (in percentage) for all Discoverer server sessions on the host machine.

10.4.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.4.4 Available Physical Memory (MB)

This metric specifies the amount of free physical memory (in megabytes) on the host machine.

10.4.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.4.5 Host

The metric indicates the host name.

10.4.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.4.6 Memory Usage (MB)

This metric specifies the memory (in megabytes) used by all the Discoverer server sessions on the host machine.

10.4.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.4.7 Total Private Memory (MB)

This metric specifies the total private memory (in megabytes) used by all the Discoverer server sessions on the host machine.

10.4.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.4.8 Number of Sessions

This metric specifies the total number of Discoverer server sessions that are active on the host machine.

10.4.8.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.4.9 Total Physical Memory (MB)

This metric specifies the total physical memory (in megabytes) available on the host machine.

10.4.9.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Hours

10.5 Discoverer Components

This metric displays the total CPU and memory consumption for all current sessions created by each Discoverer component (Discoverer Plus, Discoverer Viewer, or Discoverer Portlet Provider). Note that the total CPU and memory consumption displayed here is only for the sessions and does not include the CPU and memory consumption of the servlets for these components. This metric also displays the total number of sessions created by the component.

10.5.1 CPU Usage (%)

This metric specifies the total CPU usage for the sessions created by each Discoverer UI component. The total CPU usage does not include CPU usage for the servlets of these components. For CPU usage of servlets, see the Discoverer application page under the OC4J_BI_Forms home page.

10.5.2 Memory Usage (MB)

This metric specifies the total memory usage for the sessions spawned by the given UI component. The total memory usage does not include memory usage for the servlets of these components. For memory usage of servlets, see the Discoverer application page under the OC4J_BI_Forms home page.

10.5.3 Number of Sessions

This metric specifies the total number of Discoverer sessions created by the given component.

10.6 Total Discoverer CPU Usage

This metric displays the total CPU percentage used by all Discoverer sessions running on the machine at a given time. The total CPU usage does not include the CPU usage of the servlets for Discoverer Plus, Viewer, or Portlet Provider. For the CPU usage of these servlets, see the Discoverer application page from the OC4J_BI_Forms home page in the EM standalone console.

10.6.1 Total Discoverer CPU Usage (%)

This metric specifies the percentage of CPU used by all Discoverer sessions running on the machine at a given time. The total CPU usage does not include the CPU usage of the servlets for Discoverer Plus, Viewer, and Portlet Provider. For the CPU usage of these servlets, see the Discoverer application page from the OC4J_BI_Forms home page in the EM standalone console.

10.6.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 12 Samples	>	Not Defined	Not Defined	2	CPU Usage is %value%%%

10.7 Total Discoverer Memory Usage

This metric displays the total memory used by all Discoverer sessions running on the machine at a given time. The total memory usage does not include the memory usage of the servlets for Discoverer Plus, Viewer, or Portlet Provider. For the memory usage of these servlets, see the Discoverer application page from the OC4J_BI_Forms home page in the EM standalone console.

10.7.1 Total Discoverer Memory Usage (MB)

This metric specifies the total memory usage for all Discoverer sessions.

10.7.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 12 Samples	>	Not Defined	Not Defined	2	Memory Usage is %value% MB

10.8 Discoverer Plus Sessions

This metric displays information about all the Discoverer sessions created by Discoverer Plus. The information included for each session is the session ID, OS Process ID, memory usage of the session, CPU usage for the session, database and EUL to which the session is connected, and, if SSO is enabled, the SSO user using the session.

10.8.1 Component Type

This metric specifies the Discoverer UI component that started the Discoverer Plus, Discoverer Viewer, or Discoverer Portlet Provider session.

10.8.2 CPU Usage (%)

This metric specifies the percentage CPU usage for this session.

10.8.3 DBUser@DB - EUL

This metric specifies the database user, database name, and EUL for this session.

10.8.4 Memory Usage (MB)

This metric specifies the memory used (in megabytes) for this session.

10.8.5 OS Process ID

This metric indicates the operating system process ID for the session.

10.8.6 SSO User

This metric indicates the name of the Single Sign-On (SSO) user for this session.

10.9 Discoverer Portlet Provider Sessions

This metric displays information about all the Discoverer sessions created by Discoverer Portlet Provider. The information included for each session is the session ID, OS Process ID, memory usage of the session, CPU usage for the session, database and EUL to which the session is connected, and, if SSO is enabled, the SSO user using the session.

10.9.1 Component Type

This metric specifies the Discoverer UI component that started the Discoverer Plus, Discoverer Viewer or Discoverer Portlet Provider session.

10.9.2 CPU Usage (%)

This metric specifies the percentage CPU usage for this session.

10.9.3 DBUser@DB - EUL

This metric specifies the database user, database name, and EUL for this session.

10.9.4 Memory Usage (MB)

This metric specifies the memory used (in megabytes) for this session.

10.9.5 OS Process ID

This metric specifies the operating system process ID for this session.

10.9.6 SSO User

If Single Sign-On (SSO) is enabled for Discoverer, this column identifies the SSO user that is using the current session. Information about the SSO user is meaningful only for Discoverer Plus and Discoverer Viewer because SSO users work in Plus and Viewer sessions interactively. For Discoverer Portlet Provider, information about the SSO user might not be meaningful because Discoverer Portlet Provider runs queries in the background, and a query could run as a given SSO user even when that SSO user is not currently logged in.

10.10 Discoverer Sessions

This metric displays information about all the Discoverer server sessions that are currently running on the server. It includes the following information: session ID, OS Process ID, Discoverer UI component (that is, Discoverer Plus, Discoverer Viewer, or Discoverer Portlet Provider) that created the session, memory usage of the session, CPU usage for the session, database and EUL to which the session is connected, and, if Single Sign-On (SSO) is enabled, the SSO user using the session.

10.10.1 Component Type

This metric indicates the Discoverer UI component that started the Discoverer Plus, Discoverer Viewer or Discoverer Portlet Provider session.

10.10.2 CPU Usage (%)

This metric specifies the percentage of CPU used for all the Discoverer sessions.

10.10.3 DBUser@DB - EUL

This metric specifies the database user, the database, and the EUL to which the session is connected. The format is databaseuser@database - EUL.

10.10.4 Memory Usage (MB)

This metric specifies the memory used (in megabytes) for all the Discoverer sessions.

10.10.5 OS Process ID

This metric indicates the operating system process ID for the given session.

10.10.6 SSO User

If Single Sign-On (SSO) is enabled for Discoverer, this column identifies the SSO user that is using the current session. Information about the SSO user is meaningful only for Discoverer Plus and Discoverer Viewer because SSO users work in Plus and Viewer sessions interactively. For Discoverer Portlet Provider, information about the SSO user might not be meaningful because Discoverer Portlet Provider runs queries in the background, and a query could run as a given SSO user even when that SSO user is not currently logged in.

10.11 Total Number Of Discoverer Plus Sessions

This metric displays the total number of sessions created by Discoverer Plus at any given time on the machine.

10.11.1 Total Number Of Discoverer Plus Sessions

This metric specifies the total number of sessions created by Discoverer Plus at any given time on the machine.

10.12 Total Number Of Discoverer Portlet Provider Sessions

This metric displays the total number of sessions created by Discoverer Portlet Provider at any given time on the machine.

10.12.1 Total Number Of Discoverer Portlet Provider Sessions

This metric specifies the total number of sessions created by Discoverer Portlet Provider at any given time on the machine.

10.13 Total Number Of Discoverer Sessions

This metric displays the total number of Discoverer sessions running on the machine at a given time.

10.13.1 Total Number Of Discoverer Sessions

This metric specifies the total number of Discoverer sessions running on the machine at a given time.

10.14 Total Number Of Discoverer Viewer Sessions

This metric displays the total number of sessions created by Discoverer Viewer at any given time on the machine.

10.14.1 Total Number Of Discoverer Viewer Sessions

This metric specifies the total number of sessions created by Discoverer Viewer at any given time on the machine.

10.15 Discoverer Viewer Sessions

This metric displays information about all the Discoverer sessions created by Discoverer Viewer. The information included for each session is the session ID, OS Process ID, memory usage of the session, CPU usage for the session, database and EUL to which the session is connected, and, if SSO is enabled, the SSO user using the session.

10.15.1 Component Type

This metric specifies the Discoverer UI component that started the Discoverer Plus, Discoverer Viewer or Discoverer Portlet Provider session.

10.15.2 CPU Usage (%)

This metric specifies the percentage CPU usage for this session.

10.15.3 DBUser@DB - EUL

This metric specifies the database user, database name, and EUL for this session.

10.15.4 Memory Usage (MB)

This metric specifies the memory used (in megabytes) for this session.

10.15.5 OS Process ID

This metric specifies the operating system process ID for this session.

10.15.6 SSO User

If Single Sign-On (SSO) is enabled for Discoverer, this column identifies the SSO user that is using the current session. Information about the SSO user is meaningful only for Discoverer Plus and Discoverer Viewer because SSO users work in Plus and Viewer sessions interactively. For Discoverer Portlet Provider, information about the SSO user might not be meaningful because Discoverer Portlet Provider runs queries in the background, and a query could run as a given SSO user even when that SSO user is not currently logged in.

10.16 Response

This metric displays information that indicates whether Discoverer is up or down.

10.16.1 UpDown Status

This metric displays information that indicates whether Discoverer is up or down.

10.16.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 12 Samples	=	Not Defined	0	1	The Discoverer instance is down

This chapter describes Oracle Reports Bridge metrics.

11.1 Log Message Alerts

This category provides information about log message alerts.

11.1.1 Log Monitoring Rules

11.1.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Key	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	odl_message_alert_rule_name: "MSG-110"	Not Defined	After Every Sample	MATCH	YES	Not Defined	1*	[%odl_message_alert_rule_name%] matched %odl_message_alert_occurrence_count% log message(s) since last scan. Message Text for the last matching message is as follows - [%odl_message_alert_matching_text%].
All Versions	null	Not Defined	After Every Sample	MATCH	Not Defined	Not Defined	1*	[%odl_message_alert_rule_name%] matched %odl_message_alert_occurrence_count% log message(s) since last scan. Message Text for the last matching message is as follows - [%odl_message_alert_matching_text%].

* Once an alert is triggered for this metric, it must be manually cleared.

11.1.1.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each unique combination of "Alert Rule Name" and "Occurrence Time" objects.

If warning or critical threshold values are currently set for any unique combination of "Alert Rule Name" and "Occurrence 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 "Alert Rule Name" and "Occurrence Time" objects, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

11.2 Diagnostic Log Summary

This category provides information about diagnostic log summary metrics.

11.2.1 Log Error Messages (over 15 minutes)

11.2.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 15 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Error message rate is %value%, crossed warning (%warning_threshold%) or critical(%critical_threshold%) threshold.

11.2.2 Log Incident Error Messages (over 15 minutes)

11.2.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 15 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Incident Error message rate is %value%, crossed warning (%warning_threshold%) or critical(%critical_threshold%) threshold.

11.2.3 Log Notification Messages (over 15 minutes)

11.2.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 15 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Notification message rate is %value%, crossed warning (%warning_threshold%) or critical(%critical_threshold%) threshold.

11.2.4 Log Trace Messages (over 15 minutes)

11.2.4.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 15 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Trace message rate is %value%, crossed warning (%warning_threshold%) or critical(%critical_threshold%) threshold.

11.2.5 Log Unknown Messages (over 15 minutes)

11.2.5.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 15 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Unknown message rate is %value%, crossed warning (%warning_threshold%) or critical(%critical_threshold%) threshold.

11.2.6 Log Warning Messages (over 15 minutes)

11.2.6.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 15 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Warning message rate is %value%, crossed warning threshold(%warning_threshold%) or critical(%critical_threshold%) threshold.

11.3 Remote Elements Discovered

Provides information on Bridges and Servers discovered.

11.3.1 Host.value

This metric specifies the host name of the machine in which the remote element is located.

11.3.1.1 User Action

No action required.

11.3.2 Name

This metric specifies the name of the remote element.

11.3.2.1 User Action

No action required.

11.3.3 Discovery Time (ms)

This metric specifies the time taken to discover the remote element.

11.3.3.1 User Action

If the time taken to complete this action is high, check for any network issues. Also, tune the timeout parameter in the bridge configuration file to allow for high discovery time.

11.3.4 Type of Element

This metric specifies the type of the remote element. Type can be "Server" or "Bridge".

11.3.4.1 User Action

No action required.

11.4 Resource Usage

This category provides information about resource usage metrics.

11.4.1 CPU Usage (%)

This metric specifies the percentage of CPU usage.

11.4.2 Memory Usage (MB)

This metric specifies the memory used (in megabytes).

11.4.3 Memory Usage (%)

This metric specifies the percentage of total memory used.

11.5 Response

Provides information about the status of Reports Bridge, Up or Down.

11.5.1 Up or Down Status

This metric specifies the status of Reports Bridge. Status can be Up or Down.

11.5.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every 60 Samples	=	Not Defined	0	1	%target% is down

11.5.1.2 User Action

If status is down, see the Report log files or use EM log viewer to find out why the Reports Bridge is down.

This chapter describes Oracle Virtual Directory metrics.

12.1 Adapter Metrics

The category specifies the metrics about the currently configured Adapters.

12.1.1 Open Connections

This metric specifies the total number connections opened by a particular Adapter since last collection.

12.1.1.1 User Action

No action required.

12.1.2 Total No of Connections Processed

This metric specifies the total number of connections processed by the Adapter.

12.1.2.1 User Action

No action required.

12.1.3 Total No of Connections Reused

This metric specifies the total number of connections reused by the Adapter.

12.1.3.1 User Action

No action required.

12.1.4 Enabled

This metric specifies if the Adapter is enabled or disabled.

12.1.4.1 User Action

Enable or disable an adapter using Oracle Directory Services Manager by logging in to Oracle Directory Services Manager, selecting Adapter from the task selection bar, clicking the adapter you want to enable or disable, setting the Active option, and then clicking Apply.

12.1.5 Add Operations

This metric specifies the total number of Add operations performed by a particular Adapter since last collection.

12.1.5.1 User Action

No action required.

12.1.6 Total No of Operation Binds

This metric specifies the total number of bind operations by a particular Adapter.

12.1.6.1 User Action

No action required.

12.1.7 Operations Count

Specifies the total number of all operations performed by this particular Adapter.

12.1.7.1 User Action

No action required.

12.1.8 Delete Operations

This metric specifies the total number of delete operations performed by a particular Adapter since last collection.

12.1.8.1 User Action

No action required.

12.1.9 Operational Version

This metric specifies the operational version of Adapters.

12.1.9.1 User Action

No action required.

12.1.10 Modify Operations

This metric specifies the total number of Modify operations performed by a particular Adapter since last collection.

12.1.10.1 User Action

No action required.

12.1.11 Total No of Operations Renamed

This metric specifies the total number of operations renamed by a particular Adapter.

12.1.11.1 User Action

No action required.

12.1.12 Search Operations

This metric specifies the total number of Search operations performed by a particular Adapter since last collection.

12.1.12.1 User Action

No action required.

12.1.13 Provisioned version

This metric specifies the Processed version of Adapters.

12.1.13.1 User Action

No action required.

12.1.14 Type

This metric specifies the type of the Adapter.

12.1.14.1 User Action

No action required.

12.1.15 Average Time to complete a search request

This metric specifies Average Time a particular adapter took to complete a search request since the last Oracle Virtual Directory start.

12.1.15.1 User Action

No action required.

12.1.16 Maximum Time taken to complete a search request

This metric specifies the maximum time since the last Oracle Virtual Directory start that a particular Adapter took to complete an LDAP search request.

12.1.16.1 User Action

No action required.

12.1.17 Minimum Time taken to complete a search request

This metric specifies the minimum time since the last Oracle Virtual Directory start that a particular Adapter took to complete an LDAP search request.

12.1.17.1 User Action

No action required.

12.2 Connected IPs

This category provides information about distinct IP Address that are connected to Oracle Virtual Directory

12.2.1 Current Connections

This metric specifies the Current Open connections from a particular IP Address.

12.2.1.1 User Action

Set connection Activity Limits by selecting Administration from the Oracle Virtual Directory menu and then Server Properties. Select the Enable Quota Enforcement option on the Server Properties page, edit the Maximum Connections per IP Address setting, and then click Apply.

12.2.2 Total Connections

This metric specifies the Total number of connections from a particular IP Address.

12.2.2.1 User Action

Set connection Activity Limits by selecting Administration from the Oracle Virtual Directory menu and then Server Properties. Select the Enable Quota Enforcement option on the Server Properties page, edit the Maximum Connections per IP Address setting, and then click Apply.

12.3 getUsers

The category provides information about currently connected users

12.3.1 Current Connections

This metric specifies the number of connections that are currently open for a particular user.

12.3.1.1 User Action

Set connection Activity Limits by selecting Administration from the Oracle Virtual Directory menu and then Server Properties. Select the Enable Quota Enforcement option on the Server Properties page, edit the Maximum Client Connections, Maximum Connections per Authenticated Subject, and Maximum Connections per IP Address settings, and then click Apply.

12.3.2 Total Connections

This metric specifies the total number of connections opened by a particular user.

12.3.2.1 User Action

Set connection Activity Limits by selecting Administration from the Oracle Virtual Directory menu and then Server Properties. Select the Enable Quota Enforcement option on the Server Properties page, edit the Maximum Client Connections, Maximum Connections per Authenticated Subject, and Maximum Connections per IP Address settings, and then click Apply.

12.4 Resource Usage

This category provides information about CPU and memory usage.

12.4.1 CPU Usage (%)

This metric specifies the percentage of the CPU that Oracle Virtual Directory is using.

12.4.1.1 User Action

No action required.

12.4.2 CPU Idle Time (%)

This metric specifies the percentage of the CPU Idle Time on the Oracle Virtual Directory Host.

12.4.2.1 User Action

No action required.

12.4.3 Other CPU Usage (%)

This metric specifies the percentage of the CPU that components other than Oracle Virtual Directory are using.

12.4.3.1 User Action

No action required.

12.4.4 Heap Usage (MB)

This metric specifies the JVM Heap Usage of the Oracle Virtual Directory Server.

12.4.4.1 User Action

No action required.

12.4.5 Memory Usage (MB)

This metric specifies the Memory Consumed (in MB) by Oracle Virtual Directory.

12.4.5.1 User Action

No action required.

12.4.6 Free Memory (MB)

This metric specifies the amount of Free Memory on the Oracle Virtual Directory Server Host.

12.4.6.1 User Action

No action required.

12.4.7 Other Memory Usage (MB)

This metric specifies the Memory Consumed (in MB) by components other than Oracle Virtual Directory.

12.4.7.1 User Action

No action required.

12.4.8 Total Memory (MB)

This metric specifies the Total Available Memory on the Oracle Virtual Directory Server Host.

12.4.8.1 User Action

No action required.

12.4.9 Memory Usage (%)

This metric specifies the percentage of Memory Consumed by the Oracle Virtual Directory Server.

12.4.9.1 User Action

No action required.

12.4.10 Free Memory (%)

This metric specifies the percentage of Free Memory on the Oracle Virtual Directory Server Host.

12.4.10.1 User Action

No action required.

12.4.11 Other Memory Usage (%)

This metric specifies the percentage of Memory Consumed by components other than the Oracle Virtual Directory Server.

12.4.11.1 User Action

No action required.

12.4.12 Start Time (ms since Epoch)

This metric specifies the start time (milliseconds since Epoch) of the Oracle Virtual Directory Server.

12.4.12.1 User Action

No action required.

12.4.13 Up Time (ms since Epoch)

This metric specifies the amount of time (in milliseconds) that the Oracle Virtual Directory Server has been up and running.

12.4.13.1 User Action

No action required.

12.5 Response

This category shows whether the Oracle Virtual Directory instance is up and running or down and unavailable.

12.5.1 UpDown Status

This metric specifies whether the Oracle Virtual Directory instance is up and running or down and unavailable.

12.5.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold
All Versions	Every Minute	After Every 60 Samples	=	Not Defined	0

12.5.1.2 User Action

Change the status of the Oracle Virtual Directory instance by selecting Control from the Oracle Virtual Directory menu and then selecting either Start Up or Shut Down.

12.6 Server Latency Metrics

This category defines the Oracle Virtual Directory Server Latency Metrics

12.6.1 Total Operations

This metric specifies the total number of all LDAP operations since the last collection.

12.6.1.1 User Action

No action required.

12.6.2 Average time to complete an LDAP search request

This metric specifies the Average Time to complete an LDAP search request since the last Oracle Virtual Directory start.

12.6.2.1 User Action

No action required.

12.6.3 Maximum Time to complete a search request

This metric specifies the Maximum Time for Oracle Virtual Directory to complete a search request since its last start.

12.6.3.1 User Action

No action required.

12.6.4 Minimum Time to complete a search request

This metric specifies the Minimum Time for Oracle Virtual Directory to complete a search request since its last start.

12.6.4.1 User Action

No action required.

12.6.5 Number of LDAP Add Requests

This metric specifies the Number of LDAP Add Requests since the last collection.

12.6.5.1 User Action

No action required.

12.6.6 Number of LDAP Binds Requests

This metric specifies the Number of LDAP Bind Requests since the last collection.

12.6.6.1 User Action

No action required.

12.6.7 Number of LDAP Delete Requests

This metric specifies the Number of LDAP Delete Requests since the last collection.

12.6.7.1 User Action

No action required.

12.6.8 Number of LDAP Modify Requests

This metric specifies the Number of LDAP Modify Requests since the last collection.

12.6.8.1 User Action

No action required.

12.6.9 Number of LDAP Rename Requests

This metric specifies the Number of LDAP Rename Requests since the last collection.

12.6.9.1 User Action

No action required.

12.6.10 Number of LDAP Search Requests

This metric specifies the Number of LDAP Search Requests since the last collection.

12.6.10.1 User Action

No action required.

12.7 Server wide Connection Metrics

This category shows the Oracle Virtual Directory operational Metrics.

12.7.1 Total No of IPs Currently Connected

This metric specifies the Total number of distinct IP Addresses that are currently connected to Oracle Virtual Directory.

12.7.1.1 User Action

Set connection Activity Limits by selecting Administration from the Oracle Virtual Directory menu and then Server Properties. Select the Enable Quota Enforcement option on the Server Properties page, edit the Maximum Connections per IP Address setting, and then click Apply.

12.7.2 Total No of Users Currently Connected

This metric specifies the Total number of users that are currently connected to Oracle Virtual Directory.

12.7.2.1 User Action

Set connection Activity Limits by selecting Administration from the Oracle Virtual Directory menu and then Server Properties. Select the Enable Quota Enforcement option on the Server Properties page, edit the Maximum Client Connections, Maximum Connections per Authenticated Subject, and Maximum Connections per IP Address settings, and then click Apply.

12.7.3 Total No of Operations

This metric specifies the Total number of all LDAP operations that have been completed from the last collection.

12.7.3.1 User Action

Set the maximum number of operations to allow for each connection by selecting Administration from the Oracle Virtual Directory menu and then Server Properties. Select the Enable Quota Enforcement option on the Server Properties page, enter the maximum number of operations to allow for each connection in the Maximum Operations per Connection field, and click Apply.

12.7.4 Total No of Open Connections

This metric specifies the Total number of open connections to Oracle Virtual Directory.

12.7.4.1 User Action

No action required.

Internet Directory

This chapter describes Oracle Internet Directory metrics.

13.1 Oracle Internet Directory

Oracle Enterprise Manager can be used to manage Oracle Internet Directory server. You can also use Enterprise Manager to view the metrics collected by Oracle Internet Directory LDAP server, replication server and directory integration server.

13.2 (Resource Statistics) LDAP Server's Active Database Connections

Shows the number of active database connections for the Oracle Internet Directory server.

13.3 Active Database Sessions

The number of active database connections for the Oracle Internet Directory server.

13.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	LDAP Active DB Connections %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.4 Closed LDAP Logon Session Statistics

This category provides information about closed LDAP Logon Session statistics.

13.4.1 Total Closed Logon Sessions

Provides information about total closed logon sessions to Oracle Internet Directory servers.

13.4.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold Id	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	OID Server Closed TCP conn %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.5 (Historical Resource Statistics)LDAP Server's Database Usage

This metric provides information about database usage by Oracle Internet Directory servers.

13.6 Active Database Sessions

Shows the number of active database sessions of the Oracle Internet Directory server instance at the end of the metric collection period.

13.6.1 User Action

Specific to your site.

13.7 Date and Time Stamp

Shows the date and time stamp (in UTC) of the metric collection.

13.7.1 User Action

Specific to your site.

13.8 Instance Number

Shows the instance number of the Oracle Internet Directory server instance for which this metric is collected.

13.8.1 User Action

Specific to your site.

13.9 Open Database Sessions

Shows the number of open database sessions of the Oracle Internet Directory server instance at the end of the metric collection period.

13.9.1 User Action

Specific to your site.

13.10 Failed Bind Operations

The metric value specifies the number of unsuccessful LDAP bind operations performed by a user

13.10.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Failed Bind Operation %value% crossed warning threshold (%warning_threshold%) or critical (%critical_threshold%)

13.11 (Critical Events)General System Resource Events

This category provides information about general system resource events.

13.11.1 Critical General Sys Resource Event Occurrences

Number of occurrences of general system resource events.

13.11.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	0	1	LDAP Server crashed %value% times, crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.11.2 Critical General Sys Resource Event Type

Type of general system resource event.

13.11.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Minutes

13.12 LDAP Load

This metric provides information about the total number of LDAP operations in progress.

13.13 Server Load

The total number of LDAP operations in progress.

13.13.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	LDAP Server Load %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.14 Historical LDAP Load and Response

This metric provides information about the total number of LDAP operations in progress and the average LDAP operation response time in Oracle Internet Directory servers.

13.14.1 Date and Time Stamp

Shows the date and time stamp (in UTC) of the metric collection.

13.14.1.1 User Action

Specific to your site.

13.14.2 Instance Number

Shows the instance number of the Oracle Internet Directory server instance for which this metric is collected.

13.14.2.1 User Action

Specific to your site.

13.14.3 Server Load

Shows the number of LDAP operations in progress in the Oracle Internet Directory server at the end of the metric collection period.

13.14.3.1 User Action

Specific to your site.

13.14.4 Server Average Response Time (in Milliseconds)

Shows the average response time in (milliseconds) for the Oracle Internet Directory server to perform an LDAP operation during the metric collection period.

13.14.4.1 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.15 (Resource Statistics)LDAP Server Memory Growth

Provides information about Oracle Internet Directory server memory growth.

13.15.1 Average memory growth (%)

Average Oracle Internet Directory server memory growth.

13.15.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	OID Average Memory Growth %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.16 New LDAP Logon Session Statistics

This category provides information about new LDAP logon session statistics.

13.16.1 Total New Logon Sessions

Provides information about total new logon sessions to Oracle Internet Directory servers.

13.16.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	OID Server New TCP conn %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.17 (Resource Statistics) LDAP Server's Open Database Connections

Shows the number of open database connections for the Oracle Internet Directory server.

13.17.1 Open Database Sessions

The number of open database connections for the Oracle Internet Directory server.

13.17.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	LDAP Open DB Connections %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.18 Open LDAP Logon Session Statistics

This category provides information about open LDAP logon session statistics.

13.18.1 Total Open Logon Sessions

Provides information about total logon sessions to Oracle Internet Directory servers.

13.18.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total Logon Sessions %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.18.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.19 Historical LDAP Operation Response Time

This category provides information about historical LDAP operation response time metrics.

13.19.1 Bind Operation Response Time

The average server response time for bind operations

13.19.1.1 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.19.2 Compare Operation Response Time

The average server response time for compare operations

13.19.2.1 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.19.3 Date and Time Stamp

The date and time stamp of this server instance.

13.19.4 Instance Number

The instance number of this Oracle Internet Directory server instance.

13.19.5 Messaging Search Operation Response Time

The metric value shows the average messaging search response time

13.19.5.1 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.20 LDAP Operation Response Time

This category provides information about LDAP operation response time metrics.

13.20.1 Bind Operation Response Time

The metric value shows the response time of LDAP Bind operations

13.20.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Bind Operation Response Time %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.20.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.20.2 Compare Operation Response Time

The metric value shows the response time of LDAP compare operations

13.20.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Compare Operation Response Time %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.20.2.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.20.3 Messaging Search Operation Response Time

The metric value shows the average messaging search response time

13.20.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding

Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Mesg Search Operation Response Time %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%)

13.20.3.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.21 LDAP Server Resource Usage

Provides information about system resources used by Oracle Internet Directory servers.

13.21.1 Total CPU Usage (%)

Provides information about the percentage of CPU used by Oracle Internet Directory servers.

13.21.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.4.x	Every 5 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total CPU Usage %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%)

13.21.1.2 User Action

Specific to your site.

13.21.2 Total Memory Usage (KB)

Provides information about total virtual memory used by Oracle Internet Directory servers.

13.21.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.4.x	Every 5 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total Memory Usage %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.21.2.2 User Action

Specific to your site.

13.21.3 Total Memory Usage (%)

Provides information about percent total virtual memory used by Oracle Internet Directory servers.

13.21.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.4.x	Every 5 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total Memory Usage %value% (%%) crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.22 LDAP Response

This metric provides information about the average LDAP operation response time in Oracle Internet Directory servers.

13.22.1 Server Response

The average LDAP operation response time in Oracle Internet Directory servers.

13.22.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	LDAP Server Response %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%)

13.22.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.23 Running instances of LDAP Server

Provides information about running instances of Oracle Internet Directory servers.

13.23.1 Downtime Count

Shows the number of times the instance has gone down and has been restarted by the Oracle Internet Directory monitor daemon.

13.23.1.1 User Action

None.

13.23.2 Host Name

Shows the host name of a running Oracle Internet Directory instance.

13.23.2.1 User Action

None.

13.23.3 Instance Number

The instance number of this Oracle Internet Directory server instance.

13.23.4 Up Since

Shows the time when the Oracle Internet Directory instance was first started.

13.23.4.1 User Action

None.

13.23.5 Non SSL Port Number

The metric value specifies the non-SSL port that an instance of Oracle Internet Directory server is listening on.

13.23.6 SSL Port Number

The metric value specifies the SSL port that an instance of Oracle Internet Directory server is listening on.

13.24 (Historical Resource Statistics)LDAP Server and System Memory

Provides information about system memory and Oracle Internet Directory server memory.

13.24.1 Date and Time Stamp

Shows the date and time stamp (in UTC) of the metric collection.

13.24.1.1 User Action

Specific to your site.

13.24.2 Instance Number

Shows the instance number of the Oracle Internet Directory server instance for which this metric is collected.

13.24.2.1 User Action

Specific to your site.

13.24.3 LDAP Server's average memory growth (%)

Shows the percent of memory growth of the Oracle Internet Directory server instance at the end of the metric collection period.

13.24.3.1 User Action

Specific to your site.

13.24.4 System's Total Free Physical Memory (KB)

Shows, in kilobytes, the free physical system memory at the end of metric collection period.

13.24.4.1 User Action

Specific to your site.

13.25 (Critical Events)System Resource Events(3113 Errors)

Provides information about ORA-3113 errors that occurred in Oracle Internet Directory servers.

13.25.1 Number of 3113 Error Occurrences

Provides information about ORA-3113 errors that occurred in Oracle Internet Directory servers. This error indicates a loss of connection to database.

13.25.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Minutes

13.25.1.2 User Action

Check for network problems. Check whether database has crashed.

13.26 (Critical Events)System Resource Events(3114 Errors)

Provides information about ORA-3114 errors that occurred in Oracle Internet Directory servers.

13.26.1 Number of 3114 Error Occurrences

Provides information about ORA-3114 errors that occurred in Oracle Internet Directory servers. This error indicates a loss of connection to database.

13.26.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Minutes

13.26.1.2 User Action

Check for network problems. Check whether database has crashed.

13.27 (Historical Critical Events)System Resource Events

Provides information about critical system resource events that occurred in Oracle Internet Directory servers.

13.27.1 Last Event Time

Shows the date and time stamp (in UTC) of the event occurrence.

13.27.1.1 User Action

Specific to your site.

13.27.2 Number of Event Occurrence

The number of occurrences of this event

13.27.3 Event Name

Shows the name of the event that occurred.

13.27.3.1 User Action

Specific to your site.

13.27.4 Instance Number

Shows the instance number of the Oracle Internet Directory server for which this metric is collected.

13.27.4.1 User Action

Specific to your site.

13.27.5 Process ID

Shows the process identifier of the Oracle Internet Directory server instance for which this metric is collected.

13.27.5.1 User Action

Specific to your site.

13.28 (Critical Events)System Resource Events(Ora Errors)

This category provides information about System Resource Events(Ora Errors) metrics.

13.28.1 Critical Ora Error Occurrences

Number of occurrences of critical ORA Error Code Number.

13.28.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	CONTAIN S	Not Defined	ORA-	1	Ora Error!

13.29 Historical LDAP Logon Session Statistics

This metric provides information about login sessions established with Oracle Internet Directory servers.

13.29.1 Date and Time Stamp

Shows the date and time stamp (in UTC) of the metric collection.

13.29.1.1 User Action

Specific to your site.

13.29.2 Instance Number

Shows the instance number of the Oracle Internet Directory server instance for which this metric is collected.

13.29.2.1 User Action

Specific to your site.

13.29.3 Total LDAP Logon Sessions

Shows the number of LDAP sessions established with the Oracle Internet Directory server at the end of the metric collection period.

13.29.3.1 User Action

Specific to your site.

13.30 Total Memory Usage

Provides information about total virtual memory use by Oracle Internet Directory servers.

13.30.1 Total Mem. Size(in KB)

Shows total virtual memory use by Oracle Internet Directory servers.

13.30.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
9.0.2.x and 9.0.3.x	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Memory Usage over %value% KB crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.30.1.2 User Action

Specific to your deployment.

13.31 Historical LDAP Operations Profile

This metric provides information about LDAP operations completed by Oracle Internet Directory servers.

13.31.1 Date and Time Stamp

Shows the date and time stamp (in UTC) of the metric collection.

13.31.1.1 User Action

Specific to your site.

13.31.2 Instance Number

Shows the instance number of the Oracle Internet Directory server instance for which this metric is collected.

13.31.2.1 User Action

Specific to your site.

13.31.3 Completed Abandon Operations

The metric value specifies the number of LDAP abandon operations performed by a user

13.31.4 Completed Add Operations

Shows the total number of LDAP add operations completed by the Oracle Internet Directory server instance during the metrics collection period.

13.31.4.1 User Action

Specific to your site.

13.31.5 Completed Bind Operations

Shows the total number of LDAP bind operations completed by the Oracle Internet Directory server instance during metric collection.

13.31.5.1 User Action

Specific to your site.

13.31.6 Completed Compare Operations

Shows the total number of LDAP compare operations completed by the Oracle Internet Directory server instance during metric collection.

13.31.6.1 User Action

Specific to your site.

13.31.7 Completed Delete Operations

Shows the total number of LDAP delete operations completed by the Oracle Internet Directory server instance during metric collection.

13.31.7.1 User Action

Specific to your site.

13.31.8 Completed Modify Operations

Shows the total number of LDAP modify operations completed by the Oracle Internet Directory server instance during metric collection.

13.31.8.1 User Action

Specific to your site.

13.31.9 Completed Modrdn Operations

The metric value specifies the number of unsuccessful LDAP modifydn operations performed by a user

13.31.10 Completed Search Operations

Shows the total number of LDAP search operations completed by the Oracle Internet Directory server instance during metric collection.

13.31.10.1 User Action

Specific to your site.

13.31.11 Completed Unbind Operations

The metric value specifies the number of LDAP unbind operations performed by a user

13.31.12 Failed Bind Operations

The metric value specifies the number of unsuccessful LDAP bind operations performed by a user

13.32 LDAP Operations Profile

This metric provides information about LDAP operations completed by Oracle Internet Directory servers.

13.32.1 Completed Abandon Operations

The metric value specifies the number of LDAP abandon operations performed by a user

13.32.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Minutes

13.32.2 Completed Add Operations

Shows the total number of LDAP add operations completed by the Oracle Internet Directory server instance during the metric collection period.

13.32.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	LDAP Add %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.32.3 Completed Bind Operations

This metric provides information about Oracle Internet Directory server login sessions.

13.32.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	LDAP Bind %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.32.4 Completed Compare Operations

Shows the total number of LDAP compare operations completed by the Oracle Internet Directory server instance during the metric collection period.

13.32.4.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	LDAP Compare %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.32.5 Completed Delete Operations

Shows the total number of LDAP delete operations completed by the Oracle Internet Directory server instance during the metric collection period.

13.32.5.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	LDAP Delete %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.32.6 Completed Modify Operations

Shows the total number of LDAP modify operations completed by the Oracle Internet Directory server instance during the metric collection period.

13.32.6.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	LDAP Modify %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.32.7 Completed Modrdrn Operations

The metric value specifies the number of LDAP Modrdrn operations performed by a user

13.32.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Minutes

13.32.8 Total Operations

The metric value specifies the total number of LDAP operations performed by a user

13.32.8.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Minutes

13.32.9 Completed Search Operations

Shows the total number of LDAP search operations completed by the Oracle Internet Directory server instance during the metric collection period.

13.32.9.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	LDAP Search %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.32.10 Completed Unbind Operations

The metric value specifies the number of LDAP unbind operations performed by a user

13.32.10.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 10 Minutes

13.33 LDAP Server Total User Sessions

Provides information about total LDAP user sessions currently established with Oracle Internet Directory servers.

13.33.1 Total Users Sessions

Shows the total number of user sessions currently established with an Oracle Internet Directory server.

13.33.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	LDAP User Sessions %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.33.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.34 Directory Integration Profiles

Provides basic status information about various Directory Integration Platform profiles configured in the system.

13.34.1 Last Execution Time

Shows the time when the profile was last executed by the Directory Integration Platform server.

13.34.1.1 User Action

Specific to your site.

13.34.2 Last Applied Change Number

Shows the last changenumber in Oracle Internet Directory that was successfully synchronized and propagated to the other end.

13.34.2.1 User Action

Specific to your site.

13.34.3 Execution Errors

Shows the last error that was encountered (if any) during the execution of the integration profile.

13.34.3.1 User Action

Specific to your site.

13.34.4 State

Shows the state of the integration profile (ENABLED / DISABLED).

13.34.4.1 User Action

Specific to your site.

13.34.5 Execution Status

Shows the current execution status of the integration profile.

13.34.5.1 User Action

Specific to your site.

13.34.6 Schedule (secs)

Shows the scheduling interval of the integration profiles. The Directory Integration Platform server executes the profiles using this scheduling interval.

13.34.6.1 User Action

Specific to your site.

13.34.7 Type

Shows the type of the integration profile (synchronization or provisioning).

13.34.7.1 User Action

Specific to your site.

13.35 Directory Integration Server

Provides information about various Directory Integration Platform instances running against this Oracle Internet Directory instance.

13.35.1 Downtime Count

Shows the number of times the Directory Integration Platform server went down unexpectedly.

13.35.1.1 User Action

Specific to your site.

13.35.2 Directory Integration Server Host

Shows the machine on which this Directory Integration Platform server is running.

13.35.2.1 User Action

Specific to your site.

13.35.3 Start Time

Shows the time when this Directory Integration Platform server started.

13.35.3.1 User Action

Specific to your site.

13.35.4 LDAP Server

Shows the Oracle Internet Directory server against which this Directory Integration Platform server is running.

13.35.4.1 User Action

Specific to your site.

13.36 Size of Audit Log Purge Queue

Provides information about the size of the Audit Log purge queue

13.36.1 Total Number of Audit Log Objects in Purge Queue

The metric value specifies the number of objects in the audit log purge queue

13.36.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	10000	100000	1	Audit Log Purge Queue size %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.36.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.37 Size of General Statistics Purge Queue

Provides information about the size of the general statistics purge queue.

13.37.1 Total Number of General Statistics Objects in Purge Queue

The metric value specifies the number of objects in the Oracle Internet Directory system statistics purge queue

13.37.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	10000	100000	1	General Stats Purge Queue size %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.37.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.38 Size of Health Statistics Purge Queue

Provides information about the size of the health statistics purge queue

13.38.1 Total Number of Health Statistics Objects in Purge Queue

The metric value specifies the number of objects in the Oracle Internet Directory health statistics purge queue

13.38.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	10000	100000	1	Health Stats Purge Queue size %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.38.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.39 Total Number ChangeLogs in Purge Queue

Provides information about the number of objects in local change log purge queue.

13.39.1 Total Number of Local Changelogs in Purge Queue

The metric value specifies the number of objects in local change log purge queue

13.39.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	10000	100000	1	Local Change Log count %value% in Purge Queue crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.39.1.2 User Action

If this metric is very high, it might indicate improper purge scheduling. Remove all the change logs that need not be retained. Consider modifying `orclpurgetargetage` and `orclpurgeinterval`. (ref tuning chapter)

For more information see the tuning chapter and the replication attributes chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.40 Total Number Remote ChangeLogs in Purge Queue

Provides information about the number of objects in remote change log purge queue

13.40.1 Total Number of Remote Changelogs in Purge Queue

The metric value specifies the number of objects in remote change log purge queue

13.40.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	10000	100000	1	Remote Change Log count %value% in Purge Queue crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.40.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.41 Total Number of HIQ ChangeLogs

Provides information about the total number of change logs in the Human Intervention Queue.

13.41.1 HIQ Changelog Count

The Metric value specifies the total number of change logs in the Human Intervention Queue.

13.41.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total Change Logs in HIQ %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.41.1.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each "Supplier of changelog" object.

If warning or critical threshold values are currently set for any "Supplier of changelog" 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 "Supplier of changelog" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

13.41.1.3 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.42 Replication Server Configuration Set Information

Provides information about the Replication Server configuration set information metrics.

13.42.1 Number of Threads per Supplier for Change Processing

The number of threads per supplier for change processing.

13.42.1.1 User Action

If this metric increases suddenly, it might be due to an increased load on the replication server due to new change logs to transport or apply. Consider limiting the number of threads per supplier, using one of these methods:

1. If you want to use autotuning, set replautotune to 1. If orclreplicamaxthreads is set to more than 20, reduce it to 10/20. This will limit the total number of threads used for change processing and transporting changes.

2. If you do not want to use autotuning, set replautotune to 0. Set Orclthreadspersupplier;apply and Orclthreadspersupplier;Transport to desired number and restart replication server. This will keep the number of threads used for transport and the number used for change processing constant.

For more information see the tuning chapter and the replication attributes chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.42.2 Number of Threads per Supplier for Transporting Changes

The number of threads per supplier for transporting changes.

13.42.2.1 User Action

If this metric increases suddenly, it might be due to an increased load on the replication server due to new change logs to transport or apply. Consider limiting the number of threads per supplier, using one of these methods:

1. If you want to use autotuning, set `replautotune` to 1. If `orclreplicamaxthreads` is set to more than 20, reduce it to 10/20. This will limit the total number of threads used for change processing and transporting changes.

2. If you do not want to use autotuning, set `replautotune` to 0. Set `Orclthreadspersupplier;apply` and `Orclthreadspersupplier;Transport` to desired number and restart replication server. This will keep the number of threads used for transport and the number used for change processing constant.

For more information see the tuning chapter and the replication attributes chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.43 Running instances of LDAP Replication Server

Provides information about running instances of the directory replication server.

13.43.1 Downtime Count

Shows the number of times the instance went down and was restarted by the Oracle Internet Directory monitor daemon.

13.43.1.1 User Action

Specific to your site.

13.43.2 Oracle Directory Server

Shows the number of times the Oracle Internet Directory instance went down and was restarted by the Oracle Internet Directory monitor daemon.

13.43.2.1 User Action

None.

13.43.3 Start Time

Shows the time when the instance of the directory replication server was first started.

13.43.3.1 User Action

None.

13.44 Total Number of Local ChangeLogs

Provides information about the total number of local change logs generated.

13.44.1 Total Number of Local Changelogs

This metric value specifies the total number of local change logs generated.

13.44.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total Local Change Logs %value% crossed warning (%warning_threshold%) or critical (%critical_threshold)

13.44.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.45 Total Number of New ChangeLogs

Provides information about the total number of change logs entries created.

13.45.1 New Changelog Count

This count of this metric gives the total number of change logs entries created.

13.45.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total New Change Logs %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.45.1.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each "Supplier of changelog" object.

If warning or critical threshold values are currently set for any "Supplier of changelog" 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 "Supplier of changelog" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

13.45.1.3 User Action

If this metric is increasing at a high rate, it might indicate:

1. The number of transport threads per Supplier is set very high and the replication server is not running, so change logs are not being processed. Consider setting `orclreplicmaxthreads` lower (if using autotuning) or setting both `Orclthreadspersupplier;apply` and `Orclthreadspersupplier;Transport` lower (if not using autotuning). Ensure that the replication server is running.
2. `Orclszelimit`, which determines the number of changes transported from supplier to consumer, is set too high, allowing faster new change log growth. Consider setting `orclszelimit` lower. Ensure that the number of replication apply threads is adequate to handle a large volume of changes. Consider setting `orclchangeretrycount` to a lower value to provide more resource to change log processing. Keep in mind that reducing the value of `orclchangeretrycount` might increase HIQ.

For more information see the tuning chapter and the replication attributes chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.46 Total number ChangeLogs to be processed

Provides information about the number of change logs that pending.

13.46.1 Total Changelogs to be processed

This metric value specifies the number of change logs that pending.

13.46.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	5000	20000	1	Replication backlog. %value% changelog not yet processed, crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.46.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.47 Total Number of Retry ChangeLogs

Provides information about the number of change logs in the retry queue.

13.47.1 Retry Changelog Count

This metric value specifies the number of change logs in the retry queue.

13.47.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Total Change Logs in Retry Queue %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.47.1.2 Multiple Thresholds

For this metric you can set different warning and critical threshold values for each "Supplier of changelog" object.

If warning or critical threshold values are currently set for any "Supplier of changelog" 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 "Supplier of changelog" object, use the Edit Thresholds page. See Editing Thresholds for information on accessing the Edit Thresholds page.

13.47.1.3 User Action

If this metric is very large, analyze the retry logs to see the reason for conflict and try to resolve it. Make sure orclconfresolution is set to 1. If the same change logs have been retried repeatedly, consider whether orclchangeretrycount is set appropriately.

For more information see the tuning chapter and the replication attributes chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.48 OID Replication Server Virtual Memory Size

Provides information about the total memory used by Oracle Internet Directory Replication Server.

13.48.1 Replication Server Memory Size

The metric value specifies the total memory used by Oracle Internet Directory Replication Server

13.48.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Oracle Directory Replication Server Memory Usage %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%)

13.48.1.2 User Action

If this metric is very high during normal replication, it might indicate:

1. There are many change logs to apply and updateschedule is set to 0 (immediate processing). Consider increasing updateschedule to 1 or 2.
2. There are many change logs in HIQ and orclhiqschedule is set to a short time. Consider increasing orclhiqschedule to 10-15 minutes.

If this metric is high during bootstrapping, it might indicate:

1. The attributes orclreplmaxworkerthread, orclthreadspersupplier;apply, and orclthreadspersupplier;transport are set inappropriately high. Consider decreasing these values. Also consider setting replautotune to 1.
2. The consumer is a bottleneck and you are bootstrapping an entry larger than 50K. Consider using bulkload instead of bootstrapping. Stop replication server on the consumer before the bulkload and restart when finished.

For more information see the tuning chapter and the replication attributes chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.49 OID Replication Server Percentage CPU

Provides information about the percentage of CPU used by Oracle Internet Directory Replication Server.

13.49.1 Replication Server Percentage CPU

Provides information about the percentage of CPU used by Oracle Internet Directory Replication Server.

13.49.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Oracle Directory Replication Server CPU Usage %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%)

13.49.1.2 User Action

If this metric is very high during normal replication, it might indicate:

1. There are many change logs to apply and updateschedule is set to 0 (immediate processing). Consider increasing updateschedule to 1 or 2.
2. There are many change logs in HIQ and orclhiqschedule is set to a short time. Consider increasing orclhiqschedule to 10-15 minutes.

If this metric is high during bootstrapping, it might indicate:

1. The attributes orclreplmaxworkerthread, orclthreadspersupplier;apply, and orclthreadspersupplier;transport are set inappropriately high. Consider decreasing these values. Also consider setting replautotune to 1.
2. The consumer is a bottleneck and you are bootstrapping an entry larger than 50K. Consider using bulkload instead of bootstrapping. Stop replication server on the consumer before the bulkload and restart when finished.

For more information see the tuning chapter and the replication attributes chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.50 Replication Supplier Details

Provides information about the replication supplier detail metrics.

13.50.1 Replication Agreement Type

The replication agreement type: multimaster, one-way, or two-way replication agreement.

13.50.2 HIQ Changelog Count

The Metric value specifies the total number of change logs in the Human Intervention Queue

13.50.2.1 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

If this metric is very large, it might indicate:

1. Change logs are not being retried a sufficient number of times. Consider increasing `orclchangeretrycount` to 4-10.
 2. HIQ change log processing is too slow because `orclhiqschedule` is set too high. Consider setting `orclhiqschedule` lower.
 3. Many conflicts occurred while applying change logs. Analyze the change logs. If appropriate, use the compare and reconcile tool to check the difference between directories. Move the change logs to the purge queue or retry queue, as appropriate.
- For more information see the tuning chapter and the replication attributes chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.50.3 New Changelog Count

This metric gives the total number of change logs entries created.

13.50.3.1 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.50.4 Retry Changelog Count

This metric value specifies the number of change logs in the retry queue.

13.50.4.1 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.50.5 Name of the Supplier Replica

The replica ID of the supplier

13.51 Size of Security Refresh events Purge Queue

Provides information about the number of objects in the Oracle Internet Directory Security Refresh event purge queue.

13.51.1 Total Number of Security Refresh events Objects in Purge Queue

The metric value specifies the number of objects in the Oracle Internet Directory Security Refresh event purge queue.

13.51.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	10000	100000	1	Security Refresh events Purge Queue size %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%)

13.51.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.52 Historical LDAP Entry Cache Hit Ratio

Provides information about performance of the entry cache in Oracle Internet Directory servers.

13.52.1 User Action

The entry cache is enabled only in single-system environments. When the cache is enabled, this metric should be close to 100% during searches. For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.52.2 Date and Time Stamp

Shows the date and time stamp (in UTC) of the metric collection.

13.52.2.1 User Action

Specific to your site.

13.52.3 Instance Number

Shows the instance number of the Oracle Internet Directory server instance for which this metric is collected.

13.52.3.1 User Action

Specific to your site.

13.52.4 Server Entry Cache Hit Ratio

Shows in percentage the entry cache hit ratio in Oracle Internet Directory server at the end of the metric collection period.

13.52.4.1 User Action

The entry cache is enabled only in single-system environments. When the cache is enabled, this metric should be close to 100% during searches. For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.52.5 Process ID

Shows the process identifier of the Oracle Internet Directory server instance for which this metric is collected.

13.52.5.1 User Action

Specific to your site.

13.52.6 LDAP Entry Cache Hit Ratio

Provides information about performance of the entry cache in Oracle Internet Directory servers.

13.52.6.1 User Action

The entry cache is enabled only in single-system environments. When the cache is enabled, this metric should be close to 100% during searches. For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.52.7 Server Entry Cache Hit Ratio

Shows in percentage the entry cache hit ratio in Oracle Internet Directory server at the end of the metric collection period.

13.52.7.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample		Not Defined	Not Defined	1	Server Entry Cache Hit %value% less than warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.52.7.2 User Action

The entry cache is enabled only in single-system environments. When the cache is enabled, this metric should be close to 100% during searches. For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.53 OID Server Virtual Memory Size

Provides information about memory used by Oracle Internet Directory Server.

13.53.1 OID Server Memory Size

The metric value specifies the total memory used by Oracle Internet Directory Server.

13.53.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Oracle Directory Server Memory size %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.54 Stopped instances of LDAP Server

Provides information about Oracle Internet Directory instances that were started earlier and are stopped now.

13.54.1 Config Set Number

Shows the configuration set associated with the stopped instance.

13.54.1.1 User Action

None.

13.54.2 Host Name

Shows the host name of the stopped instance.

13.54.2.1 User Action

None.

13.55 OID Server Percentage CPU Utilization

Provides information about the percentage of CPU used by Oracle Internet Directory servers.

13.55.1 OID Server Percentage CPU

Provides information about the percentage of CPU used by Oracle Internet Directory servers.

13.55.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every Sample	>	Not Defined	Not Defined	1	Oracle Directory Server CPU Usage %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.56 (Historical Critical Events)Security Events

This metric provides information about security events that occurred in Oracle Internet Directory servers.

13.56.1 Event Time

Shows the date and time stamp (in UTC) of the event occurrence.

13.56.1.1 User Action

Specific to your site.

13.56.2 Event Result

Shows the result--that is, either success or failure--of the event that occurred.

13.56.2.1 User Action

Specific to your site.

13.56.3 Event Name

Shows the name of the event that occurred.

13.56.3.1 User Action

Specific to your site.

13.56.4 Instance Number

Shows the instance number of the Oracle Internet Directory server instance for which this metric is collected.

13.56.4.1 User Action

Specific to your site.

13.56.5 Process ID

Shows the process identifier for the Oracle Internet Directory server instance for which this metric is collected.

13.56.5.1 User Action

Specific to your site.

13.56.6 User DN

Shows the identity of the user who logged into Oracle Internet Directory server and performed the operation that triggered the event.

13.56.6.1 User Action

Specific to your site.

13.57 (Critical Event)Super User Failed Logins

This metric provides information about failed super user logins that occurred in Oracle Internet Directory servers.

13.57.1 Failed LDAP Super User Login

The number of successful super user logins that occurred in Oracle Internet Directory servers.

13.57.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding

Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	15	20	1	Superuser Login failure count %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%)

13.57.1.2 User Action

A high number of failed superuser login attempts might indicate a break-in attempt.

13.58 (Critical Event)Super User Successful Logins

This metric provides information about successful super user logins that occurred in Oracle Internet Directory servers.

13.58.1 Successful LDAP Super User Login

The number of successful super user logins that occurred in Oracle Internet Directory servers.

13.58.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	40	50	1	Superuser Login Success count %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%)

13.59 Size of System Resource events Purge Queue

Provides information about the number of objects in the Oracle Internet Directory System resource event purge queue.

13.59.1 Total Number of System Resource events Objects in Purge Queue

The metric value specifies the number of objects in the Oracle Internet Directory System resource event purge queue.

13.59.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	10000	100000	1	System Resource events Purge Queue size %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%) threshold

13.59.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.60 Size of Tombstone Purge Queue

13.60.1 Total Number of Tombstone Objects in Purge Queue

The metric value specifies the number of objects in the tombstone purge queue

13.60.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 10 Minutes	After Every Sample	>	10000	100000	1	Tombstone Purge Queue size %value% crossed warning (%warning_threshold%) or critical (%critical_threshold%)

13.60.1.2 User Action

For information about tuning Oracle Internet Directory, see the tuning chapter in Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.

13.61 TopProcesses

Provides information about the name of the process.

13.61.1 Process Name

The name of this process.

13.62 Resource Usage

Provides information about resource usage metrics.

13.62.1 CPU Usage (%)

Provides information about the percentage of CPU used by Oracle Internet Directory servers.

13.62.2 CPU Idle Time (%)

Percent CPU Idle time on the system where Oracle Internet Directory is running.

13.62.3 Other CPU Usage (%)

Percentage of CPU used by processes other than Oracle Internet Directory.

13.62.4 Memory Usage (MB)

Provides information about total virtual memory used by Oracle Internet Directory servers.

13.62.5 Free Memory (MB)

The metric value shows the system free memory where Oracle Internet Directory server is running.

13.62.6 Other Memory Usage (MB)

Amount of memory used by processes other than Oracle Internet Directory.

13.62.7 Total Memory (MB)

Metric value represents the total memory as used by Oracle Internet Directory Server.

13.62.8 Memory Usage (%)

Provides information about percent total virtual memory used by Oracle Internet Directory servers.

13.62.9 Free Memory (%)

The metric value shows the percentage of system free memory where Oracle Internet Directory server is running.

13.62.10 Other Memory Usage (%)

Percent of memory used by processes other than Oracle Internet Directory.

13.62.11 Start Time (ms since Epoch)

The start time of the Oracle Internet Directory process.

13.63 Response

For Oracle internal use only.

13.63.1 Status

Shows whether any Oracle Internet Directory LDAP server is up or not.

13.63.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every Sample	=	Not Defined	0	1	The Internet Directory is down

13.63.1.2 User Action

Specific to your deployment.

13.63.2 Total Number

Shows the total number of Oracle Internet Directory servers that are up and running.

13.63.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x	Every 5 Minutes

13.63.2.2 User Action

None.

13.64 User LDAP Operations Stats

Provides information about user LDAP operations stats metrics.

13.64.1 Abandon Operations

The metric value specifies the number of LDAP abandon operations performed by a user

13.64.2 Failed Add Operations

The metric value specifies the number of unsuccessful LDAP add operations performed by a user.

13.64.3 Failed Base Search Operations

The metric value specifies the number of unsuccessful LDAP base search operations performed by a user

13.64.4 Failed Bind Operations

The metric value specifies the number of unsuccessful LDAP bind operations performed by a user

13.64.5 Failed Compare Operations

The metric value specifies the number of unsuccessful LDAP compare operations performed by a user

13.64.6 Failed Delete Operations

The metric value specifies the number of unsuccessful LDAP delete operations performed by a user.

13.64.7 Failed Modify Operations

The metric value specifies the number of unsuccessful LDAP modify operations performed by a user.

13.64.8 Failed ModRdn Operations

The metric value specifies the number of unsuccessful LDAP ModRdn operations performed by a user.

13.64.9 Failed OneLevel Search Operations

The metric value specifies the number of unsuccessful LDAP OneLevel Search operations performed by a user.

13.64.10 Failed Proxy Bind Operations

The metric value specifies the number of unsuccessful LDAP proxy bind operations performed by a user

13.64.11 Failed Subtree Search Operations

The metric value specifies the number of unsuccessful LDAP Subtree Search operations performed by a user.

13.64.12 Successful Add Operations

The metric value specifies the number of successful LDAP add operations performed by a user.

13.64.13 Successful Base Search Operations

The metric value specifies the number of successful LDAP base search operations performed by a user

13.64.14 Successful Bind Operations

The metric value specifies the number of successful LDAP bind operations performed by a user

13.64.15 Successful Compare Operations

The metric value specifies the number of successful LDAP compare operations performed by a user

13.64.16 Successful Delete Operations

The metric value specifies the number of successful LDAP delete operations performed by a user.

13.64.17 Successful Modify Operations

The metric value specifies the number of successful LDAP modify operations performed by a user.

13.64.18 Successful ModRdn Operations

The metric value specifies the number of successful LDAP ModRdn operations performed by a user.

13.64.19 Successful OneLevel Search Operations

The metric value specifies the number of successful LDAP OneLevel Search operations performed by a user.

13.64.20 Successful Proxy Bind Operations

The metric value specifies the number of successful LDAP proxy bind operations performed by a user

13.64.21 Successful Subtree Search Operations

The metric value specifies the number of successful LDAP Subtree Search operations performed by a user.

13.64.22 Unbind Operations

The metric value specifies the number of LDAP unbind operations performed by a user

Directory Integration Platform

This chapter describes Oracle Directory Integration Platform (DIP) metrics.

14.1 DIP Aggregated Metrics

The aggregation of all the values across profile executions

14.1.1 Total Changes Attempted

This metric specifies the total number of changes attempted across all profiles.

14.1.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.1.1.2 User Action

No action required.

14.1.2 Total Successful Changes

This metric specifies the total number of successful changes across all profiles.

14.1.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.1.2.2 User Action

No action required.

14.1.3 Average LDAP Search Time

This metric specifies the average LDAP search time across all profiles.

14.1.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.1.3.2 User Action

No action required.

14.1.4 Average LDAP Write Time

This metric specifies the average LDAP write time across all profiles.

14.1.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.1.4.2 User Action

No action required.

14.1.5 Average Profile Execution Time

This metric specifies the average profile execution time across all profiles.

14.1.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.1.5.2 User Action

No action required.

14.2 DIP All Profiles (Consolidated)

This metric provides a consolidated list of profiles that are currently enabled in DIP

14.2.1 Host

This metric specifies the host where the DIP application is running.

14.2.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.2.1.2 User Action

No action required.

14.2.2 Last Change Number

This metric specifies the last change number that was processed by the profile during the previous execution.

14.2.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.2.2.2 User Action

No action required.

14.2.3 Last Execution Error

This metric specifies the error when the profile was last executed.

14.2.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.2.3.2 User Action

No action required.

14.2.4 Last Execution Time

This metric specifies the time when the profile was last executed.

14.2.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.2.4.2 User Action

No action required.

14.2.5 Last Successful Execution Time

This metric specifies the time when the profile was last executed successfully.

14.2.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.2.5.2 User Action

No action required.

14.2.6 Status of Last Sync Operation

This metric specifies the status of the last operation during profile execution.

14.2.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.2.6.2 User Action

No action required.

14.2.7 DIP Profile Name

This metric specifies the name of the profile.

14.2.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.2.7.2 User Action

No action required.

14.2.8 DIP Process

This metric specifies the DIP process name and ID.

14.2.8.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.2.8.2 User Action

No action required.

14.2.9 Average Processing Time

This metric specifies the average processing time (in milliseconds) of the profile.

14.2.9.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.2.9.2 User Action

No action required.

14.2.10 Maximum Processing Time

This metric specifies the largest amount of time (in milliseconds) taken to process the profile.

14.2.10.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.2.10.2 User Action

No action required.

14.2.11 Minimum Processing Time

This metric specifies the least amount of time (in milliseconds) taken to process the profile.

14.2.11.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.2.11.2 User Action

No action required.

14.2.12 Profile Job Status

This metric specifies the status of the quartz job related to this profile.

14.2.12.1 User Action

Check if the profile is scheduled, then check if the profile is enabled.

14.3 DIP Prov profile runtime operational metrics

This metric provides detailed execution statistics for all the currently enabled provisioning profiles configured in DIP

14.3.1 Failed Add Group operations

This metric specifies the number of failed add group operations.

14.3.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.1.2 User Action

No action required.

14.3.2 Failed Add Identity operations

This metric specifies the number of failed add identity operations.

14.3.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.2.2 User Action

No action required.

14.3.3 Failed Add Subscription operations

This metric specifies the number of failed add subscription operations.

14.3.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.3.2 User Action

No action required.

14.3.4 Failed Add User operations

This metric specifies the number of failed add user operations.

14.3.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.4.2 User Action

No action required.

14.3.5 Failed Delete Group operations

This metric specifies the number of failed delete group operations.

14.3.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.5.2 User Action

No action required.

14.3.6 Failed Delete Identity operations

This metric specifies the number of failed delete identity operations.

14.3.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.6.2 User Action

No action required.

14.3.7 Failed Delete Subscription operations

This metric specifies the number of failed delete subscription operations.

14.3.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.7.2 User Action

No action required.

14.3.8 Failed Delete User operations

This metric specifies the number of failed delete user operations.

14.3.8.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.8.2 User Action

No action required.

14.3.9 Failed Modify Group operations

This metric specifies the number of failed modify group operations.

14.3.9.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.9.2 User Action

No action required.

14.3.10 Failed Modify Identity operations

This metric specifies the number of failed modify identity operations.

14.3.10.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.10.2 User Action

No action required.

14.3.11 Failed Modify User operations

This metric specifies the number of failed modify user operations.

14.3.11.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.11.2 User Action

No action required.

14.3.12 Profile Name

This metric specifies the provisioning profile name.

14.3.12.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.12.2 User Action

No action required.

14.3.13 Successful Add Group operations

This metric specifies the number of successful add group operations.

14.3.13.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.13.2 User Action

No action required.

14.3.14 Successful Add Identity operations

This metric specifies the number of successful add identity operations.

14.3.14.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.14.2 User Action

No action required.

14.3.15 Successful Add Subscription operations

This metric specifies the number of successful add subscription operations.

14.3.15.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.15.2 User Action

No action required.

14.3.16 Successful Add User operations

This metric specifies the number of successful add user operations.

14.3.16.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.16.2 User Action

No action required.

14.3.17 Successful Delete Group operations

This metric specifies the number of successful delete group operations.

14.3.17.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.17.2 User Action

No action required.

14.3.18 Successful Delete Identity operations

This metric specifies the number of successful delete identity operations.

14.3.18.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.18.2 User Action

No action required.

14.3.19 Successful Delete Subscription operations

This metric specifies the number of successful delete subscription operations.

14.3.19.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.19.2 User Action

No action required.

14.3.20 Successful Delete User operations

This metric specifies the number of successful delete user operations.

14.3.20.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.20.2 User Action

No action required.

14.3.21 Successful Modify Group operations

This metric specifies the number of successful modify group operations.

14.3.21.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.21.2 User Action

No action required.

14.3.22 Successful Modify Identity operations

This metric specifies the number of successful modify identity operations.

14.3.22.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.22.2 User Action

No action required.

14.3.23 Successful Modify User operations

This metric specifies the number of successful modify user operations.

14.3.23.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.3.23.2 User Action

No action required.

14.4 DIP Sync profile runtime operational metrics

This metric provides detailed execution statistics for all the currently enabled synchronization profiles configured in DIP

14.4.1 Skipped Operations

This metric specifies the number of skipped operations.

14.4.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.4.1.2 User Action

No action required.

14.4.2 Failed Add Operations

This metric specifies the number of failed add operations.

14.4.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.4.2.2 User Action

No action required.

14.4.3 Failed Delete Operations

This metric specifies the number of failed delete operations.

14.4.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.4.3.2 User Action

No action required.

14.4.4 Failed Modify Operations

This metric specifies the number of failed modify operations.

14.4.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.4.4.2 User Action

No action required.

14.4.5 Profile Name

This metric specifies the synchronization profile name.

14.4.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.4.5.2 User Action

No action required.

14.4.6 Successful Add Operations

This metric specifies the number of successful add operations.

14.4.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.4.6.2 User Action

No action required.

14.4.7 Successful Delete Operations

This metric specifies the number of successful delete operations.

14.4.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.4.7.2 User Action

No action required.

14.4.8 Successful Modify Operations

This metric specifies the number of successful modify operations.

14.4.8.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.4.8.2 User Action

No action required.

14.5 DIP Health

This metric provides information on the functional health of the DIP server

14.5.1 Host

This metric specifies the host where the WebLogic server is running.

14.5.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.5.1.2 User Action

No action required.

14.5.2 MBean Status

This metric specifies the status of the DIP JMX mbeans.

14.5.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.5.2.2 User Action

Check if the DIP application has started in the Weblogic server.

14.5.3 Name

This metric specifies the internal name of the DIP application.

14.5.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.5.3.2 User Action

No action required.

14.5.4 Quartz Scheduler Status

This metric specifies the status of the internal quartz scheduler.

14.5.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.5.4.2 User Action

Check if the DIP application has started in the Weblogic server.

14.5.5 WLS Server Name

This metric specifies the WebLogic server name.

14.5.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.5.5.2 User Action

No action required.

14.6 DIP Provisioning Profiles

This metric provides the list of currently enabled Provisioning profiles

14.6.1 DIP Provisioning Profile Name

This metric specifies the name of the provisioning profile.

14.6.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.6.1.2 User Action

No action required.

14.6.2 Host

This metric specifies the host where the DIP application is running.

14.6.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.6.2.2 User Action

No action required.

14.6.3 Name

This metric specifies the name of the profile.

14.6.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.6.3.2 User Action

No action required.

14.7 DIP Synchronization Profiles

This metric provides the list of currently enabled synchronization profiles

14.7.1 DIP Synchronization Profile Name

This metric specifies the name of the synchronization profile.

14.7.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.7.1.2 User Action

No action required.

14.7.2 Host

This metric specifies the host where the DIP application is running.

14.7.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.7.2.2 User Action

No action required.

14.7.3 Name

This metric specifies the name of the profile.

14.7.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
11Plus	Every 15 Minutes

14.7.3.2 User Action

No action required.

14.8 Resource Usage

Provides information about resource usage metrics.

14.8.1 CPU Usage (%)

14.8.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
11Plus	Every 15 Minutes	After Every Sample	>	75	90	1	The CPU utilization is greater than %value%

14.8.1.2 User Action

No action required.

14.8.2 Memory Usage (%)

14.8.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
11Plus	Every 15 Minutes	After Every Sample	>	75	90	1	The Memory utilization is greater than %value%

14.8.2.2 User Action

No action required.

14.9 DIP server status

Provides information about the DIP server status.

14.9.1 Status

14.9.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every Sample	=	Not Defined	0	1	The DIP Server Instance is down

14.9.1.2 User Action

Change the status of the Directory Integration Platform instance by selecting Control from the DIP Server menu, and then selecting either Start Up or Shut Down.

This chapter describes Oracle WebCenter metrics.

15.1 Response

This metric indicates the current status of the WebCenter application:

Up (Green Up Arrow) - The WebCenter application is up and running and the last operation was successful.

Down (Red Down Arrow) - The WebCenter application is not currently available. The last operation was unsuccessful due to an unexpected error or exception. User errors, such as an authentication failure, do not change the status to Down.

Unknown (Clock) - Unable to query the status of the WebCenter application for some reason.

15.1.1 UpDown Status

This metric indicates the current status of the WebCenter application:

Up (Green Up Arrow) - The WebCenter application is up and running and the last operation was successful.

Down (Red Down Arrow) - The WebCenter application is not currently available. The last operation was unsuccessful due to an unexpected error or exception. User errors, such as an authentication failure, do not change the status to 'Down'.

Unknown (Clock) - Unable to query the status of the WebCenter application for some reason.

15.1.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every 60 Samples	=	Not Defined	0	1	The WebCenter Application is down

15.1.1.2 User Action

If you are having problems with the WebCenter application and the status is 'Down' or 'Unavailable', check the diagnostic logs to establish why the application is not up and running. The WebCenter application, and maybe the WebLogic server, must be restarted.

15.2 WebCenter Spaces (Aggregated)

Provides information about the WebCenter Spaces (Aggregated) metrics

15.2.1 Application Name

This metric identifies the WebCenter application to which these group space metrics apply.

15.2.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.2.1.2 User Action

No action required.

15.2.2 Successful Invocations (%)

This metric indicates the percentage of group space invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count. *All* group spaces in the WebCenter application are included in this count. Processing time is available on application's main metric page.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.2.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.2.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric further. Navigate to the application's group space metric page where more detailed metric data is available. See, [Monitoring WebCenter Spaces Information](#).

See also, [Analyzing Group Space Metrics](#).

You can also check the diagnostic logs to establish why group space invocations are failing.

15.2.3 Invocations (per minute)

This metric indicates the number of group space invocations per minute.

All group spaces in the WebCenter application are included in this count. The number of invocations per group space is available on application's group space metric page. See, [Monitoring WebCenter Spaces Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.2.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.2.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for group spaces is high (compared with other WebCenter services), consider prioritizing troubleshooting efforts in this area.

See also, [Analyzing Group Space Metrics](#).

15.2.4 Page Throughput (per minute)

This metric shows the average number of pages accessed per minute.

All group spaces in the WebCenter application are included in this count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.2.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.2.4.2 User Action

Use this metric to monitor typical and peak usage of group spaces. You can evaluate when pages are accessed the most, as well as the maximum and minimum usage.

See also, *Analyzing Group Space Metrics*.

15.2.5 Average Page Processing Time (ms)

This metric indicates the average time (in ms) spent processing group space operations.

All group spaces in the WebCenter application are included in the count. Processing times for individual group spaces are available on application's main metric page. See, *Monitoring WebCenter Spaces Information*.

The last 10-15 minutes of data is used to calculate the average page processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.2.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.2.5.2 User Action

Use this metric to determine the performance of group space operations.

If this metric is out-of-bounds (the average time for group space operations is increasing or higher than expected), navigate to the application's group space metric page where more detailed metric data for individual group spaces is available.

See also, *Analyzing Group Space Metrics*.

15.2.6 Service Name

These metrics apply to the WebCenter group space.

15.2.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.2.6.2 User Action

No action required.

15.3 WebCenter Content Repositories (Aggregated)

Provides information about the WebCenter Content Repositories (Aggregated) metrics.

15.3.1 Application Name

This metric identifies the WebCenter application to which these content repository metrics apply.

15.3.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.3.1.2 User Action

No action required.

15.3.2 Successful Invocations (%)

This metric indicates the percentage of Documents service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count. *All* the content repositories used by the WebCenter application are included in this count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.3.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.3.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data for individual content repositories is also available. See, [Viewing Performance Information](#).

See also, [Analyzing Content Repository Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.3.3 Invocations (per minute)

This metric indicates the total number of Documents service invocations per minute. All the content repositories used by this WebCenter application are included in this count.

Invocation rates and invocation counts for individual content repositories are available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.3.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.3.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Documents service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Content Repository Metrics](#).

15.3.4 Average Login Time (ms)

This metric indicates the average time spent logging in to content repositories (in ms).

All the content repositories used by the WebCenter application are included in this count.

The last 10-15 minutes of data is used to calculate the average login time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.3.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.3.4.2 User Action

Use this metric to determine the performance of the Login operation. If this metric is out-of-bounds (the average time to login is increasing or higher than expected), navigate to the applications main metric page where more detailed metric data for the Login operation (and all other operations) is available. Use the information on this page to help determine the underlying cause of the time delay. See, Viewing Performance Information.

See also, Analyzing Content Repository Metrics.

15.3.5 Logins (per minute)

This metric counts the number of times a WebCenter user logs in to a content repository per minute.

All the content repositories used by the WebCenter application are included in this count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.3.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.3.5.2 User Action

Use this metric to monitor typical and peak usage of content repositories. You can evaluate when most logins occur, as well as the maximum and minimum usage--which can be useful for tuning content repository performance.

If you are experiencing performance issues outside normal peak login times, navigate to the application's main metric page where detailed metric data for all operations is available. See, Viewing Performance Information.

See also, Analyzing Content Repository Metrics.

15.3.6 Average Processing Time (ms)

This metric indicates the average time the Documents service spends processing content repository operations (in ms).

Content repository operations include: Download, Upload, Search, Login, Delete. Processing times for individual operations are available on the application's main metric page. See, Viewing Performance Information.

All the content repositories used by the WebCenter application are included in this count.

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.3.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.3.6.2 User Action

Use this metric to determine the performance of content repository-related operations.

If this metric is out-of-bounds (the average time for content repository operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See, Viewing Performance Information.

See also, Analyzing Content Repository Metrics.

15.3.7 Service Name

These metrics apply to content repositories used by the Documents service.

15.3.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.3.7.2 User Action

No action required.

15.4 WebCenter Events (Aggregated)

Provides information about WebCenter Events (Aggregated).

15.4.1 Application Name

This metric identifies the WebCenter application to which these Group Space and Personal Events service metrics apply.

15.4.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.4.1.2 User Action

No action required.

15.4.2 Successful Invocations (%)

This metric indicates the percentage of Group Space and Personal Events service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.4.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.4.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric further. Navigate to the application's main metric page where detailed metric data for individual operations is also available. See, [Viewing Performance Information](#).

See also, [Analyzing Events Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.4.3 Invocations (per minute)

This metric indicates the total number of invocations per minute for the Group Space and Personal Events services.

The number of invocations per operation is available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are

detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.4.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.4.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for Group Space and Personal Events services is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Events Metrics](#).

15.4.4 Average Processing Time (ms)

This metric indicates the average time the Group Space Events service and the Personal Events service spends processing events operations (in ms).

Group Space Events and Personal Events service operations include: Create Event, Update Event, List Event, Delete Event, Search Event. Processing times for individual operations are available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.4.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.4.4.2 User Action

Use this metric to determine the performance of Group Space Events and Personal Events service operations. If this metric is out-of-bounds (the average time for Group Space Events and Personal Events operations is increasing or higher than expected),

navigate to the application's main metric page where detailed metric data for individual operations is also available. See, [Viewing Performance Information](#).

See also, [Analyzing Events Metrics](#).

15.4.5 Service Name

These metrics apply to group space events and personal events used by the WebCenter application.

15.4.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.4.5.2 User Action

No action required.

15.5 WebCenter External Applications (Aggregated)

Provides information about the WebCenter External Applications (Aggregated) metrics.

15.5.1 Application Name

This metric identifies the WebCenter application to which these external application metrics apply.

15.5.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.5.1.2 User Action

No action required.

15.5.2 Successful Invocations (%)

This metric indicates the percentage of external application invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count. *All* the external applications accessed by the WebCenter application are included in this count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are

detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.5.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.5.2.2 User Action

If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why service operations are failing. Typical examples are, trying to invoke the automated login servlet without being authenticated or specifying an incorrect external application name. Navigate to the application's main metric page where more detailed metric data for individual external applications is also available. See, Viewing Performance Information.

See also, Analyzing External Application Metrics.

15.5.3 Invocations (per minute)

This metric indicates the total number of external application invocations per minute.

All external applications used by this WebCenter application are included in this count. Invocation rates and invocation counts for individual external applications are available on application's main metric page.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.5.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.5.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate application performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for external applications is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, Viewing Performance Information.

See also, [Analyzing External Application Metrics](#).

15.5.4 Page Throughput (per minute)

This metric shows the average number of pages accessed per minute.

All pages in the WebCenter application are included in this count.

15.5.5 Average Processing Time (ms)

This metric indicates the average time spent processing external application operations (in ms).

External application operations include: Fetch Credentials, Store Credentials, Fetch External Application, and Automated Logins. Processing times for individual operations are available on application's main metric page. See, [Viewing Performance Information](#).

All external applications used by the WebCenter application are included in this count.

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.5.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.5.5.2 User Action

Use this metric to determine the overall external application performance. If this metric is out-of-bounds (the average time for external application operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing External Application Metrics](#).

15.5.6 Service Name

These metrics apply to the External Applications service.

15.5.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.5.6.2 User Action

No action required.

15.6 WebCenter Spaces Pages (Aggregated)

Provides information about the WebCenter Spaces Pages (Aggregated) metrics.

15.6.1 Application Name

This metric identifies the WebCenter application to which these portlet metrics apply.

15.6.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.6.1.2 User Action

No action required.

15.6.2 Successful Invocations (%)

All portlets used in the WebCenter application are included in this count.

This metric indicates the percentage of portlet invocations that succeeded.

Any request that fails will impact this percentage. This includes WebCenter application-related failures such as timeouts and internal errors, as well as remote/server failures such as requests returned with response codes HTTP4xx or HTTP5xx, responses with a bad content type, and SOAP faults, where applicable. Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.6.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.6.2.2 User Action

If Successful Invocations (%) is below 100%, investigate the metric in more detail. Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Metrics](#).

You can also check the diagnostic logs to establish why portlet requests are failing.

15.6.3 Invocations (per minute)

This metric indicates the total number of portlet invocations per minute.

All portlets used in the WebCenter application are included in this count. The number of invocations is available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.6.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.6.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate portlet performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the invocation rate for portlets is high (compared with other WebCenter services), consider prioritizing troubleshooting efforts in this area. Navigate to the application's main metric page where additional metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Metrics](#).

15.6.4 Page Throughput (per minute)

This metric shows the average number of pages accessed per minute.

All pages in the WebCenter application are included in this count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.6.5 Average Page Processing Time (ms)

This metric indicates the average time (in ms) to display the page.

All pages in the WebCenter application are included in the count. Processing times for individual pages are available on application's main metric page. See *Monitoring Page Metrics for Spaces* and *Monitoring Page Metrics for WebCenter Portal Applications*.

The last 10-15 minutes of data is used to calculate the average page processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.6.6 Service Name

These metrics apply to the WebCenter Spaces page.

15.7 WebCenter Portlets (Aggregated)

Provides information about WebCenter Portlets (Aggregated) metrics.

15.7.1 Application Name

This metric identifies the WebCenter application to which these portlet metrics apply.

15.7.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.7.1.2 User Action

No action required.

15.7.2 Successful Invocations (%)

This metric indicates the percentage of portlet invocations that succeeded.

All portlets used in the WebCenter application are included in this count.

Any request that fails will impact this percentage. This includes WebCenter application-related failures such as timeouts and internal errors, as well as remote/server failures such as requests returned with response codes HTTP4xx or HTTP5xx, responses with a bad content type, and SOAP faults, where applicable. Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, *Viewing Performance Information*.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.7.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.7.2.2 User Action

If Successful Invocations (%) is below 100%, investigate the metric in more detail. Navigate to the application's main metric page where more detailed metric data is available. See [Viewing Performance Information](#).

See also [Analyzing Portlet Metrics](#).

You can also check the diagnostic logs to establish why portlet requests are failing.

15.7.3 Average HTTP 2xx Time (ms)

This metric indicates the average time portlets take to process requests returning a HTTP2xx response code (in ms).

All portlets used in the WebCenter application are included in this count.

The last 10-15 minutes of data is used to calculate the average HTTP 2xx time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

See, [Viewing Performance Information](#).

15.7.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.7.3.2 User Action

Use this metric to track overall portlet performance.

If this metric is out-of-bounds, consider prioritizing troubleshooting efforts for the portlets. Navigate to the application's main metric page where additional metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Metrics](#).

15.7.4 Invocations (per minute)

This metric indicates the total number of portlet invocations per minute.

All portlets used in the WebCenter application are included in this count. The number of invocations is available on application's main metric page. See [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.7.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.7.4.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate portlet performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the invocation rate for portlets is high (compared with other WebCenter services), consider prioritizing troubleshooting efforts in this area. Navigate to the application's main metric page where additional metric data is available. See [Viewing Performance Information](#).

See also [Analyzing Portlet Metrics](#).

15.7.5 Average Processing Time (ms)

This metric indicates the average time spent processing portlet operations (in ms).

All portlets used in the WebCenter application are included in this count. Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.7.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.7.5.2 User Action

Use this metric to determine the portlet performance. If this metric is out-of-bounds (the average time for portlet operations is increasing or higher than expected),

navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Metrics](#).

15.7.6 Service Name

These metrics apply to WebCenter portlets.

15.7.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.7.6.2 User Action

No action required.

15.8 WebCenter Producers (Aggregated)

Provides information about WebCenter Producers (Aggregated) metrics.

15.8.1 Application Name

This metric identifies the WebCenter application to which these portlet producer metrics apply.

15.8.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.8.1.2 User Action

No action required.

15.8.2 Successful Invocations (%)

This metric indicates the percentage of producer invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count. *All* portlet producers used in the WebCenter application are included in this count.

Any request that fails will impact this percentage. This includes WebCenter application-related failures such as timeouts and internal errors, as well as remote/server failures such as requests returned with response codes HTTP4xx or HTTP5xx, responses with a bad content type, and SOAP faults, where applicable. Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15

minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.8.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.8.2.2 User Action

If Successful Invocations (%) is below 100%), investigate the metric in more detail. Navigate to the application's main metric page where more detailed metric data is available. See, Viewing Performance Information.

See also, Analyzing Portlet Producer Metrics.

You can also check the diagnostic logs to establish why producer requests are failing.

15.8.3 Average HTTP 2xx Time (ms)

This metric indicates the average time that all portlets used in the WebCenter application take to process requests returning a HTTP2xx response code (in ms).

The last 10-15 minutes of data is used to calculate the average HTTP 2xx time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

See, Viewing Performance Information.

15.8.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.8.3.2 User Action

Use this metric to track the overall performance of producers in the WebCenter application.

If this metric is out-of-bounds, consider prioritizing troubleshooting efforts for the producers. Navigate to the application's main metric page where additional metric data is available. See, Viewing Performance Information.

See, Analyzing Portlet Producer Metrics.

15.8.4 Invocations (per minute)

This metric indicates the total number of producer invocations per minute.

All portlet producers used in the WebCenter application are included in this count. The number of invocations is available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.8.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.8.4.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate producer performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the invocation rate for producers is high (compared with other WebCenter services), consider prioritizing troubleshooting efforts in this area. Navigate to the application's main metric page where additional metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Producer Metrics](#).

15.8.5 Average Processing Time (ms)

This metric indicates the average time spent processing portlet producer operations (in ms).

All portlet producers used in the WebCenter application are included in this count. Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.8.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.8.5.2 User Action

Use this metric to determine the portlet producer performance. If this metric is out-of-bounds (the average time for producer operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also [Analyzing Portlet Producer Metrics](#).

15.8.6 Service Name

These metrics apply to the portlet producers used by the WebCenter application.

15.8.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.8.6.2 User Action

No action required.

15.9 WebCenter Search (Aggregated)

Provides information about WebCenter Search (Aggregated) metrics.

15.9.1 Application Name

This metric identifies the WebCenter application to which these Search service metrics apply.

15.9.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.9.1.2 User Action

No action required.

15.9.2 Successful Invocations (%)

This metric indicates the percentage of Search service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count. *All* WebCenter search sources are included in the count: Announcements, Documents, Discussion Forums, Group Spaces, Group Space Events,

Lists, Notes, Oracle Secure Enterprise Search, Pages, People Connections, Wikis and Blogs, and Tagged Items.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.9.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.9.2.2 User Action

If Successful Invocations (%) is below 100%, investigate the metric in more detail. Navigate to the application's main metric page where more detailed metric data for individual search sources is available. See, [Viewing Performance Information](#).

See also, [Analyzing Search Metrics](#).

You can also check the diagnostic logs to establish why search requests are failing.

15.9.3 Invocations (per minute)

This metric indicates the total number of Search service invocations per minute across *all* WebCenter search sources: Announcements, Documents, Discussion Forums, Group Spaces, Group Space Events, Links, Lists, Notes, Oracle Secure Enterprise Search, Pages, People Connections, Wikis and Blogs, and Tagged Items.

Invocation rates and invocation counts for individual search sources are available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.9.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.9.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Search service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Search Metrics](#).

15.9.4 Average Processing Time (ms)

This metric indicates the average time (in ms) the Search service spends processing search operations.

All WebCenter search sources are included in the count: Announcements, Documents, Discussion Forums, Group Spaces, Group Space Events, Lists, Notes, Oracle Secure Enterprise Search, Pages, People Connections, Wikis and Blogs, and Tagged Items. Processing times for individual search sources are available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.9.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.9.4.2 User Action

Use this metric to determine the performance of Search service operations.

If this metric is out-of-bounds (the average time for search operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual search sources is available. See, [Viewing Performance Information](#).

See also, [Analyzing Search Metrics](#).

15.9.5 Service Name

These metrics apply to the Search service.

15.9.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.9.5.2 User Action

No action required.

15.10 WebCenter Services

Provides information about WebCenter Services metrics.

15.10.1 Application Name

This metric identifies the WebCenter application to which these WebCenter service metrics apply.

15.10.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.10.1.2 User Action

No action required.

15.10.2 Successful Invocations (%)

This metric indicates the percentage of invocations that succeeded for this WebCenter service. Any operation or request that fails will impact this percentage.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.10.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.10.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric. Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Understanding WebCenter Performance Metrics Information](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.10.3 Invocations (per minute)

This metric indicates the total number of invocations per minute for this WebCenter service. The number of invocations is available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.10.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.10.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the invocation rate for this WebCenter service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

See, [Understanding WebCenter Performance Metrics Information](#).

15.10.4 Average Processing Time (ms)

This metric indicates the average time spent processing operations for this WebCenter service (in ms).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.10.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.10.4.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

Use this metric to determine the performance of WebCenter services. If this metric is out-of-bounds (the average time for operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Understanding WebCenter Performance Metrics Information](#).

15.11 WebCenter BPEL Worklists (Aggregated)

Provides information about WebCenter BPEL Worklists (Aggregated) metrics.

15.11.1 Application Name

This metric identifies the WebCenter application to which these Worklists service metrics apply.

15.11.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.11.1.2 User Action

No action required.

15.11.2 Successful Invocations (%)

This metric indicates the percentage of Worklists service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count. *All* the BPEL worklists used by the WebCenter application are included in this count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.11.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.11.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data for individual worklists is also available. See, [Viewing Performance Information](#).

See also, [Analyzing Worklist Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.11.3 Invocations (per minute)

This metric indicates the total number of Worklists service invocations per minute.

All the BPEL worklists used by the WebCenter application are included in this count. Invocation rates and invocation counts for individual worklists are available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.11.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.11.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Worklists service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Worklist Metrics](#).

15.11.4 Average Processing Time (ms)

This metric indicates the average time the Worklists service spends processing operations (in ms).

All the BPEL worklists used by the WebCenter application are included in this count. Processing times for individual worklists are available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.11.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.11.4.2 User Action

Use this metric to determine the performance of Worklists service operations. If this metric is out-of-bounds (the average time for worklist operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Worklist Metrics](#).

15.11.5 Service Name

These metrics apply to the Worklists service.

15.11.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.11.5.2 User Action

No action required.

15.12 WebCenter Portlets

Provides information about WebCenter Portlet metrics.

15.12.1 Application Name

This metric identifies the WebCenter application to which these portlet metrics apply.

15.12.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All versions	Every 300 seconds

15.12.2 Successful Invocations (%)

This metric indicates the percentage of portlet invocations that succeeded.

Any request that fails will impact this percentage. This includes WebCenter application-related failures such as timeouts and internal errors, as well as remote/server failures such as requests returned with response codes HTTP4xx or HTTP5xx, responses with a bad content type, and SOAP faults, where applicable. Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.12.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.12.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

You can also check the diagnostic logs to establish why portlet requests are failing.

15.12.3 Average HTTP 2xx Time (ms)

This metric indicates the average time the portlet takes to process requests returning a HTTP2xx response code (in ms).

The last 10-15 minutes of data is used to calculate the average HTTP 2xx time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

See, [Viewing Performance Information](#).

15.12.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.12.3.2 User Action

Use this metric to detect the most performing portlets. If you use this metric in conjunction with the Invocations (per minute) metric, then you can identify the top performing portlet.

See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Metrics](#).

15.12.4 Invocations (per minute)

This metric indicates the total number of invocations per minute for a portlet. The number of invocations is available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.12.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.12.4.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate portlet performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the invocation rate for the portlet is high (compared with other portlets), consider prioritizing troubleshooting efforts in this area. Navigate to the application's main metric page where additional metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Metrics](#).

15.12.5 Average Processing Time (ms)

This metric indicates the average time spent processing portlet requests (in ms). Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.12.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.12.5.2 User Action

Use this metric to determine portlet performance. If this metric is out-of-bounds (the average time for portlet operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, Viewing Performance Information.

See also, Analyzing Portlet Metrics.

15.12.6 Producer Type

The portlet producer type: Web or WSRP

Web portlet producer - deployed to a J2EE application server, which is often remote and communicates through Simple Object Access Protocol (SOAP) over HTTP.

WSRP portlet producer - Web Services for Remote Portlets (WSRP) is a Web services standard that allows interoperability between a standards enabled container and a WSRP client application.

15.12.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.12.6.2 User Action

No action required.

15.13 WebCenter Portlets (Top 5)

Provides information about the WebCenter Portlets (Top 5) metrics.

15.13.1 Application Name

This metric identifies the WebCenter application to which these portlet metrics apply.

15.13.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.13.1.2 User Action

No action required.

15.13.2 Successful Invocations (%)

This metric indicates the percentage of portlet invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count. Any request that fails will impact this percentage. This includes WebCenter application-related failures such as timeouts and internal errors, as well as remote/server failures such as requests returned with response codes HTTP4xx or HTTP5xx, responses with a bad content type, and SOAP faults, where applicable. Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

This portlet is one of the top five, most popular portlets in this WebCenter application.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.13.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.13.2.2 User Action

Successful Invocations (%) is below 100%, investigate the metric in more detail. Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Metrics](#).

You can also check the diagnostic logs to establish why portlet requests are failing.

15.13.3 Average HTTP 2xx Time (ms)

This metric indicates the average time the portlet takes to process requests returning a HTTP2xx response code (in ms).

This portlet is one of the top five, most popular portlets in this WebCenter application.

The last 10-15 minutes of data is used to calculate the average HTTP 2xx time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

See, Viewing Performance Information.

15.13.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.13.3.2 User Action

Use this metric to track the performance of popular portlets.

If this metric is out-of-bounds, consider prioritizing troubleshooting efforts for the most popular portlets. Navigate to the application's main metric page where additional metric data is available. See, Viewing Performance Information.

See also, Analyzing Portlet Metrics.

15.13.4 Invocations (per minute)

This metric indicates the total number of invocations per minute for the portlet.

This portlet is one of the top five, most popular portlets in this WebCenter application. The number of invocations is available on application's main metric page. See, Viewing Performance Information.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.13.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.13.4.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate portlet performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the invocation rate for the portlet is high (compared with other portlets), start by troubleshooting this portlet.

Navigate to the application's main metric page where additional metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Metrics](#).

15.13.5 Average Processing Time (ms)

This metric indicates the average time (in ms) spent processing operations for portlet.

This portlet is one of the top five, most popular portlets in this WebCenter application. Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.13.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.13.5.2 User Action

Use this metric to determine the performance of popular portlets.

If this metric is out-of-bounds (the average time for portlet operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Metrics](#).

15.13.6 Producer Type

The portlet producer type: Web or WSRP

Web portlet producer - deployed to a J2EE application server, which is often remote and communicates through Simple Object Access Protocol (SOAP) over HTTP.

WSRP portlet producer - Web Services for Remote Portlets (WSRP) is a Web services standard that allows interoperability between a standards enabled container and a WSRP client application.

15.13.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.13.6.2 User Action

No action required.

15.14 WebCenter Producers

Provides information about the WebCenter Producers metrics.

15.14.1 Application Name

This metric identifies the WebCenter application to which these portlet producer metrics apply.

15.14.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.14.1.2 User Action

No action required.

15.14.2 Successful Invocations (%)

This metric indicates the percentage of portlet invocations that succeeded for this producer.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

Any request that fails will impact this percentage. This includes WebCenter application-related failures such as timeouts and internal errors, as well as remote/server failures such as requests returned with response codes HTTP4xx or HTTP5xx, responses with a bad content type, and SOAP faults, where applicable. Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.14.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.14.2.2 User Action

If Successful Invocations (%) is below 100%, investigate the metric in more detail. Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Producer Metrics](#).

You can also check the diagnostic logs to establish why producer requests are failing.

15.14.3 Average HTTP 2xx Time (ms)

This metric indicates the average time that all portlets from this producer take to process requests returning a HTTP2xx response code (in ms).

The last 10-15 minutes of data is used to calculate the average HTTP 2xx time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

See, [Viewing Performance Information](#).

15.14.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.14.3.2 User Action

Use this metric to detect the most performing producers. If you use this metric in conjunction with the Invocations (per minute) metric, then you can identify the top-performing producer.

See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Producer Metrics](#).

15.14.4 Invocations (per minute)

This metric indicates the total number of invocations per minute for a producer. The number of invocations is available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.14.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.14.4.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate producer performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the invocation rate for the producer is high (compared with other producers), consider prioritizing troubleshooting efforts in this area. Navigate to the application's main metric page where additional metric data is available. See, [Viewing Performance Information](#).

See, [Analyzing Portlet Producer Metrics](#).

15.14.5 Average Processing Time (ms)

This metric indicates the average time spent processing operations for a particular portlet producer (in ms). Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.14.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.14.5.2 User Action

Use this metric to determine the performance of portlet producers. If this metric is out-of-bounds (the average time for producer operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Producer Metrics](#).

15.14.6 Producer Type

The portlet producer type: Web or WSRP

Web portlet producer - deployed to a J2EE application server, which is often remote and communicates through Simple Object Access Protocol (SOAP) over HTTP.

WSRP portlet producer - Web Services for Remote Portlets (WSRP) is a Web services standard that allows interoperability between a standards enabled container and a WSRP client application.

15.14.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.14.6.2 User Action

No action required.

15.15 WebCenter Producers (Top 5)

Provides information about the WebCenter Producers (Top 5) metrics.

15.15.1 Application Name

This metric identifies the WebCenter application to which these portlet producer metrics apply.

15.15.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.15.1.2 User Action

No action required.

15.15.2 Successful Invocations (%)

This metric indicates the percentage of producer invocations that succeeded.

15.15.2.1 Description

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

This producer is one of the top five, most popular producers in this WebCenter application.

Any request that fails will impact this percentage. This includes WebCenter application-related failures such as timeouts and internal errors, as well as remote/server failures such as requests returned with response codes HTTP4xx or HTTP5xx, responses with a bad content type, and SOAP faults, where applicable. Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and

continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.15.2.2 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.15.2.3 User Action

If Successful Invocations (%) is below 100%, investigate the metric in more detail. Navigate to the application's main metric page where more detailed metric data is available. See, Viewing Performance Information.

See also, Analyzing Portlet Producer Metrics.

You can also check the diagnostic logs to establish why producer requests are failing.

15.15.3 Average HTTP 2xx Time (ms)

This metric indicates the average time that all portlets from this producer take to process requests returning a HTTP2xx response code (in ms).

This producer is one of the top five, most popular producers in this WebCenter application.

The last 10-15 minutes of data is used to calculate the average HTTP 2xx time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

See, Viewing Performance Information.

15.15.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.15.3.2 User Action

Use this metric to track the performance of popular producers.

If this metric is out-of-bounds, consider prioritizing troubleshooting efforts for the most popular producers. Navigate to the application's main metric page where additional metric data is available. See, Viewing Performance Information.

See also, Analyzing Portlet Producer Metrics.

15.15.4 Invocations (per minute)

This metric indicates the total number of invocations per minute for this producer.

This producer is one of the top five, most popular producers in this WebCenter application.

The number of invocations is available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.15.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.15.4.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate producer performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for this producer is high (compared with other producers), consider prioritizing troubleshooting efforts in this area. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Producer Metrics](#).

15.15.5 Average Processing Time (ms)

This metric indicates the average time (in ms) spent processing operations for this producer. Maximum processing times for individual HTTP request types are also available on the application's main metric page. See, [Viewing Performance Information](#).

This producer is one of the top five, most popular producers in this WebCenter application.

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.15.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.15.5.2 User Action

Use this metric to determine the performance of the top five producers. If this metric is out-of-bounds (the average time for producer operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Portlet Producer Metrics](#).

15.15.6 Producer Type

The portlet producer type: Web or WSRP

Web portlet producer - deployed to a J2EE application server, which is often remote and communicates through Simple Object Access Protocol (SOAP) over HTTP.

WSRP portlet producer - Web Services for Remote Portlets (WSRP) is a Web services standard that allows interoperability between a standards enabled container and a WSRP client application.

15.15.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.15.6.2 User Action

No action required.

15.16 WebCenter Announcements

Provides information about the WebCenter Announcements metrics.

15.16.1 Application Name

This metric identifies the WebCenter application to which these Announcements service metrics apply.

15.16.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.16.1.2 User Action

No action required.

15.16.2 Successful Invocations (%)

This metric indicates the percentage of Announcements service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.16.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.16.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data for individual operations is also available. See, [Viewing Performance Information](#).

See also, [Analyzing Announcement Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.16.3 Invocations (per minute)

This metric indicates the total number of Announcements service invocations per minute.

The number of invocations per operation is available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.16.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.16.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Announcements service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Announcement Metrics](#).

15.16.4 Average Login Time (ms)

This metric indicates the average time spent logging in to the back-end discussions server hosting announcements (in ms).

The last 10-15 minutes of data is used to calculate the average login time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.16.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.16.4.2 User Action

Use this metric to determine the performance of the Login operation. If this metric is out-of-bounds (the average time to login is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for the Login operation (and all other operations) is available. Use the information on this page to help to determine the underlying cause of the time delay. See, [Viewing Performance Information](#).

See also, [Analyzing Announcement Metrics](#).

15.16.5 Logins (per minute)

This metric counts the number of times per minute a WebCenter user logs in to the back-end discussions server hosting announcements.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests

metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.16.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.16.5.2 User Action

Use this metric to monitor typical and peak usage of the Announcements service. You can evaluate when most logins occur, as well as the maximum and minimum usage--which can be useful for tuning discussion server performance.

If you are experiencing performance issues outside normal peak login times, navigate to the application's main metric page where detailed metric data for all operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Announcement Metrics](#).

15.16.6 Average Processing Time (ms)

This metric indicates the average time the Announcements service spends processing announcement operations (in ms).

Announcement operations include: Create, List, Search, Login, Logout. Processing times for individual operations are available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.16.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.16.6.2 User Action

Use this metric to determine the performance of Announcements service operations.

If this metric is out-of-bounds (the average time for announcement operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Announcement Metrics](#).

15.16.7 Service Name

These metrics apply to the Announcements service.

15.16.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.16.7.2 User Action

No action required.

15.17 WebCenter Spaces

Provides information about WebCenter Spaces metrics.

15.17.1 Application Name

15.17.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.17.1.2 User Action

No action required.

15.17.2 Successful Invocations (%)

This metric indicates the percentage of group space invocations that succeeded. Processing time is available on application's main metric page.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.17.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 300 Seconds
--------------	-------------------

15.17.2.2 User Action

If Successful Invocations (%) is below 100% investigate this metric in more detail.

Navigate to the application's group space metric page where more detailed metric data is available. See, [Monitoring WebCenter Spaces Information](#).

15.17.3 Invocations (per minute)

This metric indicates the number of group space invocations per minute.

The number of invocations for this group space is available on application's group space metric page. See, [Monitoring WebCenter Spaces Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.17.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.17.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for this group space is high (compared with other group spaces), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's group space metric page where more detailed metric data is available. See, [Monitoring WebCenter Spaces Information](#).

See also, [Analyzing Group Space Metrics](#).

15.17.4 Page Throughput (per minute)

This metric shows the average number of pages accessed per minute for the group space.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests

metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.17.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.17.4.2 User Action

Use this metric to monitor typical and peak usage of the group space. You can evaluate when pages are accessed the most, as well as the maximum and minimum usage.

See also, *Analyzing Group Space Metrics*.

15.17.5 Average Page Processing Time (ms)

This metric indicates the average time spent processing operations for a particular group space (in ms).

The last 10-15 minutes of data is used to calculate the average page processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.17.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.17.5.2 User Action

Use this metric to determine the performance of a group space.

If this metric is out-of-bounds (the average time for group space operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, *Monitoring WebCenter Spaces Information*.

See also, *Analyzing Group Space Metrics*.

15.18 WebCenter Spaces (Top 5)

Provides information about WebCenter Spaces (Top 5) metrics.

15.18.1 Application Name

This metric identifies the WebCenter application to which these group space metrics apply.

15.18.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.18.1.2 User Action

No action required.

15.18.2 Successful Invocations (%)

This metric indicates the percentage of group space invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

This group space is one of the top five, most popular group spaces in this WebCenter application.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.18.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.18.2.2 User Action

If Successful Invocations (%) is below 100%, navigate to the application's group space metric page where more detailed metric data is available. See, [Monitoring WebCenter Spaces Information](#).

See also, [Analyzing Group Space Metrics](#).

You can also check the diagnostic logs to establish why group space invocations are failing.

15.18.3 Invocations (per minute)

This metric indicates the total number of group space invocations per minute for a single group space. This group space is one of the top five, most popular group spaces in this WebCenter application.

The number of invocations for this group space is available on application's main metric page. See, [Monitoring WebCenter Spaces Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15

minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.18.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.18.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for this group space is high (compared with other group spaces), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's group space metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Group Space Metrics](#).

15.18.4 Page Throughput (per minute)

This metric shows the average number of pages accessed per minute for the group space. This group space is one of the top five, most popular group spaces in this WebCenter application.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.18.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.18.4.2 User Action

Use this metric to monitor typical and peak usage of the group space. You can evaluate when pages are accessed the most, as well as the maximum and minimum usage.

See also, [Analyzing Group Space Metrics](#).

15.18.5 Average Page Processing Time (ms)

This metric indicates the average time (in ms) spent processing operations for the group space.

This group space is one of the top five, most popular group spaces in this WebCenter application. Processing times for all group spaces is available on application's group space metric page. See, [Monitoring WebCenter Spaces Information](#).

The last 10-15 minutes of data is used to calculate the average page processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.18.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.18.5.2 User Action

Use this metric to determine the performance of the five most popular group spaces.

If this metric is out-of-bounds (the average time for group space operations is increasing or higher than expected), navigate to the application's group space metric page where more detailed metric data is available. See, [Monitoring WebCenter Spaces Information](#).

See also, [Analyzing Group Space Metrics](#).

15.19 WebCenter Content Repositories

Provides information about WebCenter Content Repository metrics.

15.19.1 Application Name

This metric identifies the WebCenter application to which these content repository metrics apply.

15.19.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.19.1.2 User Action

No action required.

15.19.2 Successful Invocations (%)

This metric indicates the percentage of Documents service invocations that succeeded for this content repository.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.19.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.19.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Content Repository Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.19.3 Average Get Item Time (ms)

This metric indicates the average time spent selecting and displaying items, such as a document or image, in Content Presenter task flows.

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing metrics that are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.19.3.1 User Action

If this metric is out-of-bounds (the average time for Content Presenter operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See [Viewing Performance Information](#).

See also [Analyzing Content Repository Metrics](#).

15.19.4 Get Item

This metric indicates the number of times an item, such as a document or image, is selected or displayed in Content Presenter task flows.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing metrics that are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.19.4.1 User Action

If you are experiencing performance issues and the number of Get Item operations is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See Viewing Performance Information.

See also Analyzing Content Repository Metrics.

15.19.5 Total Get Item Time (ms)

This metric indicates the total time spent selecting and displaying items, such as a document or image, in Content Presenter task flows.

15.19.5.1 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate application performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total time spent on Get Item operations is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See Viewing Performance Information.

See also Analyzing Content Repository Metrics.

15.19.6 Invocations (per minute)

This metric indicates the total number of Documents service invocations per minute for this content repository.

The number of invocations per operation is available on the application's main metric page. See, Viewing Performance Information.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.19.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.19.6.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for this content repository is high (compared with other content repositories), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Content Repository Metrics](#).

15.19.7 Average List Folder Time (ms)

This metric indicates the average time Content Presenter task flows spend listing folders stored in a content repository.

The last 10-15 minutes of data is used to calculate the average time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing metrics that are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.19.7.1 User Action

If this metric is out-of-bounds (the average time for Content Presenter operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See [Viewing Performance Information](#).

See also [Analyzing Content Repository Metrics](#).

15.19.8 List Folder

This metric indicates the number of times folder content is displayed through Content Presenter task flows.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing metrics that are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.19.8.1 User Action

If you are experiencing performance issues and the number of List Folder operations is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See [Viewing Performance Information](#).

See also [Analyzing Content Repository Metrics](#).

15.19.9 Total List Folder Time (ms)

This metric indicates the total time Content Presenter task flows spend listing folders stored in a content repository.

15.19.9.1 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate application performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total time spent listing folders in Content Presenter task flows is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See [Viewing Performance Information](#).

See also [Analyzing Content Repository Metrics](#).

15.19.10 Average Login Time (ms)

This metric indicates the average time spent logging in to the content repository (in ms).

The last 10-15 minutes of data is used to calculate the average login time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.19.10.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.19.10.2 User Action

Use this metric to determine the performance of the Login operation. If this metric is out-of-bounds (the average time to login is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for the Login operation (and all other operations) is available. Use the information on this

page to help determine the underlying cause of the time delay. See, [Viewing Performance Information](#).

See also, [Analyzing Content Repository Metrics](#).

15.19.11 Logins (per minute)

This metric counts the number of times a WebCenter user logs in to the content repository per minute.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.19.11.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.19.11.2 User Action

Use this metric to monitor typical and peak usage of the content repository. You can evaluate when most logins occur, as well as the maximum and minimum usage--which can be useful for tuning content repository performance.

If you are experiencing performance issues outside normal peak login times, navigate to the application's main metric page where detailed metric data for all operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Content Repository Metrics](#).

15.19.12 Average Processing Time (ms)

This metric indicates the average time the Documents service spends processing operations relating to this content repository (in ms).

Content repository operations include: Download, Upload, Login, Delete. Processing times for individual operations are available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.19.12.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.19.12.2 User Action

Use this metric to determine the performance of Document service operations, for this content repository.

If this metric is out-of-bounds (the average time for content repository operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Content Repository Metrics](#).

15.20 WebCenter Events

Provides information about the WebCenter events metrics.

15.20.1 Application Name

This metric identifies the WebCenter application to which these Group Space Events or Personal Events service metrics apply.

15.20.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.20.1.2 User Action

No action required.

15.20.2 Successful Invocations (%)

This metric indicates the percentage of Events service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.20.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 300 Seconds
--------------	-------------------

15.20.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric further. Navigate to the application's main metric page where detailed metric data for individual operations is also available for group space events and personal events. See, [Viewing Performance Information](#).

See also, [Analyzing Events Metrics](#).

15.20.3 Invocations (per minute)

This metric indicates the total number of invocations per minute for the Group Space Events or Personal Events service.

The number of invocations per operation for both group space and personal events is available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.20.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.20.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Group Space Events service or the Personal Events service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Events Metrics](#).

15.20.4 Average Processing Time (ms)

This metric indicates the average time the Events service spends processing operations (in ms) associated with group space events or personal events.

Events operations include: Create Event, Update Event, List Event, Delete Event, Search Event. Processing times for individual operations are available on the

application's main metric page for both group space and personal events. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10- 15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.20.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.20.4.2 User Action

Use this metric to determine the performance of Events service operations. If this metric is out-of-bounds (the average time for Events operations is increasing or higher than expected), navigate to the application's main metric page where detailed metric data for individual operations is also available for both group space and personal events. See, [Viewing Performance Information](#).

See also, [Analyzing Events Metrics](#).

15.21 WebCenter External Applications

Provides information about WebCenter External Applications metrics.

15.21.1 Application Name

This metric identifies the WebCenter application to which these External Application metrics apply.

15.21.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.21.1.2 User Action

No action required.

15.21.2 Invocations (per minute)

This metric indicates the total number of times per minute a single external application was accessed.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15

minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.21.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.21.2.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate application performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for this external application is high (compared with other external applications), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See, [Analyzing External Application Metrics](#).

15.22 WebCenter External Applications

Provides information about the WebCenter external applications metrics.

15.22.1 Application Name

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.22.2 Name

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.23 WebCenter Discussion Forum

Provides information about the WebCenter Discussion forum metrics.

15.23.1 Application Name

This metric identifies the WebCenter application to which these Discussions service metrics apply.

15.23.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.23.1.2 User Action

No action required.

15.23.2 Successful Invocations (%)

This metric indicates the percentage of Discussions service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.23.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.23.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Discussions Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.23.3 Invocations (per minute)

This metric indicates the total number of Discussions service invocations per minute. The number of invocations per operation is available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.23.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.23.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Discussions service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Discussions Metrics](#).

15.23.4 Average Login Time (ms)

This metric indicates the average time spent logging in to the back-end discussions server hosting discussion forums (in ms).

The last 10-15 minutes of data is used to calculate the average login time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.23.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.23.4.2 User Action

Use this metric to determine the performance of the Login operation. If this metric is out-of-bounds (the average time to login is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for the Login operation (and all other operations) is available. Use the information on this page to help to determine the underlying cause of the time delay. See, [Viewing Performance Information](#).

See also, [Analyzing Discussions Metrics](#).

15.23.5 Logins (per minute)

This metric counts the number of times per minute a WebCenter user logs in to the back-end discussions server hosting discussion forums.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.23.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.23.5.2 User Action

Use this metric to monitor typical and peak usage of the Discussions service. You can evaluate when most logins occur, as well as the maximum and minimum usage-- which can be useful for tuning discussion server performance.

If you are experiencing performance issues outside normal peak login times, navigate to the application's main metric page where detailed metric data for all operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Discussions Metrics](#).

15.23.6 Average Processing Time (ms)

This metric indicates the average time the Discussions service spends processing operations (in ms).

Discussions service operations include: Login, Create Forum, Create Topic, List Forums, List Topics, Search, Logout. Processing times for individual operations are available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.23.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.23.6.2 User Action

Use this metric to determine the performance of Discussions service operations.

If this metric is out-of-bounds (the average time for discussion forum operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See, Viewing Performance Information.

See also, Analyzing Discussions Metrics.

15.23.7 Service Name

These metrics apply to the Discussions service.

15.23.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.23.7.2 User Action

No action required.

15.24 WebCenter Spaces Pages

Provides information about WebCenter spaces pages metrics.

15.24.1 Application Name

This metric identifies the WebCenter application to which these page metrics apply.

15.24.1.1 User Action

No action required.

15.24.2 Successful Invocations (%)

This metric indicates the percentage of page invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.24.2.1 User Action

If Successful Invocations (%) is below 100%, navigate to the application's page metric page where more detailed metric data is available.

See Monitoring Page Metrics for Spaces and Monitoring Page Metrics for WebCenter Portal Applications.

You can also check the diagnostic logs to establish why page invocations are failing.

15.24.3 Invocations (per minute)

This metric indicates the total number of page invocations per minute.

The number of invocations for this page is available on application's main metric page. See [Monitoring Page Metrics for Spaces](#) and [Monitoring Page Metrics for WebCenter Portal Applications](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.24.3.1 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate page performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for this page is high (compared with other pages), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's page metric page where more detailed metric data is available. See, [Monitoring Page Metrics for Spaces](#) and [Monitoring Page Metrics for WebCenter Portal Applications](#).

15.24.4 Page Throughput (per minute)

This metric shows the average number of pages accessed per minute.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.24.4.1 User Action

Use this metric to monitor typical and peak usage of the page. You can evaluate when this page is accessed the most, as well as the maximum and minimum usage.

See, [Monitoring Page Metrics for Spaces](#) and [Monitoring Page Metrics for WebCenter Portal Applications](#).

15.24.5 Average PPR Processing Time (ms)

This metric indicates the average time (in ms) to refresh the partial page.

The last 10-15 minutes of data is used to calculate the average page processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection

period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.24.6 Average Page Processing Time (ms)

This metric indicates the average time (in ms) to display the page.

The last 10-15 minutes of data is used to calculate the average page processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.24.6.1 User Action

Use this metric to determine the performance of a page.

If this metric is out-of-bounds (the average time to display pages is increasing or higher than expected), navigate to the application's page metric page where more detailed metric data is available.

See, *Monitoring Page Metrics for Spaces* and *Monitoring Page Metrics for WebCenter Portal Applications*.

15.25 WebCenter Spaces Pages (Top 5)

Provides information about WebCenter spaces pages (Top 5) metrics

15.25.1 Application Name

This metric identifies the WebCenter application to which these page metrics apply.

15.25.1.1 User Action

No action required.

15.25.2 Successful Invocations (%)

This metric indicates the percentage of page invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

This page is one of the top five, most popular pages in this WebCenter application.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests

metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.25.2.1 User Action

If Successful Invocations (%) is below 100%, navigate to the application's page metric page where more detailed metric data is available. See, [Monitoring Page Metrics for Spaces](#) and [Monitoring Page Metrics for WebCenter Portal Applications](#).

You can also check the diagnostic logs to establish why page invocations are failing.

15.25.3 Invocations (per minute)

This metric indicates the total number of page invocations per minute. This page is one of the top five, most popular pages in this WebCenter application.

The number of invocations for this page is available on application's main metric page. See, [Monitoring Page Metrics for Spaces](#) and [Monitoring Page Metrics for WebCenter Portal Applications](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.25.3.1 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for this page is high (compared with other pages), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's page metric page where more detailed metric data is available. See, [Monitoring Page Metrics for Spaces](#) and [Monitoring Page Metrics for WebCenter Portal Applications](#).

15.25.4 Page Throughput (per minute)

This metric shows the average number of pages accessed per minute. This page is one of the top five, most popular pages in this WebCenter application.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.25.4.1 User Action

Use this metric to monitor typical and peak usage of the page. You can evaluate when pages are accessed the most, as well as the maximum and minimum usage.

15.25.5 Average Page Processing Time (ms)

This metric indicates the average time (in ms) to display the page.

This page is one of the top five, most popular pages in this WebCenter application. Processing times for all pages is available on application's page metric page. See, [Monitoring Page Metrics for Spaces](#) and [Monitoring Page Metrics for WebCenter Portal Applications](#).

The last 10-15 minutes of data is used to calculate the average page processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.25.5.1 User Action

Use this metric to determine the performance of the five most popular pages.

If this metric is out-of-bounds (the average time to display pages is increasing or higher than expected), navigate to the application's page metric page where more detailed metric data is available. See, [Monitoring Page Metrics for Spaces](#) and [Monitoring Page Metrics for WebCenter Portal Applications](#).

15.26 WebCenter Import/Export

Provides information about WebCenter Import/Export metrics.

15.26.1 Application Name

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.26.2 Service Name

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.27 WebCenter Lists

Provides information about WebCenter Lists metrics.

15.27.1 Application Name

This metric identifies the WebCenter application to which these Lists service metrics apply.

15.27.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.27.1.2 User Action

No action required.

15.27.2 Successful Invocations (%)

This metric indicates the percentage of Lists service invocations that succeeded. Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.27.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.27.2.2 User Action

If Successful Invocations (%) is less than 100%, navigate to the application's main metric page where more detailed metric data for individual operations is also available. See, Viewing Performance Information.

See also, Analyzing List Metrics.

You can also check the diagnostic logs to establish why service requests are failing.

15.27.3 Invocations (per minute)

This metric indicates the total number of Lists service invocations per minute. The number of invocations per operation is available on the application's main metric page. See Viewing Performance Information.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise

Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.27.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.27.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Lists service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, Viewing Performance Information.

See also, Analyzing List Metrics.

15.27.4 Average Processing Time (ms)

This metric indicates the average time the Lists service spends processing list operations (in ms).

List operations include: Create List, Copy List, Delete List, Create Row, Delete Row, Update Row, Search, and Save Data. Processing times for individual operations are available on application's main metric page. See, Viewing Performance Information.

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.27.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.27.4.2 User Action

Use this metric to determine the performance of Lists service operations. If this metric is out-of-bounds (the average time for list operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See, Viewing Performance Information.

See also, Analyzing List Metrics.

15.27.5 Service Name

These metrics apply to the Lists service.

15.27.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.27.5.2 User Action

No action required.

15.28 WebCenter Mail

Provides information about the WebCenter Mail metrics.

15.28.1 Application Name

This metric identifies the WebCenter application to which these Mail service metrics apply.

15.28.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.28.1.2 User Action

No action required.

15.28.2 Successful Invocations (%)

This metric indicates the percentage of Mail service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.28.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.28.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data for individual operations is also available. See, [Viewing Performance Information](#).

See also, [Analyzing Mail Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.28.3 Invocations (per minute)

This metric indicates the total number of Mail service invocations per minute.

The number of invocations per operation is available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.28.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.28.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Mail service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Mail Metrics](#).

15.28.4 Average Login Time (ms)

This metric indicates the average time spent logging in to the mail server that is hosting mail services (in ms).

The last 10-15 minutes of data is used to calculate the average login time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period

(until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.28.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.28.4.2 User Action

Use this metric to determine the performance of the Login operation. If this metric is out-of-bounds (the average time to login is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for the Login operation (and all other operations) is available. Use the information on this page to help determine the underlying cause of the time delay. See, [Viewing Performance Information](#).

See also, [Analyzing Mail Metrics](#).

15.28.5 Logins (per minute)

This metric counts the number of times per minute a WebCenter user logs in to the back-end mail server that is hosting mail services.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.28.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.28.5.2 User Action

Use this metric to monitor typical and peak usage of the Instant Messaging and Presence (IMP) service.

Use this metric to monitor typical and peak usage of the Mail service. You can evaluate when most logins occur, as well as the maximum and minimum usage--which can be useful for tuning mail server performance.

If you are experiencing performance issues outside normal peak login times, navigate to the application's main metric page where detailed metric data for all operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Mail Metrics](#).

15.28.6 Average Processing Time (ms)

This metric indicates the average time the Mail service spends processing operations (in ms).

Mail operations include: Login, Logout, Receive, Send, Search. Processing times for individual operations are available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.28.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.28.6.2 User Action

Use this metric to determine the performance of Mail service operations.

If this metric is out-of-bounds (the average time for mail operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Mail Metrics](#).

15.28.7 Service Name

These metrics apply to the Mail service.

15.28.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.28.7.2 User Action

No action required.

15.29 WebCenter Notes

Provides information about the WebCenter Notes metrics.

15.29.1 Application Name

This metric identifies the WebCenter application to which these Notes service metrics apply.

15.29.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.29.1.2 User Action

No action required.

15.29.2 Successful Invocations (%)

This metric indicates the percentage of Notes service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.29.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.29.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric further. Navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Note Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.29.3 Invocations (per minute)

This metric indicates the total number of Notes service invocations per minute.

The count includes the following Notes operations: Create, Delete, Find, Find All Notes, Save Changes, Update. The number of invocations per operation is available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.29.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.29.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Notes service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

Also see, [Analyzing Note Metrics](#).

15.29.4 Average Processing Time (ms)

This metric indicates the average time the Notes service spends processing note operations (in ms).

Notes operations include: Create, Update, Find. Processing times for individual operations are available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.29.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 300 Seconds
--------------	-------------------

15.29.4.2 User Action

Use this metric to determine the performance of Notes service operations. If this metric is out-of-bounds (the average time for note operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Note Metrics](#).

15.29.5 Service Name

These metrics apply to the Notes service.

15.29.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.29.5.2 User Action

No action required.

15.30 WebCenter Page Service

Provides information about the WebCenter Page Service metrics.

15.30.1 Application Name

This metric identifies the WebCenter application to which these Page service metrics apply.

15.30.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.30.1.2 User Action

No action required.

15.30.2 Successful Invocations (%)

This metric indicates the percentage of Page service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and

continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.30.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.30.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data for individual operations is available. See, Viewing Performance Information.

See also, Analyzing Page Metrics.

You can also check the diagnostic logs to establish why service requests are failing.

15.30.3 Invocations (per minute)

This metric indicates the total number of Page service invocations per minute.

The number of invocations per operation is available on the application's main metric page. See, Viewing Performance Information.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.30.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.30.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Page service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, Viewing Performance Information.

See also Analyzing Page Metrics.

15.30.4 Average Processing Time (ms)

This metric indicates the average time the Page service spends processing operations associated with the Page service (in ms).

Page operations include: Create, Update Properties, Copy, Delete, Search. Processing times for individual operations are available on the application's main metric page. See, Viewing Performance Information.

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.30.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.30.4.2 User Action

Use this metric whilst assessing the overall performance of the Page service.

If this metric is out-of-bounds (the average time for page operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See, Viewing Performance Information.

See also, Analyzing Page Metrics.

15.30.5 Service Name

These metrics apply to the Page service.

15.30.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.30.5.2 User Action

No action required.

15.31 WebCenter People Connections

Provides information about the WebCenter People Connections metrics.

15.31.1 Application name

This metric identifies the WebCenter application to which these People Connections service metrics apply.

15.31.1.1 User Action

No action required.

15.31.2 Successful Invocations (%)

This metric indicates the percentage of People Connection service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.31.2.1 User Action

If Successful Invocations (%) is below 100%, investigate this metric further. Navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing People Connection Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.31.3 Invocations (per minute)

This metric indicates the total number of People Connection service invocations per minute.

The count includes the following People Connection operations: Get Profiles, Get Activities, Publish Activities, Get Messages, Get Feedback, Get Connections. The number of invocations per operation is available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.31.3.1 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the People Connections service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

Also see, [Analyzing People Connection Metrics](#).

15.31.4 Average Processing Time (ms)

This metric indicates the average time the People Connections service spends processing user operations (in ms).

People Connection operations include: Get Profiles, Get Activities, Publish Activities, Get Messages, Get Feedback, Get Connections. Processing times for individual operations are available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.31.4.1 User Action

Use this metric to determine the performance of People Connection service operations. If this metric is out-of-bounds (the average time for user operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing People Connection Metrics](#).

15.31.5 Service Name

These metrics apply to the People Connections service.

15.31.5.1 User Action

No action required.

15.32 WebCenter Polls

Provides information about WebCenter Polls metrics.

15.32.1 Application Name

This metric identifies the WebCenter application to which these Poll service metrics apply.

15.32.1.1 User Action

No action required.

15.32.2 Successful Invocations (%)

This metric indicates the percentage of Poll service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.32.2.1 User Action

If Successful Invocations (%) is below 100%, investigate this metric further. Navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Poll Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.32.3 Invocations (per minute)

This metric indicates the total number of Poll service invocations per minute.

The count includes the following Polls operations: Create Poll, Edit Poll, Delete Poll, Get Poll By ID, Submit Poll, Analyze Results. The number of invocations per operation is available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.32.3.1 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Polls service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

Also see, [Analyzing Poll Metrics](#).

15.32.4 Average Processing Time (ms)

This metric indicates the average time the Polls service spends processing poll operations (in ms).

Polls operations include: Create Poll, Edit Poll, Delete Poll, Get Poll By ID, Submit Poll, Analyze Results. Processing times for individual operations are available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.32.4.1 User Action

Use this metric to determine the performance of Poll service operations. If this metric is out-of-bounds (the average time for poll operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Poll Metrics](#).

15.32.5 Service Name

These metrics apply to the Polls service.

15.32.5.1 User Action

No action required.

15.33 WebCenter Recent Activity

Provides information about the WebCenter Recent Activity metrics.

15.33.1 Application Name

This metric identifies the WebCenter application to which these Recent Activity metrics apply.

15.33.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.33.1.2 User Action

No action required.

15.33.2 Successful Invocations (%)

This metric indicates the percentage of Recent Activities service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.33.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.33.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Recent Activity Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.33.3 Invocations (per minute)

This metric indicates the total number of Recent Activities service invocations per minute.

The number of invocations is available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.33.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.33.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the Recent Activities service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Recent Activity Metrics](#).

15.33.4 Average Processing Time (ms)

This metric indicates the total time the Recent Activities service spends processing Recent Activity operations (in ms).

Processing time is available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.33.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.33.4.2 User Action

Use this metric to determine the performance of Recent Activities service operations.

If this metric is out-of-bounds (the average time for recent activity operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Recent Activity Metrics](#).

15.33.5 Service Name

These metrics apply to the Recent Activities service.

15.33.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.33.5.2 User Action

No action required.

15.34 WebCenter RSS News Feeds

Provides information about the WebCenter RSS News Feeds metrics.

15.34.1 Application Name

This metric identifies the WebCenter application to which these RSS News Feed metrics apply.

15.34.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.34.1.2 User Action

No action required.

15.34.2 Successful Invocations (%)

This metric indicates the percentage of RSS service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.34.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.34.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing RSS News Feed Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.34.3 Invocations (per minute)

This metric indicates the total number of RSS New Feed invocations per minute.

The number of invocations is available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.34.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.34.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for RSS news feeds is high (compared with other services consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing RSS News Feed Metrics](#).

15.34.4 Average Processing Time (ms)

This metric indicates the average time the RSS service spends processing operations (in ms).

Processing time is available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.34.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.34.4.2 User Action

Use this metric to determine the performance of RSS service operations.

If this metric is out-of-bounds (the average time for RSS operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing RSS News Feed Metrics](#).

15.34.5 Service Name

These metrics apply to the RSS service.

15.34.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.34.5.2 User Action

No action required.

15.35 WebCenter Instant Messaging and Presence

Provides information about the WebCenter Instant Messaging and Presence metrics.

15.35.1 Application Name

This metric identifies the WebCenter application to which these Instant Messaging and Presence service metrics apply.

15.35.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.35.1.2 User Action

No action required.

15.35.2 Successful Invocations (%)

This metric indicates the percentage of Instant Messaging and Presence (IMP) service invocations that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.35.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.35.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data for individual operations is available. See, Viewing Performance Information.

See also, Analyzing Instant Messaging and Presence (IMP) Metrics.

You can also check the diagnostic logs to establish why service requests are failing.

15.35.3 Invocations (per minute)

This metric indicates the total number of Instant Messaging and Presence (IMP) service invocations per minute.

The number of invocations per operation is available on the application's main metric page. See, Viewing Performance Information.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.35.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.35.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for the IMP service is high (compared with other services), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Instant Messaging and Presence \(IMP\) Metrics](#).

15.35.4 Average Login Time (ms)

This metric indicates the average time spent logging in to the back-end server hosting instant messaging and presence services (in ms).

The last 10-15 minutes of data is used to calculate the average login time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.35.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.35.4.2 User Action

Use this metric to determine the performance of the Login operation. If this metric is out-of-bounds (the average time to login is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for the Login operation (and all other operations) is available. Use the information on this page to help determine the underlying cause of the time delay. See, [Viewing Performance Information](#).

See also, [Analyzing Instant Messaging and Presence \(IMP\) Metrics](#).

15.35.5 Logins (per minute)

This metric counts the number of times a minute a WebCenter user logs in to the server that is hosting instant messaging and presence services.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.35.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.35.5.2 User Action

Use this metric to monitor typical and peak usage of the Instant Messaging and Presence (IMP) service. You can evaluate when most logins occur, as well as the maximum and minimum usage-- which can be useful for tuning IMP server performance.

If you are experiencing performance issues outside normal peak login times, navigate to the application's main metric page where detailed metric data for all operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Instant Messaging and Presence \(IMP\) Metrics](#).

15.35.6 Average Processing Time (ms)

This metric indicates the average time the Instant Messaging and Presence (IMP) service spends processing IMP operations (in ms).

IMP operations include: Get Presence, Login, Logout. Processing times for individual operations are available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.35.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.35.6.2 User Action

Use this metric to determine the performance of Instant Messaging and Presence (IMP) service operations.

If this metric is out-of-bounds (the average time for IMP operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for individual operations is available. See, [Viewing Performance Information](#).

See also, [Analyzing Instant Messaging and Presence \(IMP\) Metrics](#).

15.35.7 Service Name

These metrics apply to the Instant Messaging and Presence (IMP) service.

15.35.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.35.7.2 User Action

No action required.

15.36 WebCenter Search

Provides information about WebCenter Search metrics.

15.36.1 Application Name

This metric identifies the WebCenter application to which these Search service metrics apply.

15.36.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.36.1.2 User Action

No action required.

15.36.2 Successful Invocations (%)

This metric indicates the percentage of Search service invocations that succeeded for a particular search source.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.36.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.36.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data for the search source is available. See, [Viewing Performance Information](#).

See also, [Analyzing Search Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.36.3 Invocations (per minute)

This metric indicates the total number of Search service invocations per minute for a single search source.

WebCenter search sources include data associated with: Announcements, Documents, Discussion Forums, Group Spaces, Group Space Events, Lists, Notes, Oracle Secure Enterprise Search, Pages, Wikis and Blogs, People Connections, and Tagged Items.

The number of invocations for this search source is available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.36.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.36.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for this search source is high (compared with other search sources), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Search Metrics](#).

15.36.4 Average Processing Time (ms)

This metric indicates the average time the Search service spends processing search operations (in ms) for a single search source.

WebCenter search sources include data associated with: Announcements, Documents, Discussion Forums, Group Spaces, Group Space Events, Lists, Notes, Oracle Secure

Enterprise Search, Pages, Wikis and Blogs, People Connections, and Tagged Items. Processing times for all search sources are available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.36.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.36.4.2 User Action

Use this metric to determine the performance of Search service operations for a single search source.

If this metric is out-of-bounds (the average time for search operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data for the search source is available. See, [Viewing Performance Information](#).

See also, [Analyzing Search Metrics](#).

15.37 WebCenter Search (Top 5)

Provides information about the WebCenter Search (Top 5) metrics.

15.37.1 Application Name

This metric identifies the WebCenter application to which these Search service metrics apply.

15.37.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.37.1.2 User Action

No action required.

15.37.2 Successful Invocations (%)

This metric indicates the percentage of Search service invocations that succeeded. This search source is one of the top five most popular search sources in this WebCenter application.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

WebCenter search sources include data associated with: Announcements, Documents, Discussion Forums, Group Spaces, Group Space Events, Lists, Notes, Oracle Secure Enterprise Search, Pages, Wikis and Blogs, People Connections, and Tagged Items.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.37.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.37.2.2 User Action

If Successful Invocations (%) is below 100%, investigate the metric in more detail. Navigate to the application's main metric page where more detailed metric data for the search sources is available. See, [Viewing Performance Information](#).

See also, [Analyzing Search Metrics](#).

You can also check the diagnostic logs to establish why search requests are failing.

15.37.3 Invocations (per minute)

This metric indicates the total number of Search service invocations per minute for a single search source. This search source is one of the top five most popular search sources in this WebCenter application.

WebCenter search sources include data associated with: Announcements, Documents, Discussion Forums, Group Spaces, Group Space Events, Lists, Notes, Oracle Secure Enterprise Search, Pages, Wikis and Blogs, People Connections, and Tagged Items.

The number of invocations for this search source is available on application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.37.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.37.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for this search source is high (compared with other search sources), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Search Metrics](#).

15.37.4 Average Processing Time (ms)

This metric indicates the average time (in ms) the Search service spends processing search operations for a single search source. This search source is one of the top five, most popular search sources in this WebCenter application.

WebCenter search sources include data associated with: Announcements, Documents, Discussion Forums, Group Spaces, Group Space Events, Lists, Notes, Oracle Secure Enterprise Search, Pages, Wikis and Blogs, People Connections, and Tagged Items. Processing times for all search sources are available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.37.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.37.4.2 User Action

Use this metric to determine search performance for the five most popular search sources.

If this metric is out-of-bounds (the average time for search operations is increasing or higher than expected), navigate to the application's main metric page where more

detailed metric data for all search sources is available. See, [Viewing Performance Information](#).

See also, [Analyzing Search Metrics](#).

15.38 WebCenter BPEL Worklists

Provides information about the WebCenter BPEL worklists metrics.

15.38.1 Application Name

This metric identifies the WebCenter application to which these Worklists service metrics apply.

15.38.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.38.1.2 User Action

No action required.

15.38.2 Successful Invocations (%)

This metric indicates the percentage of Worklists service invocations, for this BPEL worklist, that succeeded.

Successful Invocations (%) is equal to the number of successful invocations divided by the invocation count.

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.38.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.38.2.2 User Action

If Successful Invocations (%) is below 100%, investigate this metric in more detail. Navigate to the application's main metric page where more detailed metric data is also available. See, [Viewing Performance Information](#).

See also, [Analyzing Worklist Metrics](#).

You can also check the diagnostic logs to establish why service requests are failing.

15.38.3 Invocations (per minute)

This metric indicates the total number of Worklists service invocations per minute for a single BPEL worklist. The number of invocations is available on the application's main metric page. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate this metric. If 10-15 minutes of data is not yet available, the metric is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.38.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.38.3.2 User Action

Use this metric to monitor minimum, typical, and peak usage, and to evaluate service performance under peak load conditions. You can use this information to assess whether additional resources or hardware are required for scalability.

If you are experiencing performance issues and the total number of invocations per minute for this BPEL worklist is high (compared with other worklists), consider prioritizing troubleshooting efforts in this area.

Navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Worklist Metrics](#).

15.38.4 Average Processing Time (ms)

This metric indicates the average time the Worklists service spends processing operations relating to this BPEL worklist (in ms).

Processing time is available on application's main metric page where detailed metric data is available. See, [Viewing Performance Information](#).

The last 10-15 minutes of data is used to calculate the average processing time. If 10-15 minutes of data is not yet available, the average is calculated over the collection period (until 10-15 minutes is reached). Metric collection starts when the application first starts up and continues indefinitely providing that metrics are requested by the client (Enterprise Manager or WLST). Metric collection stops temporarily if no metric requests are detected over a long period of time and will restart when the client next requests metrics. In this instance, the metric shows data for the delta period since metric collection stopped (until 10-15 minutes is reached).

15.38.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Seconds

15.38.4.2 User Action

Use this metric to determine the performance of Worklists service operations, for this BPEL worklist. If this metric is out-of-bounds (the average time for worklist operations is increasing or higher than expected), navigate to the application's main metric page where more detailed metric data is available. See, [Viewing Performance Information](#).

See also, [Analyzing Worklist Metrics](#).

This chapter describes Oracle Web Cache metrics.

16.1 Performance by Web Cache Instance

Provides information about the performance metrics for the Web Cache instance.

16.1.1 Instance

This category provides performance-request metrics for the cache.

16.1.2 Request Processing Time (milli seconds)

This metric specifies the average number of seconds used to process a request during the current time period.

16.1.2.1 User Action

If the number is low, then improve the connection between the cache and origin server, or upgrade the origin server computer.

16.1.3 Request Throughput (per second)

This metric specifies the average number of requests served for each second during the current time period.

16.1.3.1 User Action

If this number is high, then increase the capacity of this origin server from the Origin Servers page or load balance the requests among cache cluster members.

16.2 Resource Usage

This category of metrics shows the resource usage of the computer with Oracle Web Cache.

16.2.1 CPU Usage (%)

This metric specifies the percentage of the CPU that is being used for Oracle Web Cache. As traffic increases, CPU utilization increases.

16.2.1.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 3 Samples	>	Not Defined	Not Defined	2	CPU Utilization is %value%%%

16.2.1.2 User Action

If the load is high, then consider upgrading the cache computer.

16.2.2 CPU Idle Time (%)

This metric specifies the percentage of system CPU time that is idle.

16.2.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 5 Minutes

16.2.2.2 User Action

If the load is high, then consider upgrading the cache computer.

16.2.3 Other CPU Usage (%)

The percentage of physical memory usage by applications other than Oracle Web Cache.

16.2.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 5 Minutes

16.2.3.2 User Action

If CPU usage is high, then consider upgrading the cache computer.

16.2.4 Memory Usage (MB)

This metric specifies the total memory used by Oracle Web Cache.

16.2.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 5 Minutes

16.2.4.2 User Action

If the load is high, then consider upgrading the cache computer.

16.2.5 Free Memory (MB)

The metric specifies the amount of free physical memory, in megabytes, for the system.

16.2.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 5 Minutes

16.2.6 Other Memory Usage (MB)

16.2.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 5 Minutes

16.2.7 Total Memory (MB)

This metric shows the total memory of the host that has Oracle Web Cache.

16.2.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x	Every 5 Minutes

16.2.8 Memory Usage (%)

This metric specifies the percentage of total memory that is being utilized for cache memory.

16.2.8.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every 5 Minutes	After Every 3 Samples	>	80	90	2	Memory Utilization is %value%%%

16.2.9 Free Memory (%)

This metric specifies the percentage of total memory that is being utilized for cache memory.

16.2.9.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x	Every 5 Minutes

16.2.10 Other Memory Usage (%)

Percent of memory used by processes other than Oracle Web Cache.

16.2.10.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
9.0.2.x and 9.0.3.x	Every 5 Minutes

16.2.11 Start Time (ms since Epoch)

This metric shows the start time of Oracle Web Cache.

16.2.12 Up Time (ms since Epoch)

This metric shows the total time the Oracle Web Cache has been running.

16.3 Response

This category shows the availability of the cache server process, which manages the cache.

16.3.1 Response Admin

This category shows the availability of the admin server process, which provides administrative access to the cache.

16.3.2 Admin Process Status

This metric indicates whether the admin server process for the Oracle Web Cache instance is up (1) or down (0). The admin server process provides administrative access to cache.

16.3.2.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every 60 Samples	=	Not Defined	0	1	The Oracle Web Cache admin server process is down

16.3.2.2 User Action

If the admin server process is down, the context pane for the configuration pages displays an error, indicating that configuration is unavailable because the admin server process is down. To start the admin server process, from the Web Cache menu, choose Control, then Start Up, Shut Down, or Restart. See Starting and Stopping Oracle Web Cache for more information about starting Oracle Web Cache.

16.3.3 Response - Status

This metric indicates whether the cache server process for the Oracle Web Cache instance is up (1) or down (0).

16.3.3.1 Metric Summary

The following table shows how often the metric's value is collected and compared against the default thresholds. The 'Consecutive Number of Occurrences Preceding Notification' column indicates the consecutive number of times the comparison against thresholds should hold TRUE before an alert is generated.

Target Version	Evaluation and Collection Frequency	Upload Frequency	Operator	Default Warning Threshold	Default Critical Threshold	Consecutive Number of Occurrences Preceding Notification	Alert Text
All Versions	Every Minute	After Every 60 Samples	=	Not Defined	0	1	The Oracle Web Cache cache server process is down

16.3.3.2 User Action

See Starting and Stopping Oracle Web Cache for more information about starting Oracle Web Cache.

16.4 Performance of each Site with Summary

This category provides performance metrics for each site.

16.4.1 Total Number of Bytes Saved by Compression

This metric specifies the accumulated number of bytes that would be sent to clients if in-cache compression is disabled.

16.4.1.1 User Action

If the number is low, then compression is not providing much benefit to the site. Try enabling compression in all caching rules to allow Oracle Web Cache to determine compressibility automatically based on the content type.

16.4.2 Cache Size

The metric specifies the size, in megabytes, of the objects currently stored in the cache. For a cache cluster member, this number is an aggregate of the owned and on-demand objects.

16.4.2.1 User Action

No action user required. As Oracle Web Cache only cleans up invalid or expired objects on demand when memory is needed, this number may remain at a large value.

16.4.3 Total Number Error Pages Served

The metric specifies the accumulated number of error pages that Oracle Web Cache served to Web browsers since the cache was started.

16.4.3.1 User Action

Oracle Web Cache serves error pages when there are problems fetching content from the origin server or when there ESI processing issues with the content. Even a small number can indicate an issue with site content or with origin server health.

16.4.4 Hit Rate (in %)

This metric specifies the percentage of requests resolved by cache content. This percentage should be high, except when objects are being invalidated.

16.4.4.1 User Action

If percentage of hits is low, then perform the following:

1. Review the caching rules configuration in the Caching Rules page.
2. In the Popular Requests page, select Non Cache Popular Requests from the Show Popular Requests list to view the most frequent misses.
3. In the results table, view the Caching Reason column for the most frequent URL requests that were not cached.
4. Configure caching rules to cache the popular requests. See Configure Caching Rules.

16.4.5 Number of Cached

This metric specifies the number of objects cached.

16.4.6 Currently Open connections

This metric specifies the current number of incoming open connections to the Oracle Web Cache server and outgoing open connections to the origin servers.

16.4.6.1 User Action

If the number is close to the origin server connection limit, then set a larger capacity for the origin server from the Origin Servers page.

16.4.7 Total Number of Requests Denied

This metric specifies the accumulated number of requests denied by request filters.

16.4.7.1 User Action

Any non-zero number may be an indication of an attack on the site or an issue with the configuration of request filters.

16.4.8 Total Number of Requests Served

This metric specifies the accumulate number of requests that Oracle Web Cache has served.

16.5 WebCache_nontimedHost

This category provides performance-request metrics for the origin servers. Please also review the metrics in the WebCache_timedHost category for additional metrics for origin servers.

16.5.1 Capacity.value

This informational metric specifies the capacity you specified in the Origin Servers page for this origin server.

16.5.2 Load.value

This metric specifies the current number of connections from Oracle Web Cache that the origin server has open.

16.5.3 MaxLoad.value

The metric specifies the maximum number of connections that the origin server has had open at one time.

16.5.4 Port.value

The information metric specifies the port number from which the origin server is listening for Oracle Web Cache requests.

16.6 WebCache_timedHost

This category provides performance-request metrics for the origin servers. Please also review the metrics in the WebCache_nontimedHost category for additional metrics for origin servers.

16.6.1 AvgLatency.value

This metric shows the average amount of time for client requests forwarded by Oracle Web Cache to be resolved.

16.6.1.1 User Action

If the latency appears high, then the administrator for the origin server or content source may need to add more system resources.

16.6.2 Latency.value

This metric shows the total time for client requests forwarded by Oracle Web Cache to be resolved, since the cache was started.

16.6.2.1 User Action

If the latency appears high, then the administrator for the origin server or content source may need to add more system resources.

16.6.3 Request.value

This metric specifies the accumulated number of client requests forwarded by Oracle Web Cache that the origin server has processed.

16.6.3.1 User Action

You can increase cache hit rate and lower the number client requests to the origin server by configuring caching rules for popular requests:

If percentage of hits is low, then perform the following:

1. Review the caching rules configuration in the Caching Rules page.
2. In the Popular Requests page, select Non Cache Popular Requests from the Show Popular Requests list to view the most frequent misses.
3. In the results table, view the Caching Reason column for the most frequent URL requests that were not cached.
4. Configure caching rules to cache the popular requests. See Configure Caching Rules.

16.6.4 RequestFailure.value

This metric specifies the accumulated number of client requests forwarded by Oracle Web Cache that resulted in an error.

Business Activity Monitoring

This chapter describes Oracle Business Activity Monitoring metrics.

17.1 Report Server Active Data Page

Provides information about the report server active data page metrics.

17.1.1 Open Connections

Number of open connections to the Report Server.

17.1.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

17.1.1.2 User Action

No action required.

17.1.2 Open Reports

Number of reports open in the Report Server.

17.1.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

17.1.2.2 User Action

No action required.

17.1.3 Open Viewsets

Number of open viewsets in the Report Server.

17.1.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

17.1.3.2 User Action

No action required.

17.1.4 Activity Message Rate (per sec)

17.1.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

17.1.4.2 User Action

No action required.

17.1.5 Average Time To Open Viewsets (in sec)

Average length of time in seconds to open a viewset. This is purely from the server's perspective and not from the client's perspective.

17.1.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

17.1.5.2 User Action

No action required.

17.2 Response

Shows if the BAM Web is available or not.

17.2.1 UpDown Status

Displays the current status of the Oracle BAM Report Server. A value of 1 indicates that the server is running, and any other value indicates that it is not running.

17.2.1.1 User Action

No action required.

Business Activity Monitoring Server

This chapter describes Oracle Business Activity Monitoring Server metrics.

18.1 ADC Client Requests For All DataObjects

Client requests include all of the requests made to the Oracle BAM Server EJBs. There will be a lot of requests made when opening a single report (for permissions, the report metadata, the views data, and so on).

18.1.1 Client Requests In Progress

Client requests include all of the requests made to the Oracle BAM Server EJBs. There are several requests made when opening a single report (for permissions, the report metadata, the views data, and so on).

18.1.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.1.1.2 User Action

No action required.

18.1.2 Client Requests Total

Client requests include all of the requests made to the Oracle BAM Server EJBs. There are several requests made when opening a single report (for permissions, the report metadata, the views data, and so on).

18.1.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.1.2.2 User Action

No action required.

18.2 ADC Waiting Threads For All DataObjects

The number of Active Data Cache server threads waiting to access this data object.

18.2.1 Deletes Per Second

Rate of delete operations per second on the data object in the last 5 minutes.

18.2.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.1.2 User Action

No action required.

18.2.2 Inserts Per Second

Rate of insert operations per second on the data object in the last 5 minutes.

18.2.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.2.2 User Action

No action required.

18.2.3 Updates Per Second

Rate of update operations per second on the data object in the last 5 minutes.

18.2.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.3.2 User Action

No action required.

18.2.4 Upserts Per Second

Rate of upsert operations per second on the data object in the last 5 minutes.

18.2.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.4.2 User Action

No action required.

18.2.5 Deletes Per Second

Rate of delete operations on the data object in the last 5 minutes.

18.2.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.5.2 User Action

No action required.

18.2.6 Inserts Per Second

Rate of insert operations on the data object in the last 5 minutes.

18.2.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.6.2 User Action

No action required.

18.2.7 Updates Per Second

Rate of update operations on the data object in the last 5 minutes.

18.2.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.7.2 User Action

No action required.

18.2.8 Upserts Per Second

Rate of upsert operations on the data object in the last 5 minutes.

18.2.8.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.8.2 User Action

No action required.

18.2.9 ADC Waiting Threads For All DataObjects

The number of Active Data Cache server threads waiting to access this data object.

18.2.10 Waiting Threads For All DataObjects

The number of Active Data Cache server threads waiting to access this data object.

18.2.10.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.10.2 User Action

No action required.

18.2.11 ADC Waiting Threads For All DataObjects

The number of Active Data Cache server threads waiting to access this data object.

18.2.12 Waiting Threads For All DataObjects

The number of Active Data Cache server threads waiting to access this data object.

18.2.12.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.12.2 User Action

No action required.

18.2.13 ADC Batch Size For All DataObjects

Average number of rows in the given batch.

18.2.14 Current Delete Batch Size

Average number of rows in the delete batch.

18.2.14.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.14.2 User Action

No action required.

18.2.15 Current Insert Batch Size

Average number of rows in the insert batch.

18.2.15.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.15.2 User Action

No action required.

18.2.16 Current Update Batch Size

Average number of rows in the update batch.

18.2.16.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.16.2 User Action

No action required.

18.2.17 Current Upsert Batch Size

Average number of rows in the upsert batch.

18.2.17.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.17.2 User Action

No action required.

18.2.18 Current Update Batch Size

Average number of rows in the update batch.

18.2.18.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.18.2 User Action

No action required.

18.2.19 Current Insert Batch Size

Average number of rows in the insert batch.

18.2.19.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.19.2 User Action

No action required.

18.2.20 Current Delete Batch Size

Average number of rows in the delete batch.

18.2.20.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.20.2 User Action

No action required.

18.2.21 Current Upsert Batch Size

Average number of rows in the upsert batch.

18.2.21.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.21.2 User Action

No action required.

18.2.22 Current Delete Batch Size

Number of active viewsets.

18.2.22.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.22.2 User Action

No action required.

18.2.23 Current Delete Batch Size

Number of all of the active viewsets.

18.2.23.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.23.2 User Action

No action required.

18.2.24 Current Insert Batch Size

Number of all of the open viewsets.

18.2.24.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.24.2 User Action

No action required.

18.2.25 Current Insert Batch Size

Number of open viewsets.

18.2.25.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.25.2 User Action

No action required.

18.2.26 Waiting Threads For All DataObjects

The number of Active Data Cache server threads waiting to access this data object.

18.2.26.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.2.26.2 User Action

No action required.

18.3 Active Data Cache Transaction Information

Transactions shows, number of transactions on BAM Data Objects e.g. if you add a row in BAM DO and save it (commit), its one transaction.

18.3.1 Committed Transactions

Number of transactions on Oracle BAM data objects.

18.3.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.3.1.2 User Action

No action required.

18.3.2 Completed Transactions

Number of completed transactions on Oracle BAM data objects.

18.3.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
----------------	----------------------

All Versions	Every 300 Hours
--------------	-----------------

18.3.2.2 User Action

No action required.

18.3.3 Open Transaction

Number of open transactions on Oracle BAM data objects.

18.3.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.3.3.2 User Action

No action required.

18.3.4 Rolledback Transactions

If there is an invalid operation on a data object, the Oracle BAM Server will roll back the transaction.

18.3.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.3.4.2 User Action

No action required.

18.4 EMS Metrics

Provides information about EMS metrics.

18.4.1 Total number of received messages.

The total number of messages that have been received by all Oracle BAM Enterprise Message Sources.

18.4.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.1.2 User Action

No action required.

18.4.2 Total number of rejected messages.

The total number of messages that have been rejected by all Oracle BAM Enterprise Message Sources.

18.4.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.2.2 User Action

No action required.

18.4.3 Total delete operations.

The total number of successful Delete operations that have been performed by all Oracle BAM Enterprise Message Sources.

18.4.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.3.2 User Action

No action required.

18.4.4 Total insert operations.

The total number of successful Insert operations that have been performed by all Oracle BAM Enterprise Message Sources.

18.4.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.4.2 User Action

No action required.

18.4.5 Total update operations.

The total number of successful Update operations that have been performed by all Oracle BAM Enterprise Message Sources.

18.4.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.5.2 User Action

No action required.

18.4.6 Total upsert operations.

The total number of successful Upsert operations that have been performed by all Oracle BAM Enterprise Message Sources.

18.4.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.6.2 User Action

No action required.

18.4.7 Total number of received messages.

The total number of messages that have been received by all Oracle BAM Enterprise Message Sources.

18.4.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.7.2 User Action

No action required.

18.4.8 Total number of rejected messages.

The total number of messages that have been rejected by all Oracle BAM Enterprise Message Sources.

18.4.8.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.8.2 User Action

No action required.

18.4.9 Delete operations rate.

Rate of total number of successful Delete operations that have been performed by all Oracle BAM Enterprise Message Sources.

18.4.9.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.9.2 User Action

No action required.

18.4.10 Insert operations rate.

Rate of total number of successful Insert operations that have been performed by all Oracle BAM Enterprise Message Sources.

18.4.10.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.10.2 User Action

No action required.

18.4.11 Update operations rate.

Rate of total number of successful Update operations that have been performed by all Oracle BAM Enterprise Message Sources.

18.4.11.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.11.2 User Action

No action required.

18.4.12 Upsert operations rate.

Rate of total number of successful Upsert operations that have been performed by all Oracle BAM Enterprise Message Sources.

18.4.12.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.12.2 User Action

No action required.

18.4.13 Total received messages.

The total number of messages that have been received by the Oracle BAM Enterprise Message Source.

18.4.13.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.13.2 User Action

No action required.

18.4.14 Total rejected messages.

The total number of messages that have been rejected by the Oracle BAM Enterprise Message Source.

18.4.14.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.14.2 User Action

No action required.

18.4.15 Delete operations rate.

Number of successful delete operations per second in the last 5 minutes.

18.4.15.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.15.2 User Action

No action required.

18.4.16 Insert operations rate.

Number of successful insert operations per second in the last 5 minutes.

18.4.16.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.16.2 User Action

No action required.

18.4.17 Update operations rate.

Number of successful update operations per second in the last 5 minutes.

18.4.17.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.17.2 User Action

No action required.

18.4.18 Upsert operations rate.

Number of successful upsert operations per second in the last 5 minutes.

18.4.18.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.18.2 User Action

No action required.

18.4.19 Total received messages.

The total number of messages that have been received by the Oracle BAM Enterprise Message Source.

18.4.19.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.19.2 User Action

No action required.

18.4.20 Total rejected messages.

The total number of messages that have been rejected by the Oracle BAM Enterprise Message Source.

18.4.20.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.20.2 User Action

No action required.

18.4.21 Total delete operations.

The total number of successful Delete operations that have been performed by the Oracle BAM Enterprise Message Source.

18.4.21.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.21.2 User Action

No action required.

18.4.22 Total insert operations.

The total number of successful Insert operations that have been performed by the Oracle BAM Enterprise Message Source.

18.4.22.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.22.2 User Action

No action required.

18.4.23 Total update operations.

The total number of successful Update operations that have been performed by the Oracle BAM Enterprise Message Source.

18.4.23.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.23.2 User Action

No action required.

18.4.24 Total upsert operations.

The total number of successful Upsert operations that have been performed by the Oracle BAM Enterprise Message Source.

18.4.24.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.4.24.2 User Action

No action required.

18.5 Event Engine General Information

Provides information about event engine general information metrics.

18.5.1 Active Rules

Number of active alert rules in the Event Engine.

18.5.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.5.1.2 User Action

No action required.

18.5.2 Action Queue Size

Number of Oracle BAM Event Engine actions waiting to be processed.

18.5.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.5.2.2 User Action

No action required.

18.5.3 Alert History Queue Size

Number of alert history items to be saved.

18.5.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.5.3.2 User Action

No action required.

18.5.4 Rule Fire Queue Size

Number of events fired waiting to be processed.

18.5.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.5.4.2 User Action

No action required.

18.5.5 Event Engine Activity Rule Fire Rate

Number of events fired per second in Event Service.

18.5.6 Events Accepted Rate (per sec)

Number of events accepted per second.

18.5.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.5.6.2 User Action

No action required.

18.5.7 Events Fire Rate (per sec)

Number of events fired per second in the Oracle BAM Event Engine.

18.5.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.5.7.2 User Action

No action required.

18.5.8 Events Rejected Rate (per sec)

Number of events per second rejected by the Oracle BAM Event Engine.

18.5.8.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.5.8.2 User Action

No action required.

18.6 Active Data Reports Information

Total Number of Active ActiveDataReports

18.6.1 Active Data Reports

Total number of active reports containing active data.

18.6.1.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.6.1.2 User Action

No action required.

18.6.2 Active Data Sessions

Total number of sessions with active data.

18.6.2.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.6.2.2 User Action

No action required.

18.6.3 Outgoing Message Rate (per sec)

Number of messages delivered per second by the Report Cache to all of its clients.

18.6.3.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.6.3.2 User Action

No action required.

18.6.4 File Count

Total number of files currently managed by FileBasedPersistenceManager.

18.6.4.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.6.4.2 User Action

No action required.

18.6.5 Element Count

Total number of elements currently managed by InMemoryPersistenceManager.

18.6.5.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.6.5.2 User Action

No action required.

18.6.6 Viewset Average Open Time (in sec)

Average time period in seconds to open a view set.

18.6.6.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.6.6.2 User Action

No action required.

18.6.7 Independent Viewsets

Number of open independent viewsets. Independent (unshared) viewsets are neither shared or inherited. See *Monitoring the Report Cache* for more information.

18.6.7.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.6.7.2 User Action

No action required.

18.6.8 Open Viewsets

Total number of viewsets opened by the Report Cache.

18.6.8.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.6.8.2 User Action

No action required.

18.6.9 Shared Viewsets

Number of open shared viewsets. Shared viewsets are the parent viewsets (opened first). See Monitoring the Report Cache for more information.

18.6.9.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.6.9.2 User Action

No action required.

18.6.10 Sharing Viewsets

Number of open sharing viewsets. Sharing (inherited) viewsets are the child viewsets (opened later). See Monitoring the Report Cache for more information.

18.6.10.1 Metric Summary

The following table shows how often the metric's value is collected.

Target Version	Collection Frequency
All Versions	Every 300 Hours

18.6.10.2 User Action

No action required.

18.7 Response

Displays the availability of the Oracle BAM Server.

18.7.1 UpDown Status

Displays the current status of the Oracle BAM Server. A value of 1 indicates that the server is running, and any other value indicates that it is not running.

18.7.1.1 User Action

To start or stop the Oracle BAM Server, use the Start Up and Shut Down controls in the Oracle BAM Server menu.

