ORACLE ENTERPRISE MANAGER **12**c

An Oracle White Paper

September, 2012

Enterprise Manager 12c Cloud Control: Monitoring and Managing Oracle Coherence for High Performance

ORACLE

Executive Overview

Oracle Enterprise Manager is Oracle's integrated enterprise IT management product line and provides the industry's first complete cloud lifecycle management solution. Oracle Enterprise Manager's Business-Driven IT Management capabilities allow you to quickly set up, manage and support enterprise clouds and traditional Oracle IT environments from applications to disk. Oracle Enterprise Manager allows customers to achieve:

- *Best service levels for traditional and cloud applications* through management from a business perspective including Oracle Fusion Applications

- *Maximum return on IT management* investment through the best solutions for intelligent management of the Oracle stack and engineered systems

- *Unmatched customer support experience* through real-time integration of Oracle's knowledgebase with each customer environment

Introduction

Oracle Enterprise Manager's Fusion Middleware Management solutions provide full-lifecycle management for Oracle WebLogic, SOA suite, Coherence, Identity Management, WebCenter Suite, and Business Intelligence Enterprise Edition. Oracle Enterprise Manager provides a single console to manage these assets from a business and service perspective, including user experience management, change and configuration management, patching, provisioning, testing, performance management, business transaction management, and automatic tuning for these diverse environments.

Abstract Data Layer

Oracle Coherence is an in-memory data grid solution that provides linear scalability, reliability, and high performance to applications. Enterprises are increasingly designing mission critical applications around Oracle Coherence as an abstract data layer. Oracle Fusion Middleware provides the runtime engine and platform for mission-critical Java EE, SOA and middleware applications. Oracle Coherence is designed to work seamlessly with Oracle WebLogic Server and Oracle SOA Suite to support mission critical business applications. In some cases, enterprises share a single Oracle Coherence cluster across a host of applications in a business unit while in other cases they prefer a one-to-one relationship between an application and a cluster.

Since direct revenue is routinely impacted by business applications, monitoring and managing the service levels of these applications has become that increasingly more critical. Traditionally administrators have to rely on various scripts to manage and monitor the middleware environment. Some use different point solutions to manage different tiers of the applications stack. Given all the investment, the risk involved in managing IT systems using scripts and point solutions is too high. Therefore, for effective application performance monitoring, enterprises need a solution that provides performance visibility into all tiers of applications stack, proactive notifications of potential issues, an ability to perform deep diagnostics in each tier, and an ability to perform cross-tier diagnostics. These facets are necessary across all stages of the application lifecycle.

Managing Oracle Coherence with Oracle Enterprise Manager

Oracle Coherence provides JMX interface that exposes valuable information about the runtime health of the cluster via several MBeans. These MBeans can be accessed using JConsole or any JMX based point solution. However, just showing the MBean values doesn't provide all the useful information. The MBeans are organized based on nodes (or JVMs) for different Oracle Coherence resources. It is important to intelligently collate these MBeans, and aggregate where necessary, to provide meaningful monitoring data. E.g. JMX exposes cache metrics per node at that point of time. The real value is provided when administrator is able to see aggregation of cache metrics across all the nodes over period of time (e.g. 2 hours, 24 hours, etc). There are several cases where such aggregation and trending is necessary to identify performance hotspot. Oracle Coherence is a multi-threaded environment. Although, JMX provides plenty of valuable information about the health of the node and sophisticated tools can extract more value from it, there are some cases where deep visibility into the JVM runtime is necessary. Similarly, JMX doesn't provide much help for developers to optimize the Oracle Coherence queries ahead of time. Apart from monitoring and diagnostics there are several other aspects to administration which are key to success of production deployment i.e. configuration management and cache data or operations management. The sections below provide details into Oracle Enterprise Manager's comprehensive monitoring and management solution for Oracle Coherence.

Oracle Coherence Topology and Health Dashboard

After discovering your Oracle Coherence clusters with Oracle Enterprise Manager, Oracle Enterprise Manager immediately starts monitoring the environment with a predefined set of status and performance metrics. Oracle Enterprise Manager performs several aggregations and correlations on the raw JMX statistics. Based on the relationships between the components Oracle Enterprise Manager creates a topology view. The topology shows the cluster, caches, nodes, the Hosts where the nodes are running, and Oracle WebLogic servers (for HTTP Session caching). Administrators can drill down to get a topology view of any particular cache or node. The topology view can show alerts, incidents and key performance indicators for any entity in the topology as shown in Figure 1. These alerts are automatically generated by Oracle Enterprise Manager based on the performance thresholds. In production, administrators can leverage several framework features of Oracle Enterprise Manager. For instance, custom monitoring template can be created to set thresholds on key metrics. The template

can then be applied to all nodes in a cluster. This simplifies setting of thresholds in a large deployment. Similarly, incident rules can be defined to generate notifications, such as SNMP or email, in response to an alert. Some availability and performance thresholds are defined out-of-box. For example, an alert is generated when a cache server node goes down or when service becomes ENDANGERED.
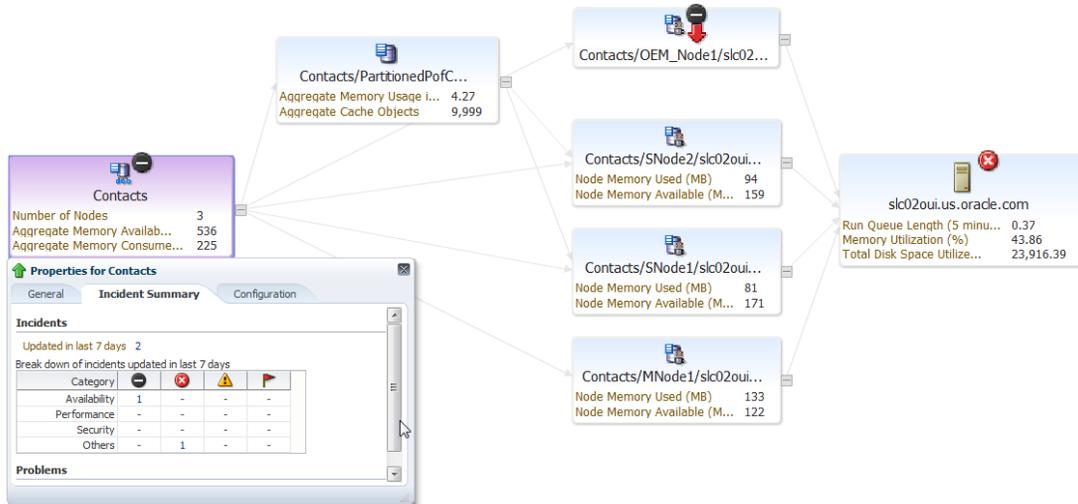

Figure 1 – Topology view provides high level visibility into the cluster and alerts across different resources.

Oracle Enterprise Manager provides out-of-box alert setting for ENDANGERED services. This alert indicates that a crash of any of the storage enabled nodes could result in loss of data. Ideally the service status should be MACHINE-SAFE. This status indicates that the data loss will not happen even if a machine crashes. The NODE-SAFE status indicates that data will not be lost if a node crashes but data loss can take place if a machine crashes. As can be seen in the Figure 2, the Services table shows all the services along with their type (e.g. Invocation, Distributed, Replicated, etc), status, and number of storage enabled nodes, number of endangered nodes, and number of active transactions, if any.

The cluster home page dashboard shows several vital details about the overall health of the cluster. Administrators get a complete picture about bad caches, nodes uptime, top hosts based on CPU or Memory, etc. Oracle Enterprise Manager highlights top caches in the cluster that have lowest Hits to Gets Ratio (%). Such data is not available from raw JMX. Each 'miss' on 'get' operation leads to overhead on the cache to communicate with backend Database and high response time for the application. Administrators can drill down to such a cache for more detailed diagnostics.
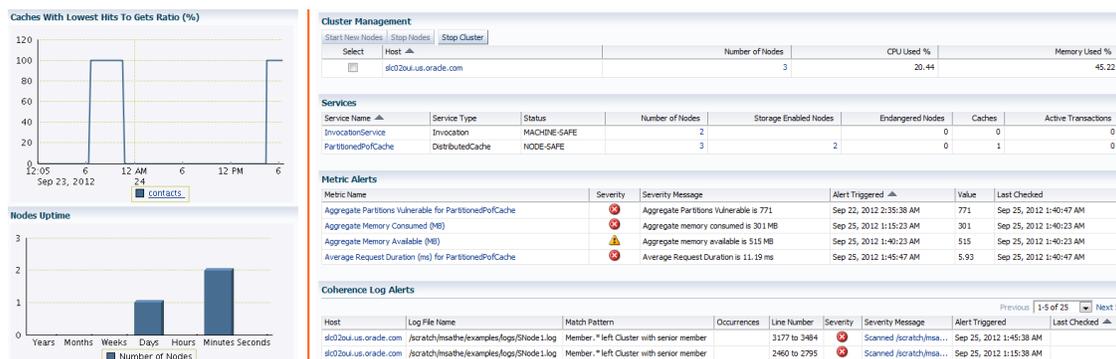
Figure 2 – Oracle Coherence cluster home page provides a dashboard view into overall health of the cluster such as top caches, service status, hosts, etc.

Log Alerts

Not all events are exposed via JMX. Some events are registered in the log files. By default, Oracle Coherence uses 'stdout' as a log destination. If Oracle Coherence is running on an Oracle WebLogic server then the container log will be used for Oracle Coherence log messages. However, if the Oracle Coherence is running as a standalone Java server then administrators need to specify the log file path using tangosol.coherence.log system property. The log level can be set using tangosol.coherence.log.level system property. Oracle Enterprise Manager provides log alert feature than enables administrators to define a pattern on a log file and get an alert when the pattern is found in the log. Administrators can also define number of times the pattern has to match before an alert is generated. Moreover, similar to other alerts, administrators can define notification rules to get an email, generate SNMP trap, etc.

Monitoring and Diagnostics

From an operational point of view there is a difference between monitoring and diagnostics. Monitoring involves checking the cluster status and health using alerts and performance trends for a set of important health metrics. Oracle Enterprise Manger allows organizations to create a custom performance view with such metrics and use it as the default page for the console. Metrics such as number of storage enabled nodes and total number of nodes (including process nodes) indicate how stable the cluster is. Aggregate memory consumed and aggregate memory available metrics can be used to make sure the cluster is not running out of storage capacity. Send/receive success rate per minute metrics show the network performance based on every sample.
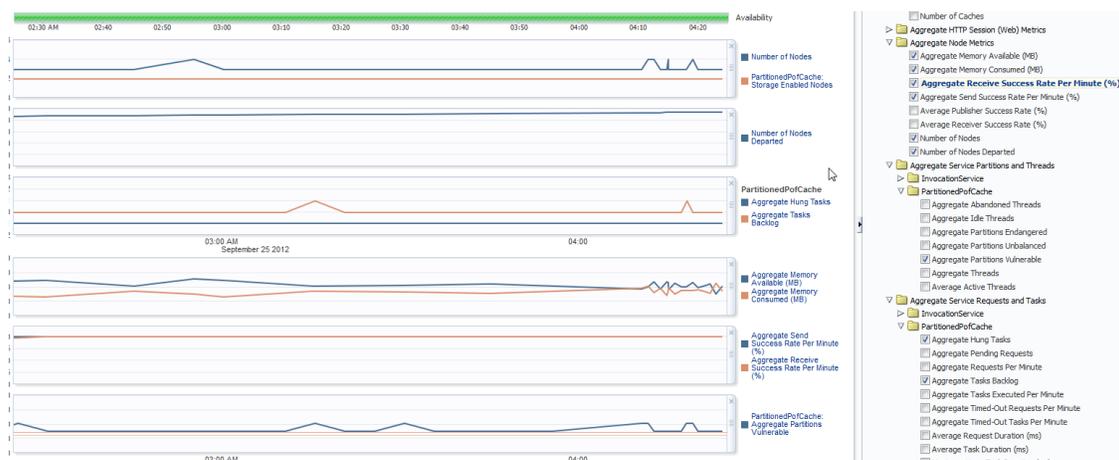
Figure 3 – Cluster level monitoring view showing performance trends of metrics indicating overall health

Monitoring solution should have an ability to separate one-time events from potential performance issues. Also, administrators should not have to watch the console 24x7. Oracle Enterprise Manager provides ability to set thresholds in such a way that alert will be generated only if the metric value crosses the threshold repeatedly. Although, in some cases, administrators may choose to get notified at the first occurrence of an event. Historical monitoring of the metrics allows administrators to observe the performance charts over period of time with respect to the thresholds. The data collected from the trend analysis can be used to fine tune configuration and plan capacity. Figure 3 shows the trend of the selected metrics that can be used to get high level visibility into the health of the cluster. You can overlay related metrics on the same charts for better analysis.

Diagnostics, on the other hand, typically involves finding root-cause of application slowness, performance bottlenecks in a cache, set of nodes, hosts, etc. Diagnostics will typically be triggered by alerts. In some cases Oracle Coherence queries used in application may not be efficient and may require turning. Database slowness can severely impact the persistence of the Oracle Coherence. Often, Oracle Coherence administrators don't have visibility into the Database tier. Diagnosing cross-tier issues in such cases is a real challenge. While diagnosing performance issues administrators may also need to check the configuration of the Oracle Coherence as well as the Host. Bad configuration of JVMs, cluster services and Hosts can lead to availability and performance issues for applications. It is imperative for administrators to consider these aspects for managing mission critical Oracle Coherence applications. Unlike point solutions, Oracle Enterprise Manager provides end-to-end management, monitoring and diagnostics for Oracle Coherence. Administrators can slice and dice the performance metrics across various resources such as JVMs, caches, services, Hosts, and Database. JVM Diagnostics is particularly useful in real-time diagnostics as it collects JVM runtime information at a high sampling rate (even at 2 seconds interval).

Cluster Stability

Oracle Coherence can be thought of as a data cloud that enables nodes (JVMs) and even machines to dynamically join or leave the cluster. Oracle Coherence treats each node that joins the cluster as a new node. Oracle Coherence doesn't keep an identity of the departed or dead node. That means, even if the crashed node is started again with exactly same configuration and name the Oracle Coherence treats it as a completely new node. Although this provides flexibility to Oracle Coherence, it poses a big challenge from management point of view. If management tool doesn't recognize the node across the lifecycle then administrators will not be able to find the performance and configuration of the node across lifecycle. Administrators need to know about the nodes that are leaving the cluster or crashing, their performance before their crash, changes in their configuration, etc. Without knowing the performance and configuration of the node across the lifecycle administrators can't effectively address the cluster stability issues. Oracle Enterprise Manager provides a unique ability to track historical performance and configuration of the nodes across lifecycles. Administrators can track every node, time it went down, time it came up again and all the performance metrics across the selected timeline. JVM Diagnostics also maintains the history of the JVM's runtime performance across lifecycles. Most of the tools don't provide such ability to track performance of the node across lifecycles.

Node Memory Performance

One of the critical parameters that affect every Java program is the heap. Heap utilization and garbage collection (GC) directly impact the performance of the nodes. Typically, the heap utilization chart should look like a saw tooth. The fall from high to low level of heap utilization in such a chart indicates a GC cycle. The heap utilization chart may remain stable if the cache data is not changing (e.g. read-only caches). However, the JVM may eventually crash due to 'out of memory' error if the heap utilization continues to grow and GC is unable to clean up the heap. There are two types of GCs, minor and major (sometimes called as 'full GC'). Minor GC is not a 'stop all' GC and has very low overhead, it clears the garbage objects from new generation section of the heap. On the other hand, Major or Full GC is a 'stop all' GC and inflicts very high overhead. No other request is served by the JVM when the Major GC kicks in and that means that the node is not able to communicate with rest of the cluster while the GC is on. If the Major GC occurs frequently on a node, Oracle Coherence will most likely drop that node from the cluster. Oracle Enterprise Manager collects GC performance metrics such as GC overhead (%), Minor GC (invocations/min) and Major GC (invocations/min) for each JVM. Administrators can also compare the trends of these metric across several nodes. This helps in identifying heap issues that are common across a cluster vs. the issues that are specific to a particular node.
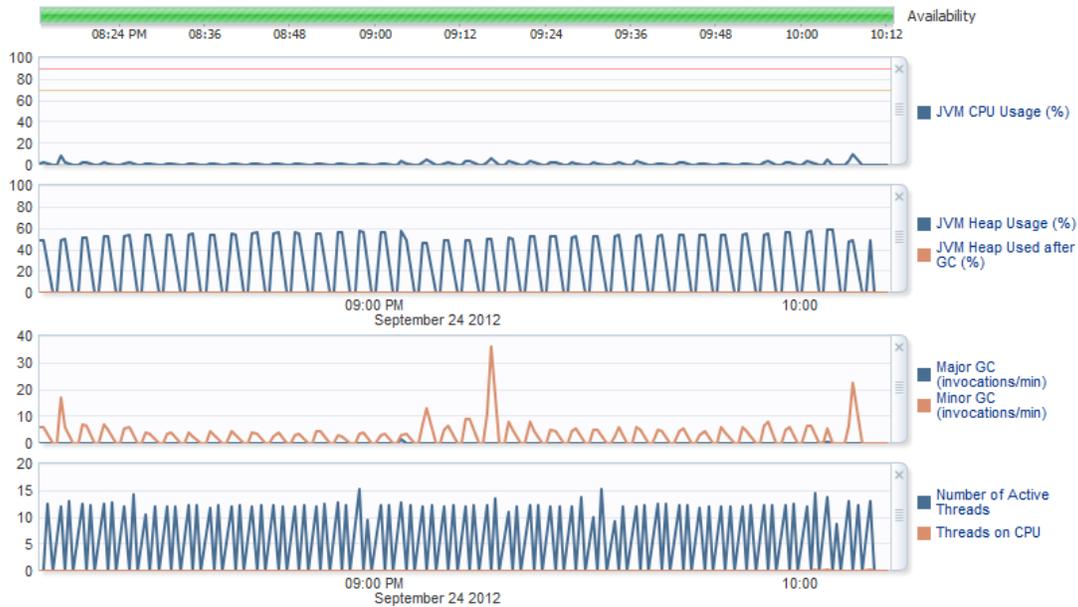
Figure 4 – Node heap performance charts along with CPU usage (%) and number of active threads

Planning Storage Capacity

A node can host multiple caches. Administrators can use Units metric to find out the memory consumed by each cache on a node. If the UnitCalculator is set to BINARY the Units metric indicates the number of bytes consumed by cache entries. Administrator can use HighUnits value as a performance threshold for Units metric. HighUnits is configured per caching scheme to limit the number of units that can be placed in the cache before pruning occurs. Please note that the Units metric doesn't indicate the memory consumed by the indexes. It is extremely important to consider the following parameters while planning storage capacity

- Total size of the cache entries * 2 (assuming one backup copy)

- Size of the indexes

- JVM footprint

- Max heap size of the JVM

- Number of nodes (n) on a machine

- Available memory on a machine and

- Number of machines (m)

To ensure high availability and performance service levels there should be one extra node (n + 1) on every machine to absorb an impact of a node crash and one extra machine (m + 1) to handle a machine crash. Max JVM heap size is defined by –Xmx command line parameter. To avoid overhead of incremental heap expansion the initial heap size (-Xms) should be set equal to (-Xmx). In general, the total heap utilization of the node should remain under 70%. Beyond this level GC starts having an adverse impact on the performance. Also, enough free memory should be available on the machine after taking into account all running processes. It is important to note that the swapping significantly impacts the Oracle Coherence performance. If a node gets swapped out of the memory it could be removed from the cluster for unresponsiveness. Oracle Enterprise Manager's host monitoring provides visibility into the memory performance including swapping and paging metrics. Administrators can use performance thresholds to be proactive and avoid cluster stability issues that can occur due to host resource constraints. Additionally, Oracle Enterprise Manager shows, real-time and historical, top ten processes (ordered by memory and CPU) along with their resource utilization on the Host.

Figure 5 – Host memory performance and top 10 processes ordered by memory consumption

Network Performance Bottlenecks

Monitoring network performance of the cluster is very crucial. Administrators can monitor the 'publisher/receiver success rate since the start of the node' is exposed by JMX out-of-box. But more useful metrics are packets send/receive success rate per minute which is based on delta of the samples. Such metrics provided by Oracle Enterprise Manager add real value from operations point of view. Ideally the send/receive success rate should be close to 100%. Drop in the values of these metrics indicate some performance bottleneck. The network bottlenecks could be caused by high network latency or high packet drop rate. Prolonged GC on a node can also cause the send/receive success rate to go down. Oracle Enterprise Manager collects several such network performance metrics for each node. Send Queue Size indicates number of packets in a queue including those for which the acknowledgment has not yet been received. Packets that do not receive an acknowledgment within ResendDelay interval will be automatically resent. Set ResendDelay with careful observation of the

network performance. 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. Use the datagram test to find the network performance and fine tune the configuration based on that. Observe the PacketDeliveryEfficiency metric, low value is an indication that there is a high rate of unnecessary packet retransmissions.

Network Performance of the Host

Network performance issues can also be related to the configuration and performance of the Host. Two important parameters to watch out for are - buffer size of the OS and Maximum Transfer Units (MTU) of the network card. The OS buffers must be large enough to handle incoming network traffic while the node JVM is paused during garbage collection. As opposed to point solutions, Oracle Enterprise Manager provides out-of-box monitoring for performance of the Host as well as visibility into the Host configuration. E.g. for Linux environment Oracle Enterprise Manager captures key OS configuration metrics such as 'net.core.wmem_max' and 'net.core.rmem_max' which indicate the max OS buffer size. Similarly, Oracle Enterprise Manager collects MTU configuration metric, which indicates maximum packet size (bytes) that can be transmitted by the network card. Having an Oracle Coherence packet size that does not fill the MTU will result is an underutilization of the network. Using Oracle Enterprise Manager, administrators can create a performance view combining node and Host network performance metrics to get a holistic picture of the system.

Figure 6 – Network performance charts for both, node and Host

Optimize Query Performance

Badly written queries can slow down the application even if the Oracle Coherence nodes are performing well. It is important to know the performance of the Oracle Coherence queries on the

cache data. Application developers should know which filters are more effective than others. A filter is more effective if it eliminates maximum number of keys from search. Administrators can add indexes based on that. It should be noted that an index consumes memory and, as mentioned earlier, it is not calculated as part of Units metric for the cache. Oracle Enterprise Manager collects several performance metrics for Oracle Coherence queries. Figure 7 shows the performance of optimized and un-optimized queries for each cache on a node. These metrics are also aggregated at a cache level from all the nodes where cache is running.

Cache Name ▲∨	Average Non Optimized Query Execution Time (ms)	Average Optimized Query Execution Time (ms)	Max Query Duration (ms)	Total Non Optimized Queries Executed Per Minute	Total Optimized Queries Executed Per Minute	QueryContentions are {0} per minute
contacts	1	4	49	5.62	56.25	0

Figure 7 – Query optimization report for all caches on the node

Optimized query is the query that is fully resolved using indexes. Un-optimized query is the query that could not be resolved or is partially resolved using indexes. The QueryContentions metric shows number of times (per minute) a query had to be re-evaluated due to a concurrent update. This statistic provides a measure of an impact of concurrent updates on the query performance. If the number of queries is Q and the number of contentions is C then the expected performance degradation factor should be no more than $(Q + C)/Q$.

Oracle Enterprise Manager supports Query Explain Plan, Query Trace and Add/Remove Index features. Particularly, Query Explain Plan and Query Trace features are extremely useful for developers. The Query Explain Plan feature is used to find out an estimated cost of the query filters. Developers can determine the filters for which the indexes were found and sequence of the filters along with the estimated cost of each filter. The Query Trace feature actually runs the query and shows the actual duration (ms) and effectiveness of each query filter. Index will give maximum benefit if it is able to drastically reduce the key set for a certain filter. Oracle Coherence first selects the filters for which there exists an index. Once the indexed filters are applied the remaining key set is then deserialized and other filters are applied on that. Oracle Enterprise Manager provides centralized console to add and remove indexes on cache. Administrators should always be judicious while adding an index on a cache. Add indexes for the filters which provide maximum effectiveness. Adding an index for a filter that provides low effectiveness may not be worth the resource overhead required to create an index. Trying the queries before moving to production and creating the indexes accordingly drastically improves the application performance.

Cache Load and Performance

Oracle Enterprise Manager provides deep visibility into cache load and performance. The data collected from cluster nodes is aggregated and presented at a cache level. Administrators can plot the trends of get/put, hits/misses, etc operations and their performance (ms). As described earlier, the caches having lowest Hits to Gets Ratio (%) are highlighted on the cluster home page. The

HitProabability indicates the probability that the next invocation will be a hit. Load and performance statistics of the cache-store operations are also captured out-of-box.

It should be noted that caches are configured via a service in a cache configuration file. Each storage enabled node that participates in a service holds partition(s) of the cache defined in that service. Service provides set of threads (also called as service threads) to handle various operations such as get, put, read, write, etc. Data affinity related design issues should be considered while configuring a service. Data affinity improves performance of several operations by putting related cache entries (based on keys, not values) in the same partition. Affinity may span multiple caches provided they are managed by the same cache service.

Oracle Enterprise Manager collects several service performance metrics for each node. Such detailed information is typically used for diagnostics rather than regular monitoring. For each node you can see the number of primary and backup partitions. You can see the number of partitions that are endangered, vulnerable and unbalanced. TaskBacklog shows the size of the backlog queue indicating the number of tasks scheduled to be executed by one of the service threads. Administrators can use this metric to correlate with the number of threads in the service thread pool. This will help administrators to adjust the thread pool of the service.

Administrators can also drill down to JVM Diagnostics for threads or heap analysis of the nodes where the cache is running. Figure 8 shows various cache load and performance metrics.
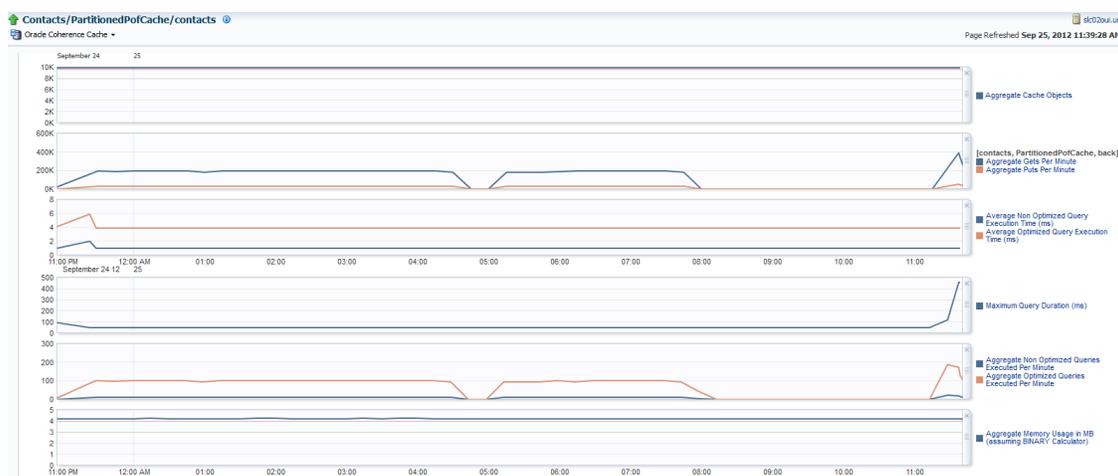
Figure 8 – Load and performance metrics of the cache.

Cache Data Management

A common cache data management problem occurs when an administrator is forced to shut down a cluster for maintenance activities. Once the cluster is shutdown it loses all cached data which forces the administrator to later 'warm up' the cluster when it comes back online. This can result in additional maintenance downtime which often impacts the overall availability of applications that are dependent on the cluster. Oracle Enterprise Manager solves this problem by providing a centralized cache data

management feature that allows administrators to export cache data to a file before bringing down the cluster for maintenance and later importing the same data. Oracle Enterprise Manger also enables administrators to run ad-hoc queries to view or update data in a cache. The queries can be stored in Oracle Enterprise Manager repository. These cache operations are protected by credentials.

User Defined Metrics

Oracle Enterprise Manager supports user defined metrics using Metric Extensions. Metric Extension can be added to Oracle Coherence target using JMX. Administrators can select a particular MBean using the MBean browser. They can add MBean attributes that have multiple entries (array). Oracle Coherence also supports addition of custom MBeans to the MBean server. Moreover, administrators can add Metric Extension for Host using variety of supported mechanisms.

JVM Diagnostics for Deep JVM Runtime Visibility

Threads Monitoring

Administrators can trace live threads and find the blocking issues using JVM Diagnostics. JVM Diagnostics also stores thread samples in the repository for historical analysis. This helps administrators to determine where the threads are spending more time – IO, Network, Lock, etc. Another unique feature of the JVM Diagnostics is its ability to correlate Java calls to Database. Administrators can see Database wait states and the SQL calls made by Java. If the Database is managed by the same Oracle Enterprise Manager then administrators can truly bridge the gap between middleware and Database by simply drilling down from Java to the Database for diagnosing SQLs further with Database Diagnostics.

Figure 9 – Thread state transition and ability to search threads by state (DB wait, IO wait, etc)

Differential Heap Analysis

Using Oracle Enterprise Manager, administrators can take heap dumps and analyze the classes for memory consumption. Administrators can easily perform differential heap analysis of heap dumps taken at two different times. Oracle Enterprise Manager quickly pinpoints the cause of memory leaks by showing the classes and their fields that have grown in size. Without Oracle Enterprise Manager finding memory leaks could take a significant amount of man-hours resulting in higher maintenance costs and lower application service levels.
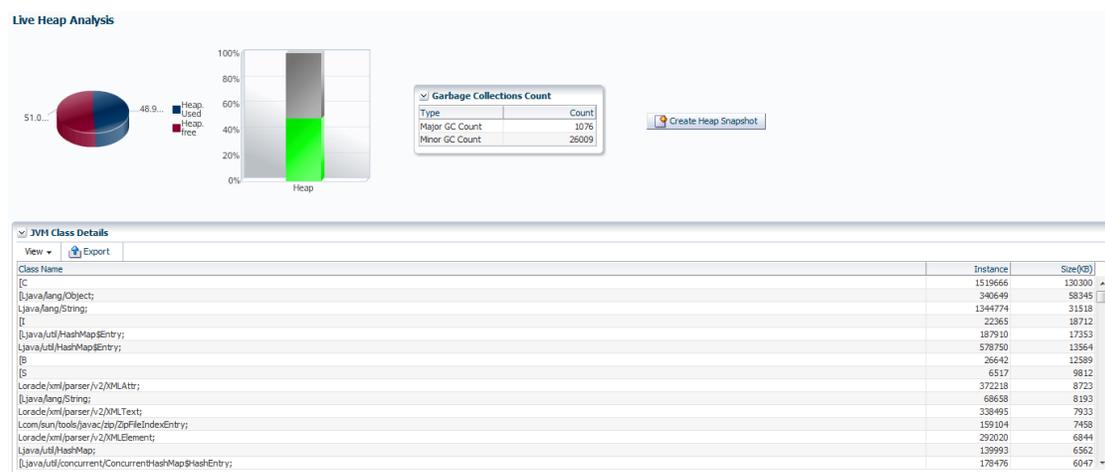
Figure 10 – Analyze both live and historical heap details and compare different time periods

Configuration and Change Management

Many performance degradation issues in production application environments occur due to unauthorized configuration change. Moreover, unlike Oracle WebLogic or SOA server, the Oracle Coherence node is not a managed entity and doesn't have separate home for each cache or a node. Managing configuration files for such products is always a challenge.

Automating Discovery and Tracking Assets

Traditionally, tracking assets and configuration items across the IT environment relies upon the knowledge of key individuals, time-consuming ad-hoc processes, and manual, error-prone spreadsheets. In contrast, Oracle Enterprise Manager automatically collects deep configuration information about Oracle Coherence as well as its underlying hardware and operating system. Configuration templates or blueprints specifying what configuration items should be collected for Oracle Coherence and its underlying operating system are available out-of-the-box and can be customized to collect only the relevant configuration items that IT personnel require. Following are some of the configuration parameters collected out-of-box for each node

- Runtime configuration – all Java runtime arguments (-D parameters), details of JVM (name, vendor, and version)

- Well known addresses

- Service configuration – name, type, number of partitions, task hung timeout, number of threads

- Cache configuration – tier, loader, batch factor, expiry delay, flush delay, high/low units, persistence type, queue delay, refresh factor, requeue threshold

- Journal resource manager configuration – buffer size, max file size, pool size, max journals files number, max backlog size

- OS configuration – OS name and version, CPU details, max file descriptor count, memory details

Apart from the out-of-box configuration collection, Oracle Enterprise Manager allows organizations to keep track of any configuration XML. Oracle Enterprise Manager provides out-of-box XML parsers. This enables administrators to track each configuration parameter in the XML separately.

Following are some of the configuration parameters collected for the Host

- Operating system patches, kernel parameter settings, and installed packages

- Hardware components including CPU, memory, storage, and network devices

Detecting Configuration Changes

Oracle Enterprise Manager also provides the ability to track configuration changes over time across the stack—from the application down to the hardware—allowing administrators to easily monitor all changes for a specific configuration between two points in time. This enables users to quickly see the changes applied to an environment that previously worked fine, but is suddenly not performing at an acceptable level.

Figure 11 – View Oracle Coherence configuration parameters collected out-of-box and custom XMLs

Lifecycle Management and Provisioning Automation

Oracle Enterprise Manage provides a complete provisioning solution for Oracle Coherence. Administrators can maintain their Oracle Coherence setup image in the software library and deploy it throughout their infrastructure to create completely new clusters or add nodes to an existing cluster.

If Oracle Coherence is already running on a machine, a new node can be easily started using the CoherenceHome and the start script of an existing node on the machine. Using these lifecycle management mechanisms for Oracle Coherence, Oracle Enterprise Manager automates mundane error-prone tasks which can result in downtime and loss of efficiency in the Oracle Coherence cluster.

Monitoring Oracle Coherence Clusters on Exalogic Elastic Cloud

Oracle Coherence is an integral part of the Exalogic Elastic Cloud platform. Oracle Coherence along with Oracle WebLogic server provides the critical middleware infrastructure to deliver high performance reliable applications on Exalogic Elastic Cloud. Oracle Enterprise Manager provides visibility into the Exalogic system group for all constituent Oracle WebLogic Severs and Oracle

Coherence clusters. The out-of-box dashboard provides a high level of visibility into the stack of strategic middleware infrastructure as well as the hardware itself thus correlating all components.

Oracle Enterprise Manager provides business-driven application management and monitoring for applications deployed on Oracle WebLogic Server and Oracle Coherence and deep visibility into the server runtime. In addition, the Middleware Diagnostics Advisor findings provide proactive performance findings pointing to the actual root-cause of the performance bottleneck across the stack. Administrators can easily drill down into the Oracle Coherence clusters from the dashboard to exploit all of the above mentioned features and get the best value out of their investment in the Oracle Fusion Middleware.

Conclusion

Today's IT organizations are increasingly adopting Java EE, Oracle Coherence, SOA, composite application, and cloud computing that enable them to quickly connect disparate applications and fulfill ever-changing business needs. Although these applications offer unprecedented flexibility and agility, they now are more challenging to manage. To effectively manage this new breed of applications, IT organizations need a new breed of management solutions. Oracle Enterprise Manager provides a new approach that enables administrators to stay focused on business priorities, using the most comprehensive management solution for the entire system stack in order to reduce the effort and cost of managing sophisticated applications built on Oracle Fusion Middleware.

ORACLE®

Enterprise Manager 12c Cloud Control:
Monitoring and Managing Oracle Coherence for
High Performance
September, 2012
Author: Madhav Sathe

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com

Oracle is committed to developing practices and products that help protect the environment

Hardware and Software, Engineered to Work Together