ORACLE ENTERPRISE MANAGER 12c

An Oracle White Paper
September, 2011

Oracle Database 11*g* Release 2: Database Manageability Overview

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. 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 Database is the market-leader and preferred database for hundreds of thousands of businesses as well as for application developers and database administrators worldwide. Over the years, enterprises have come to rely on the Oracle database to provide unparalleled performance and reliability. In Oracle Database 10g, Oracle delivered a self-managing database with breakthrough manageability that dramatically increased IT productivity and reduced management costs. Oracle is ready to raise the bar once again with the release of Oracle Enterprise Manager Cloud Control 12^c to manage Oracle Database. Designed for data center environments that are rapidly evolving and changing to keep up with the demands of the business, Oracle Database 11g and Oracle Enterprise Manager Cloud Control 12^c allow organizations to adopt new technologies quickly while minimizing risk. In addition, building on its industry-leading self-managing capabilities, Oracle Database 11g has made significant advances in the areas of manageability, testing and test data management and fault diagnostics that address many of the top challenges facing businesses today.

Manageability Challenges

The areas that continue to pose the biggest management challenges to database administrators include:

- Performance diagnostics and tuning: How to maintain production databases at their peak performance to maintain committed service levels.
- Testing and test data management: How to reduce the risk of rolling out changes through testing and managing test data in Oracle database environments at lower costs.
- Database lifecycle management and ongoing administration: How to automate the day-to-day repetitive tasks so that labor can be freed up to focus on more strategic requirements, such as security, data center consolidation and high availability.
- Cloud consolidation and Exadata management: How to consolidate databases onto a common infrastructure to reduce data center costs and increase server efficiency.

To address these challenges, Oracle Database 11*g* has made significant advances in performance, change assurance and self-management to make Oracle Database 11*g* easier to manage than ever before.

Performance Management

Performance management has traditionally been a major challenge for database administrators. With the self-managing database, Oracle Database 11*g*, managing database performance is easier than ever. Oracle Database 11*g* continues to expand its self-managing capabilities in all areas, including the two main areas of database performance management: performance diagnostics and application tuning.

Performance Diagnostics

The steps to achieve good performance are to gather the right data, make a proper analysis and to then derive an effective action plan.

The Oracle database self management framework performs these tasks for the DBA, making performance diagnostics simple and routine. The Automatic Workload Repository gathers the required data and the Automatic Database Diagnostics Monitor analyzes the data and makes targeted, concrete and actionable recommendations. Let us look at them in more detail.

Automatic Workload Repository

The Automatic Workload Repository (AWR) is a built-in repository within every Oracle database that contains operational statistics about that particular database and other such information. At regular intervals, the Oracle database makes a snapshot of all its vital statistics and workload information and stores them in AWR. By default, the snapshots are made every 60 minutes and are stored in the AWR for an 8 day period after which they are automatically purged. The administrator can easily change

these defaults. AWR is designed to be lightweight and manage itself completely in order to ensure that it does not impose any additional management burden on administrators.

The data captured allows both system level and user level analysis to be performed, again reducing the requirement to repeat the workload in order to diagnose problems. Optimizations have been performed to ensure that the capture of data is performed efficiently to minimize overhead.

AWR forms the foundation for all self-management functionality of Oracle Database. It is the source of information that gives the Oracle database an historical perspective on how it is being used and enables it to make decisions, which are accurate and specifically tailored for the environment that system is operating in.

Automatic Database Diagnostics Monitor (ADDM)

Building upon the data captured in AWR, Oracle Database includes a self-diagnostic engine called the Automatic Database Diagnostic Monitor (ADDM). ADDM makes it possible for the Oracle database to diagnose its own performance and determine how any identified problems could be resolved. ADDM runs automatically after each AWR statistics capture and makes the performance diagnostic data available immediately.

ADDM examines data captured in AWR and performs analysis to determine the major issues on the system on a proactive basis, recommends solutions and quantifies expected benefits. ADDM takes a holistic approach to the performance of the system, using time as a common currency between components. The goal of ADDM is to identify those areas of the system that are consuming the most 'DB time'. ADDM drills down to identify the root cause of problems rather than just the symptoms and reports the impact that the problem is having on the system overall. If a recommendation is made it reports the benefits that can be expected, again in terms of time.

The use of a metric such a db time throughout allows the impact of several problems or recommendations to be compared. Previously many problems have been identified based on value judgments and experience rather than quantifiable impacts. A good example of this is a system that is experiencing a high logon rate. A rule of thumb might have said that a logon rate of greater than 10 per seconds was a problem and should be fixed. However many systems can run significantly higher logon rates without it noticeably affecting performance. Using the time distribution data in AWR, ADDM can report quantitatively that logons are taking 20% of time spent in the database. This quantified value can make it much easier to convince whoever needs to do the work to fix the problem or arrange for it to be fixed, rather than just making a statement such as 'I think that you are doing too many logons'.

ADDM starts its analysis by focusing on the activities that the database is spending most time on and then drills down through a sophisticated problem classification tree. The problem classification tree used by ADDM encapsulates decades of performance tuning experience in the Server Technologies Performance Group at Oracle Corporation and other performance experts.

In developing the classification tree, the prime intent was to handle the most frequently seen problems and to drill down to the root causes of problems rather than just reporting symptoms. Some of the common problems detected by ADDM include:

- CPU bottlenecks
- Poor connection management
- Excessive parsing
- Lock contention
- I/O capacity
- Under sizing of Oracle memory structures e.g. PGA, buffer cache, log buffer
- High load SQL statements
- High PL/SQL and Java time
- High checkpoint load and cause e.g., small log files
- RAC-specific issues

Besides reporting the potential performance issues, ADDM also documents the non-problem areas of the system. The sub-components, such as I/O, memory, etc, that are not significantly impacting the performance of the system are pruned from the classification tree at an early stage and are listed so that the DBA can quickly see that there is little to be gained by performing actions in those areas. Again this saves time and wasted effort (both human and hardware) fixing things that will not impact the system performance overall.

Oracle Database 11*g* extends ADDM by greatly enhancing cluster-wide performance analysis for Real Application Clusters (RAC) databases. For RAC environments ADDM analyses the RAC cluster and reports on issues that are affecting the entire database as well as its individual instances. DBAs can now use ADDM to perform database-wide analysis of global resources, such as high-load SQL, global cache interconnect traffic, network latency issues, skew in instance response times, and I/O capacity.

Oracle Database was the first database product to introduce such a revolutionary self-diagnostic capability and has completely redefined the database administration landscape. Administrators no longer need to first collect huge volume of diagnostic data and spend endless hours analyzing them in order to find out answers to performance issues. With the Oracle Database 11*g*, they can simply ask the database what the performance issues are and ADDM does the rest. They can sit back, relax, and follow the recommendation made by ADDM using just a few mouse clicks.

Real-Time ADDM for Diagnosing Unresponsive Databases

Oracle Enterprise Manager Cloud Control 12*c* introduces Real-Time ADDM, an innovative way to analyze problems in extremely slow or hung databases, which would have traditionally required a database restart. Real-Time ADDM can help resolve issues like deadlocks, hangs, shared pool contentions and many other exception situations without resorting to a restart of the database.

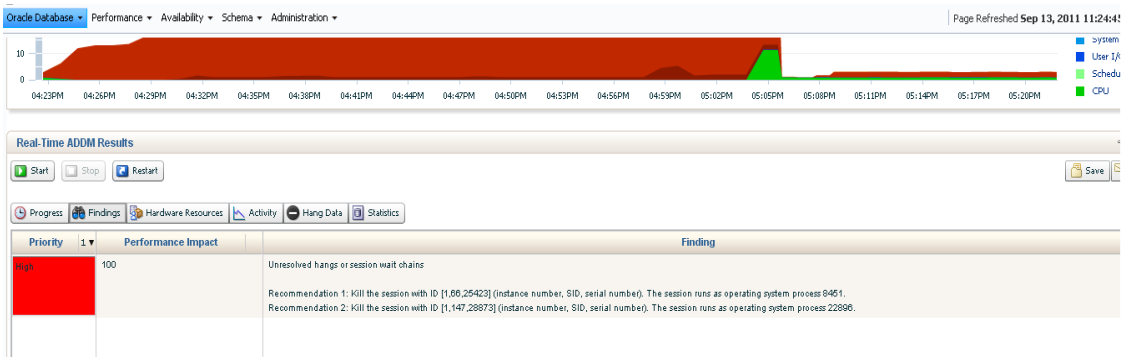

Figure 1: SQL Performance Analyzer report

Real-Time ADDM uses two different types of connection modes to connect to the target instance. The normal JDBC connection is intended to perform extensive performance analysis when some connectivity is available. The diagnostic mode which makes a latch-less connection is useful for extreme hang situations when normal connection is not possible. The database administrator can then use the diagnostic mode connection to execute analysis using Real-Time ADDM and get recommendations on how to resolve the hang situation.

Compare Period ADDM for Comparative Performance Analysis

Compare Period ADDM is a new feature in Oracle Enterprise Manager Cloud Control 12c that allows the administrator to answer the age-old question of why performance is slower today compared to yesterday. The administrator can choose from either an AWR baseline or the older AWR snapshot period or any calendar period of choice to determine why database performance during a particular period is slower than another period. Compare Period ADDM checks both the base period and the comparison period and lists out a set of findings pinpointing the root cause for the difference in performance. Compare Period ADDM also indicates whether the two periods are comparable, i.e. have similar SQL running in the same period, by the use of the SQL Commonality index for the two periods.

AWR Baselines and Adaptive Thresholds

The usefulness and value of AWR continues to expand with every new release of Oracle Database. AWR baselines allow DBAs to capture and save system performance data over time periods with interesting or representative workloads.

In addition, baselines can also be used in setting alert thresholds on system performance metrics. Most metrics can be viewed in Oracle Enterprise Manager against statistical aggregates of those same metrics observed over the baseline period. This helps users set baseline-informed thresholds rather than selecting thresholds without the context of actual data. In addition, adaptive thresholds are available for certain key performance metrics. Adaptive thresholds are performance alert thresholds that are automatically set and periodically adjusted by the system using the System Moving Window Baseline data as the basis for threshold determination. For customers who want to get started with adaptive

thresholds immediately, the new "Quick Configure" option can setup a starter kit of thresholds based on common workload profiles using a few mouse clicks.

AWR baselines provide powerful capabilities for defining dynamic and future baselines and considerably simplify the process of creating and managing performance data for comparison purposes.

Active Session History

Another key component of AWR is the Active Session History (ASH).

All active database sessions are automatically sampled once every second and stored in the Active Session History. The ASH data shows where the database is currently spending its time and highlights any performance bottlenecks.

Oracle Database 11*g* Release 2 extends ASH by gathering additional RAC information. The information is used by advisors, such as ADDM for RAC. This information can be seen in new RAC-specific sections to the ASH report. The ASH report now lists events that account for the highest percentage of session activity in the cluster wait class along with the instance numbers of the affected instance. This information gives further visibility into potential RAC-specific issues.

ASH has also been extended to run on even on standby databases to assist in analysis of the performance of queries executed on Active Dataguard instances.

ASH Analytics

Oracle Enterprise Manager Cloud Control 12*c* includes ASH Analytics, a new tool to explore ASH data that allows the administrator to rollup, drilldown, and slice or dice performance data across various performance dimensions. Using ASH Analytics the database administrator can explore the different performance attributes of a database session at any point in time.

With the ability to create filters on various dimensions, the DBA can not only identify performance issues but also get a good understanding of various performance patterns and resource usage of the system. The ASH Analytics view is also available as an active report that can be used for offline analysis of any performance issues at a later point in time. The built-in treemap view allows administrators to explore performance data using predefined performance dimension hierarchies.

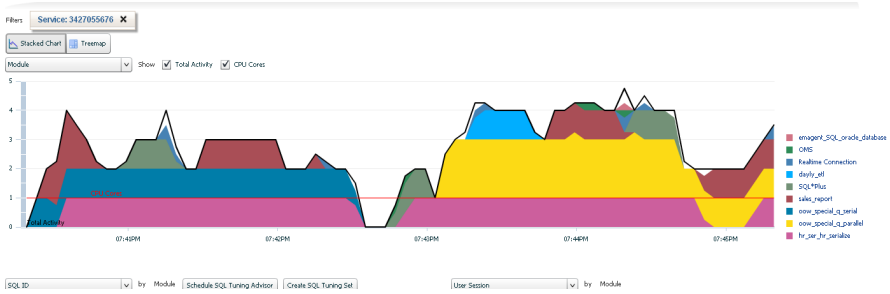

Figure 2: ASH Analytics

Application Tuning

Application design issues are the most predominant cause of performance problems. All the tuning wisdom of developers, DBAs and system administrators cannot make up for the slowdown caused by architectural and design deficiencies of the application. One important part of database system performance tuning is, therefore, the tuning of SQL statements.

The query optimizer takes crucial decisions which have a tremendous impact on the performance of a query, such as whether to use an index or not, which join techniques to use if the query involves joining multiple tables, etc. Oracle Corporation has invested considerable development effort in making the cost-based optimizer the industry's most sophisticated, mature, and thoroughly-tested query optimizer. The cost-based optimizer is also used extensively by the major packaged applications such as Oracle eBusiness Suite, SAP, PeopleSoft, etc. Since the Oracle database is the platform for the vast majority of customers using these applications, this demonstrates the success of the Oracle optimizer in a huge number of real application settings. Consequently, starting with Oracle Database 10*g*, the rule based optimizer (RBO) is no longer available and the cost-based optimizer is the only supported optimizer mode.

While Oracle Database provides best possible query optimization technology, which maximizes the application/query performance without any administrator intervention in the majority of cases, there may still be a few cases where the nature of the application or uniqueness of data distribution may cause certain SQL statements to consume an unusually high percentage of total system resources. In such situations, the SQL tuning process normally involves three basic steps:

- Identifying high load or top SQL statements that are responsible for a large share of the application workload and system resources, by looking at the past SQL execution history available in the system (e.g., the cursor cache statistics stored in the V$SQL dynamic view)
- Verifying that the execution plans produced by the query optimizer for these statements perform reasonably well
- Taking possible corrective actions to generate better execution plans for poorly performing SQL statements.

The three steps are repeated until the system performance reaches a satisfactory level or no more statements can be tuned. Besides being extremely time consuming, the SQL tuning process outlined

above also requires a high degree of expertise. Only a person with a deep knowledge of the application and database system can undertake this task.

SQL Tuning and Access Advisors

Starting with Oracle Database 10*g* the SQL tuning process has been completely automated. ADDM identifies SQL statements that are consuming unusually high system resources and are therefore causing performance problems. In addition, the top SQL statements in terms of CPU and shared memory consumption are automatically captured in AWR. Thus, the identification of high load SQL statements happens automatically in the new tuning framework and requires no intervention from the administrator.

After the top resource consuming SQL statements have been identified, the Oracle database can automatically analyze them and recommend solutions using newly added automatic tuning capability of the query optimizer, called the automatic tuning optimizer. The automatic tuning optimizer is exposed via an advisor called the SQL Tuning Advisor. The SQL Tuning Advisor takes one or more SQL statements, and produces well-tuned plans along with tuning advice. The administrator does not need to do anything other than just invoking the SQL Tuning Advisor so that it can recommend a solution. It is important to bear in mind here that the solution is coming right from the optimizer and not from any external tools using some pre-defined heuristics. This provides several advantages:

- the tuning is done by the system component that is ultimately responsible for the execution plans
- the tuning process is fully cost-based and naturally accounts for any changes and enhancements done to the query optimizer
- the tuning process takes into account the past execution statistics of a SQL statement and customizes the optimizer settings for that statement
- it collects auxiliary information in conjunction with the regular statistics based on what is considered useful by the query optimizer.

The recommendation of the automatic tuning optimizer can fall into one of the following categories:

Statistics Analysis: The automatic tuning optimizer checks each query object for missing or stale statistics, and makes recommendation to gather relevant statistics. It also collects auxiliary information to supply missing statistics or correct stale statistics in case recommendations are not implemented.

SQL Profiling: The automatic tuning optimizer verifies its own estimates and collects auxiliary information to remove estimation errors. It also collects auxiliary information in the form of customized optimizer settings (e.g., first rows vs. all rows) based on past execution history of the SQL statement. It builds a SQL profile using this auxiliary information and makes a recommendation to create it. When a SQL profile is created it enables the query optimizer (under normal mode) to generate a well-tuned plan. The most powerful aspect of SQL profiles is that they enable tuning of queries without requiring any syntactical changes and thereby proving Oracle administrators with a solution to tune the SQL statements embedded in packaged applications.

Access Path Analysis: The automatic tuning optimizer explores whether a new index can be used to significantly improve access to each table in the query, and when appropriate makes recommendations to create such indexes.

SQL Structure Analysis: The automatic tuning optimizer tries to identify SQL statements that lend themselves to bad plans, and makes relevant suggestions to restructure them. The suggested restructurings can be syntactic as well as semantic changes to the SQL code.

Both access path and SQL structure analysis can be immensely useful in tuning the performance of an application under development or a homegrown production application where the administrators/developers have access to application code.

The SQL Access Advisor is yet another major component of Oracle Database manageability. It can automatically analyze the schema design for a given workload and recommend indexes, materialized views, and materialized view logs to create, retain or drop as appropriate for the workload. While generating recommendations, the SQL Access Advisor considers the impact of adding new access structures on data manipulation activities, such as insert, update and delete, in addition to the performance improvement they are likely to provide for queries. The SQL Access Advisor provides a very easy to use interface and requires very little system knowledge. It can also be run without affecting production systems since the data can be gathered from the production system and taken to another machine where the SQL Access Advisor can be run.

The SQL Access Advisor has been enhanced in Oracle Database 11*g* to provide partition advice as a part of SQL access structure recommendations.

Automatic SQL Tuning

In Oracle Database 11*g*, the SQL tuning process was further enhanced and automated to keep databases running at their peak performance. The SQL Tuning Advisor now runs automatically during the system maintenance windows as a maintenance task. In each run, it automatically selects high-load SQL queries in the system, and generates recommendations on how to tune them.

To validate the recommendation, SQL Tuning Advisor in Oracle Database performs a test-execute of the SQL statements with the new execution plan for which a SQL profile is recommended. This dramatically increases the accuracy and reliability of SQL profile recommendations.

The SQL Tuning Advisor can be configured to automatically implement SQL profile recommendations. If you enable automatic implementation, the advisor will create SQL profiles for only those SQL statements where the performance improvement would be at least three-fold. Other types of recommendations, such as the ones to create new indexes or refresh optimizer statistics or the ones that restructure SQL, can only be implemented manually. DML statements are not considered for tuning by the automatic SQL Tuning Advisor. By default, the automatic SQL Tuning Advisor is configured to run nightly and only report recommendations but not automatically implement them.

You can view a summary of the results of automatic SQL tuning over a specified period (such as the previous seven days), as well as view a detailed report on recommendations made for all SQL statements processed. The recommendations can then be selectively implemented through a manual

process. You can also view the recommendations that were automatically implemented. The automatic SQL Tuning Advisor can be configured to run in any maintenance window or can be disabled altogether if desired.

Real Time SQL Monitoring

The Real Time SQL Monitoring feature of Oracle Database 11*g* enables monitoring of the performance of SQL statements while they are executing. Live execution plans of long running SQL are automatically displayed on the SQL Monitor page in Oracle Enterprise Manager using new, fine-grained SQL statistics that are tracked out-of-the-box.

By default, SQL monitoring is automatically started when a SQL statement runs in parallel, or when it has consumed at least 5 seconds of CPU or I/O time in a single execution. The DBA can observe the SQL statement step through the execution plan, displaying statistics for each step as it executes. Row source information at each step of the execution plan is tracked by means of key performance metrics, including elapsed time, CPU time, number of reads and writes, I/O waits and other wait time. SQL monitoring gives the DBA information on what steps long running SQL are executing and allows the DBA to decide if additional tuning action needs to be taken.

Real Time SQL Monitoring has been enhanced in Oracle Database 11*g* Release 2 to support execution plans that are being executed in part by the Oracle Database machine, Exadata.

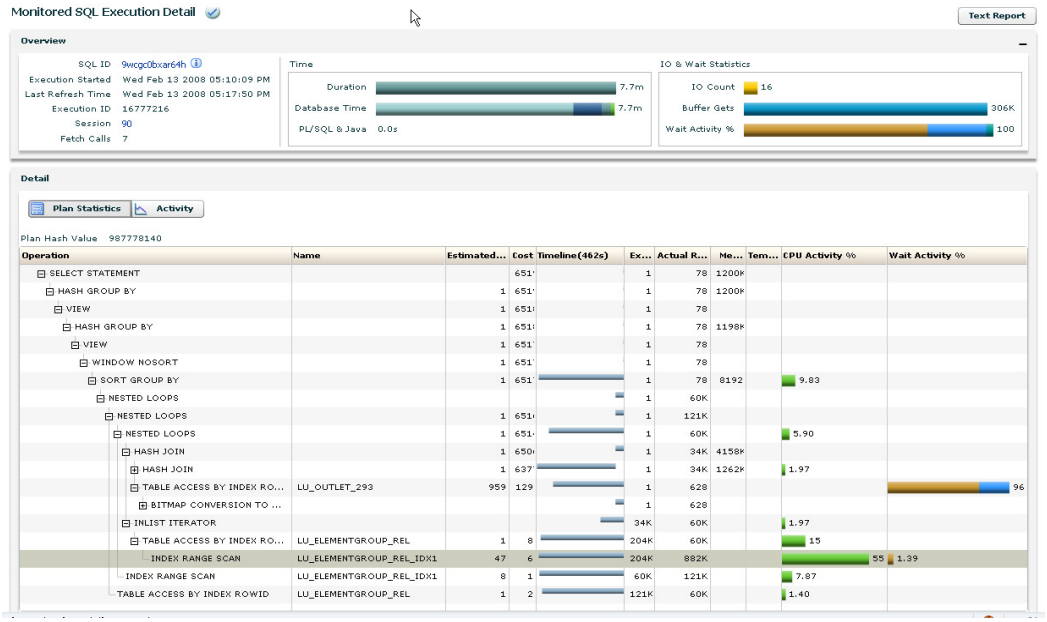

Figure 3. Real Time SQL Monitoring execution plan

In addition to being able to monitor SQL statements in real time in Oracle Database 11*g* Release 2, the DBA can also save all the execution details in an active report - an interactive report that can be used

for offline analysis. It offers the same level of interactivity as the live screens, with drill-downs to various levels of detail.

SQL Plan Management

SQL plan management prevents performance regressions resulting from sudden changes to the execution plan of a SQL statement by providing components for capturing, selecting, and evolving SQL executions plans. SQL performance can be affected by various changes, such as a new optimizer version, changes to optimizer statistics and/or parameters, or creation of SQL profiles. SQL plan management is a preventative mechanism that records and evaluates the execution plans of SQL statements over time, and builds SQL plan baselines composed of a set of existing plans known to be efficient. The SQL plan baselines are then used to preserve performance of the corresponding SQL statements, regardless of changes occurring in the system.

Common usage scenarios where SQL plan management can improve or preserve SQL performance include:

- A database upgrade that installs a new optimizer version may result in plan changes for a small percentage of SQL statements, with most of the plan changes resulting in either improvement or no performance changes. However, certain plan changes may cause performance regressions. The use of SQL plan baselines significantly minimizes potential performance regressions resulting from a database upgrade.
- Ongoing system and data changes can impact plans for some SQL statements, potentially causing performance regressions. The use of SQL plan baselines can also help to minimize performance regressions and stabilize SQL performance.
- Deployment of new application modules means introducing new SQL statements into the system. The application software may use appropriate SQL execution plans developed under a standard test configuration for the new SQL statements.

SQL plan baselines evolve over time to produce better performance. During the SQL plan baseline evolution phase, Oracle Database 11*g* routinely evaluates the performance of new plans and integrates plans with better performance into SQL plan baselines. A successful verification of a new plan consists of comparing its performance to that of a plan selected from the SQL plan baseline and ensuring that it delivers better performance.

Migration of Stored Outlines to SQL Plan Baselines

Before SQL plan baselines were introduced as part of SQL plan management, stored outlines served a similar function. However, stored outlines lack the flexibility and adaptability of SQL plan management:

- Stored outlines cannot automatically evolve over time. Consequently, a stored outline may be good when it is created, but become a bad plan after a database change, leading to performance degradation.

- Hints in a stored outline can become invalid - for example, an index hint on a dropped index. In such cases, the database still uses the outlines but excludes the invalid hints, producing a plan that is often worse than the original plan or the current best-cost plan generated by the optimizer.
- For a SQL statement, the optimizer can only choose the plan defined in the stored outline in the currently specified category. The optimizer cannot choose from other stored outlines in different categories or the current cost-based plan even if they improve performance.

Oracle Database 11*g* Release 2 provides the capability of migrating stored outlines to SQL plan baselines. The benefits of migrating to SQL plan baselines include:

- SQL plan baselines enable the optimizer to use the same good plan and allow this plan to evolve over time. For a specified SQL statement, you can add new plans as SQL plan baselines after they are verified not to cause performance regressions.
- SQL plan baselines prevent plans from going bad because of invalid hints. If hints stored in a plan baseline become invalid, the plan may not be reproducible by the optimizer. In this case, the optimizer selects an alternative reproducible plan baseline or the current best-cost plan generated by the optimizer.
- For a specific SQL statement, the database can maintain multiple plan baselines. The optimizer can choose from a set of good plans for a specific SQL statement instead of being restricted to a single plan per category, as required by stored outlines.

By utilizing the migration path in Oracle Database 11*g* Release 2, old applications using stored outlines can be transparently migrated and can instantaneously take advantage of the enhanced functionality of SQL plan management.

Testing and Test Data Management

Oracle Enterprise Manager's Application Quality Management (AQM) solutions provide high quality testing for all tiers of the application stack. Thorough testing can help users identify application quality and performance issues prior to deployment. Testing is one of the most challenging and time consuming parts of successfully deploying an application, but it is also one of the most critical to the project's success. The testing and secure test data management capabilities in Oracle Enterprise Manager provide a unique combination of test capabilities for Oracle databases which enable users to:

- Test infrastructure changes: Real Application Testing is designed and optimized for testing database tier infrastructure changes using real application production workloads to validate database performance in your test environment.
- Manage your test data and enable secure production-class testing: Oracle Data Masking and Oracle Test Data Management solutions helps enterprises achieve security & compliance objectives by obfuscating sensitive data from production in test databases and scale down production data into right-sized databases so production data can be used securely in test and development environments.

Throughput Testing using Database Replay

Database Replay provides DBAs and system administrators with the ability to faithfully, accurately and realistically rerun actual production workloads, including online user and batch workloads, in test environments. By capturing the full database workload from production systems, including all concurrency, dependencies and timing, Database Replay enables you to realistically test system changes by essentially recreating production workloads on the test system – something that a set of scripts can never duplicate. With Database Replay, DBAs and system administrators can test

- Database upgrades, patches, parameter, schema changes, etc.
- Configuration changes such as conversion from a single instance to RAC, ASM, etc.
- Storage pool, network, and interconnect changes
- Operating system and hardware migrations, patches, upgrades, and parameter changes

Lower Test Infrastructure Cost

DBAs now have a test infrastructure at their disposal to test their changes without the overhead of having to duplicate an entire application infrastructure. Database Replay does not require the set up overhead of having to recreate a middle-tier or a web server tier. Thus, DBAs and system administrators can rapidly test and upgrade data center infrastructure components with the utmost confidence, knowing that the changes have truly been tested and validated using production scenarios.

Faster Deployment

Another major advantage of Database Replay is that it does not require the DBA to spend months getting a functional knowledge of the application and developing test scripts. With a few point and clicks, DBAs have a full production workload available at their fingertips to test and rollout any change. This cuts down testing cycles from many months to days or weeks and brings significant cost savings to businesses as a result.

Database Replay consists of four main steps:

1. Workload capture - When workload capture is enabled, all external client requests directed to the Oracle database are tracked and stored in binary files, called capture files, on the file system. The user specifies the location of the capture files and the workload capture start and end time. During this process, all information pertaining to external database calls is written to the capture files.
2. Workload processing - Once the workload has been captured, the information in the capture files has to be processed. This processing transforms the captured data into replay files and creates all necessary metadata needed for replaying the workload. The capture files would typically be copied to another system for processing. This must be done once for every captured workload before they can be replayed. After the captured workload is processed, it can be replayed repeatedly on a replay system. As workload processing can be time consuming

and resource intensive, it is generally recommended that this step be performed on the test system where the workload will be replayed.

3. Workload replay - After the captured workload has been processed, it is now ready for replay. A client program, called Replay Client, then processes the replay files and submits calls to the database with the exact same timing and concurrency as in the capture system. Depending on the captured workload, you may need one or more replay clients to properly replay the workload. A calibration tool is provided to help determine the number of replay clients needed for a workload. It should be noted that since the entire workload is replayed including DML and SQL queries, it is important that the data in the replay system be identical to that in the production system, whose workload was captured, to enable reliable analysis for reporting purposes.
4. Analysis and Reporting - Extensive reports are provided to enable detailed analysis of the capture and replay. Any errors encountered during replay are reported. Any divergence in rows returned by DML or queries is shown. Basic performance comparisons between capture and replay are provided. For advanced analysis, Replay Compare Period and other AWR reports are available to allow detailed comparison of various statistics between capture and replays.

Both the workload capture and replay process support a filtering capability that is useful for targeting workload of interest, such as by service, action, module to name a few. Oracle Enterprise Manager significantly enhances the value of Real Application Testing by supporting end-to-end Database Replay automation. This simplifies the process of saving and transferring the workload capture and performance data to the test system, setting up the test system and replay clients correctly, and orchestrating the entire replay through the Oracle Enterprise Manager interface.

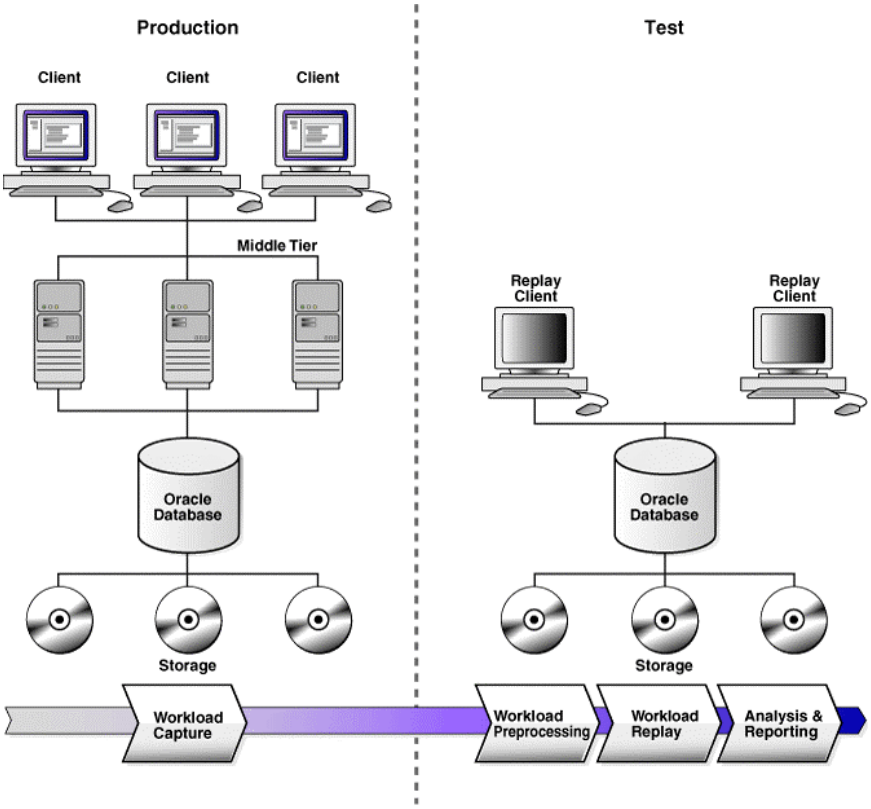

Figure 4: Database Replay Workflow

Response Time Testing using SQL Performance Analyzer

Changes that affect SQL execution plans can severely impact application performance and availability. As a result, DBAs spend enormous amounts of time identifying and fixing SQL statements that have regressed due to the system changes. SQL Performance Analyzer (SPA) can predict and prevent SQL execution performance problems caused by environment changes.

SPA provides a granular view of the impact of environment changes on SQL execution plans and statistics by running the SQL statements serially before and after the changes. SPA generates a report outlining the net benefit on the workload due to the system change as well as the set of regressed SQL statements. For regressed SQL statements, appropriate executions plan details along with recommendations to tune them are provided.

SPA is well integrated with existing SQL Tuning Set (STS), SQL Tuning Advisor and SQL plan management functionality. SPA completely automates and simplifies the manual and time-consuming process of assessing the impact of change on extremely large SQL workloads (thousands of SQL statements). DBAs can use SQL Tuning Advisor to fix the regressed SQL statements in test environments and generate new plans. These plans are then seeded in SQL plan management baselines and exported back into production. Thus, using SPA, businesses can validate with a high degree of

confidence that a system change to a production environment in fact results in net positive improvement at a significantly lower cost.

Examples of common system changes for which you can use SPA include:

- Database upgrade, patches, initialization parameter changes
- Configuration changes to the operating system, hardware, or database
- Schema changes such as adding new indexes, partitioning or materialized views
- Gathering optimizer statistics·
- SQL tuning actions, for example, creating SQL profiles

Using SPA involves the following 5 main steps:

1. Capture the SQL workload that you want to analyze with SPA. The Oracle database offers ways to capture SQL workload from several sources, such as cursor cache and Automatic Workload Repository, into a SQL tuning set (STS). This would typically be done on a production system and the STS would then be transported to the test system where SPA analysis will take place.
2. Measure the performance of the workload before a change by executing SPA on the SQL tuning set. Very short running queries are executed multiple times and their statistics are averaged to eliminate variations due to buffer cache state and other noise factors
3. Make the change, such as database upgrade or optimizer statistics refresh.
4. Measure performance of the workload after the change by executing SPA on the SQL tuning set again, as in step 2.
5. Compare performance of the two executions of the SQL tuning set to identify the SQL statements that have regressed, improved, or were unchanged.

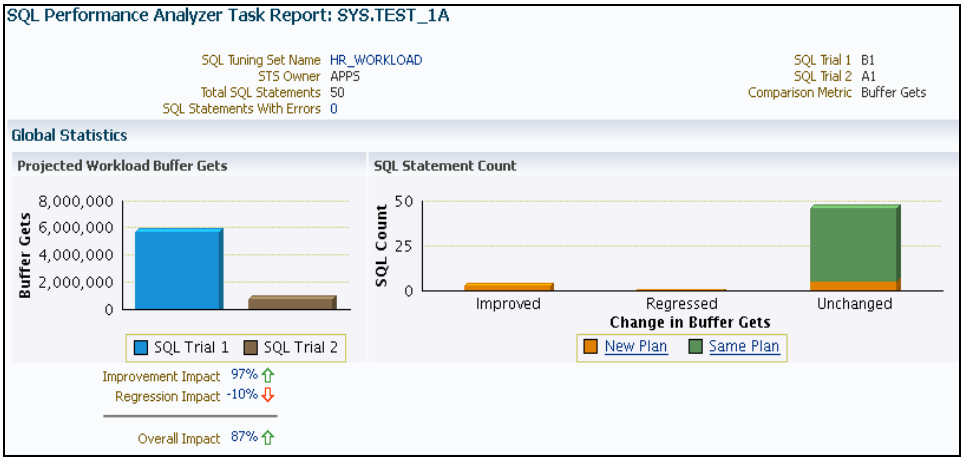

Figure 5: SQL Performance Analyzer Report

This SPA comparison report shows significant performance improvement of overall SQL workload after the proposed system change but with a few execution plan regressions. SQL Performance Analyzer takes into account the number of executions of a SQL statement when measuring its impact. A SQL statement that completes in seconds but is frequently executed may have a higher impact on the system than a long running statement executed only once. SPA takes these factors into account when predicting overall performance improvements and regressions. If any regressions are encountered, SPA allows the user to fix them using SQL Tuning Advisor or with SQL plan baselines, a new plan stability feature introduced in Oracle Database 11*g*.

SPA supports numerous other features that help assess system changes, these are briefly described below:

1. SPA helps estimate the I/O reduction that can be accomplished by migrating to an Exadata server but without actually requiring you provision the hardware. This can be used to identify potential workloads/systems that are good candidates for Exadata migration.
2. SPA supports comparing two STSs – this functionality is useful when you have mechanisms such as load testing scripts or Oracle Application Testing Suite that can be used to test the changes. By capturing the workload in to two different STSs (for before and after change runs), one can use SPA to assess the impact of the system change.
3. With Oracle Enterprise Manager, a "one-click" STS transport mechanism can be used to simplify the process of moving STS workloads between production and test databases.

Choosing the right solution helps DBAs absorb and manage change efficiently. Database Replay is designed to test and improve system performance and SQL Performance Analyzer helps DBAs improve SQL response time. Oracle 11*g* Real Application Testing makes it easy for database administrators to manage and execute changes that are critical to the business and do it all at lower risk.

Protecting Sensitive Data in Test Environments

Administrators can leverage the provisioning capabilities in Oracle Enterprise Manager to roll out pre-tested standardized gold images of Oracle Database. This provides administrators with tremendous labor savings instead of having to execute each step of the provisioning process manually. These gold images can be used to provision test systems from backups or live production databases.

When enterprises copy production data into test environments for the purposes of application development or testing, they risk falling out of compliance with regulations or incurring fines and penalties that accompany violations of these data privacy laws. The data masking capabilities available to administrators helps organizations comply with privacy and confidentiality laws by masking sensitive or confidential data in development, test or staging environments. By using an irreversible process to replace sensitive data with realistic-looking but scrubbed data based on masking rules, security administrators can ensure that the original data cannot be retrieved, recovered or restored while maintaining the integrity of the application.

Reducing Storage Costs with Data Subsetting

With the growth in the number of database applications, enterprises are faced with the challenge of provisioning non-production environments which are used for application development and testing. They cannot afford to incur the storage expenses of provisioning the same production data in their non-production databases; nor do they have the tools or the application knowledge to shrink production data to a right-sized development environment. Oracle's test data management functionality helps enterprises shrink storage costs by creating reduced size copies of production data for application development and testing while maintaining the referential integrity of the data set. Through data discovery and application modeling, Oracle's test data management functionality automatically enforces complex business rules of enterprise applications resulting in accurate subsets of production data.

Real Application Testing and Data Masking integration enables businesses to perform secure testing. Typically testing is done in a non-production environment or by a different group or organization. Sharing production data and/or the captured workload that contains sensitive information results in breach of data privacy regulations and poses significant business risk. Real Application Testing and Data Masking integration enables sharing of captured workload and data in the database in compliance with data privacy regulations.

With Oracle Enterprise Manager Cloud Control 12*c*, the data masking functionality has been enhanced to consistently mask sensitive not just in tables but also across all Real Application Testing artifacts such as SQL tuning sets and Database Replay workload capture files. This allows you to perform proper testing even after test data has been masked. With Real Application Testing and Data Masking integration, businesses can now perform secure testing in a manner compliant with data privacy regulations.

Database Lifecycle Management and Ongoing Administration

Automating the day to day repetitive tasks that in the past have taken too much of an administrators time is a key achievement of the self managing database, Oracle Database 11*g*. By relieving the administrators of the tedious management tasks, such as provisioning or patching databases, managing memory allocations and disk resources, they can be freed up to focus on more strategic requirements, such as security and high availability.

Database Lifecycle Management

Database lifecycle management covers the entire lifecycle of databases, including:

- Discovery and inventory tracking: the ability to discover your assets, and track them
- Initial provisioning: the ability to rollout databases in minutes
- Ongoing change management: end-to-end management of patches, upgrades, schema and data changes
- Configuration management: track inventory, configuration drift and detailed configuration search

- Compliance management: reporting and management of industry and regulatory compliance standards

Database lifecycle management capabilities eliminate the need to manually track IT assets involving databases. It provides non-intrusive out-of-box agentless capabilities to discover physical servers. Once servers have been discovered, they are easily promoted to a managed state automatically discovering all databases and other applications. This automated discovery simplifies the process of ensuring all your servers and software are managed along with assisting in IT infrastructure consolidation and optimization initiatives.

It comes with out-of-box deployment procedures to provision and patch Oracle Database (both single instance database and RAC) including the underlying infrastructure. Enterprise Manager supports segregation of duties, so that a designer can create the provisioning and patching workflows while an operator can simply deploy the databases using those workflows. One can also provision a new database from a reference system or from a gold image. The gold image along with configuration details can be captured in provisioning profiles which can either be sourced from a reference system or downloaded from Oracle.

Database Lifecycle Management supports the entire patch management lifecycle, including patch advisories, pre-deployment analysis, rollout and reporting. It is integrated with My Oracle Support to provide a synchronized view of available and recommended patches. Database lifecycle management also provides complete automation for the schema change deployment process. Administrators can also define gold standards and baselines for configurations allowing them to standardize their environments against those definitions.

Resource Management

Automating resource management tasks, such as managing memory allocation and disk resources, has been another key achievement of the self managing database, Oracle Database 11*g*. Let's examine these tasks in more detail.

Automatic Memory Management

Memory is a precious system resource and administrators historically have spent a significant amount of their time optimizing its use. One of the key self-management enhancements in Oracle Database 11*g* is automatic memory management. This functionality automates the management of shared memory used by an Oracle instance and liberates administrators from having to configure the shared memory components manually. The automatic memory management feature is based on sophisticated heuristics internal to the database that monitors the memory distribution and changes it according to the demands of the workload.

Oracle memory structures basically consist of shared memory – the System Global Area (SGA) – and private memory – the Program Global Area (PGA). In Oracle Database 9*i*, the automatic SQL execution memory management feature was introduced to automate management of the PGA. In Oracle Database 10*g*, the same was done for the SGA by the introduction of automatic shared memory management. This meant all the different SQL areas in the PGA were automatically sized for the

system workload to give best performance and all the memory pools in shared memory were similarly adjusted for optimal performance. The user was only required to specify the PGA and SGA target sizes and Oracle would appropriately allocate memory within these targets to give the best possible performance. PGA and SGA Advisors were also provided to help the user properly set the targets for the SGA and PGA in Oracle Database 10*g*.

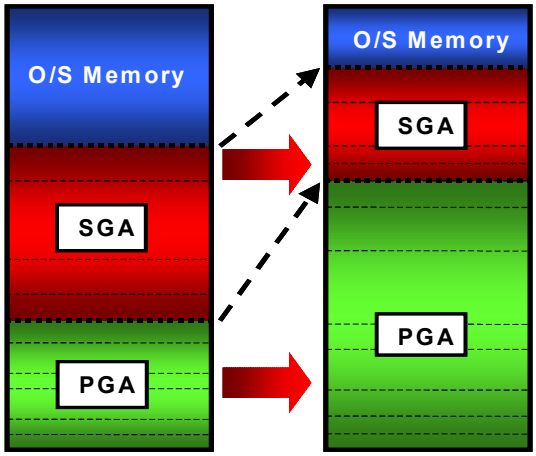

Figure 6. Automatic Memory Management

In Oracle Database 11*g*, memory management has been automated even further. All memory, PGA and SGA, is now managed centrally with the automatic memory management feature. DBAs need to specify a single parameter, MEMORY_TARGET, and Oracle will automatically size the PGA and SGA based on the workload. Using indirect memory transfer, the database transfers memory from SGA to PGA and vice versa to respond to the load. Dynamic allocation of memory is adjusted at frequent intervals to optimize memory usage in line with workload requirements to maximize memory utilization and avoid out-of-memory errors. Users can optionally set SGA and PGA targets when using the automatic memory management feature. This ensures that SGA and PGA sizes will not shrink below the values specified by their respective parameter targets in automatic tuning mode. This feature is currently available on Linux, Solaris, HP-UX, AIX and Windows platforms.

First introduced in Oracle Database 10*g*, memory advisors provide graphical analyses of total memory target settings, SGA and PGA target settings, or SGA component size settings. DBAs can use these analyses to tune database performance and to perform what-if planning scenarios. Different memory advisors become available depending on the memory management mode used with the database.

For instance, if automatic memory management is enabled, you can get advice for setting the target amount of memory to allocate to the entire instance. If automatic shared memory management is enabled, you can gain advice on configuring the target sizes of the SGA and PGA. If manual shared memory management is enabled, you can get advice on sizing the shared pool, buffer cache, and PGA.

Space Management

Space management can be one of the most time consuming tasks for database administrators. Fortunately, Oracle Database 11*g* automatically manages its space consumption, alerts administrators on potential space problems, and recommends possible solutions.

Proactive Space Management

Oracle Database 11*g* does non-intrusive and timely monitoring checks for space utilization in the database server. Oracle Database 11*g* automatically monitors its space utilization during its normal space allocation and de-allocation operations and alerts administrators if the free space availability falls below the pre-defined thresholds. Oracle Database 11*g*'s space monitoring functionality is set up out-of-box, causes no measurable performance impact, and is uniformly available across all tablespace types. Since the monitoring is performed at the same time as space is allocated and freed up in the database server, this guarantees immediate availability of space usage information whenever the user needs it.

Notification is performed using server generated alerts mechanism. The alerts are triggered when certain space related events occur in the database. For example when the space usage threshold of a tablespace is crossed, an alert is raised. Another example of an alert is when a resumable session encounters an out of space situation. An alert is sent instantaneously to the DBA to take corrective measures. The DBA may choose to get paged with the alert information and add space to the tablespace to allow the suspended operation to continue from where it left off.

The database comes with a default set of alert thresholds. The DBA may override the default for a given tablespace or set a new default for the entire database through Oracle Enterprise Manager.

Transparent Space Reclamation

Oracle Database 11*g* provides the ability of performing an in-place reorganization of data for optimal space utilization by shrinking segments. Shrinking of a segment will make unused space available to other segments in the tablespace and may improve the performance of queries and DML operations.

The segment shrink functionality provides the ability to both compact the space used in a segment and then de-allocate it from the segment. The de-allocated space is returned to the tablespace and is available to other objects in the tablespace. Sparsely populated tables may cause a performance problem for full table scans. By performing a shrink, data in the table is compacted and the high water mark of the segment is pushed down. This makes full table scans read less blocks, and hence run faster.

Segment shrink is an online operation – the table being shrunk is open to queries and DML while the segment is being shrunk. Additionally, segment shrink is performed in-place. This is a key advantage over performing online table redefinition for compaction and reclaiming space. The DBA may schedule segment shrink for one or all the objects in the database as nightly jobs without requiring any additional space to be provided to the database.

In order to easily identify candidate segments for shrinking, Oracle Database 11*g* also includes an automatic segment advisor. The automatic segment advisor runs every night in a predetermined

maintenance window to proactively identify segments that need shrinking. The advisor, whether invoked manually or automatically, performs growth trend analysis on individual objects to determine if there will be any additional space left in the object in 7 days. It then uses the reclaim space target to select candidate objects to shrink.

Segment Creation on Demand

Installation of a packaged application can often create thousands of database tables and indexes. The creation of these tables and indexes can be time consuming and use a significant amount of disk space. Many of these tables and indexes may never be used if you have not licensed all the modules of the packaged application. In Oracle Database 11*g* Release 2, when creating nonpartitioned tables and indexes, the database by default uses delayed segment creation to update only database metadata and avoids the initial creation of user segments, saving disk space and greatly speeding up installation time. When a user inserts the first row into a table, the database creates segments for the table, its LOB columns, and its indexes.

Segment creation on demand saves time, space and computing resources.

Compression Advisor

In Oracle Database 11*g*, compression of your data saves disk space, reduces memory use in the data buffer cache, and can significantly speed query execution. Compression does have a cost in CPU overhead for data loading and DML. However, this cost is easily offset by greatly reduced I/O requirements.

Oracle Database 11*g* table compression is completely transparent to applications. It is especially useful in decision support systems, where there are lengthy read-only operations and large amounts of data, but it can also be used in online transaction processing systems. You can specify compression for a tablespace, a table, or a partition.

A compression advisor has been added in Oracle Database 11*g* Release 2 to facilitate choosing the correct compression level for your data. As part of the existing advisor framework in Oracle Database 11*g*, the compression advisor analyzes the objects in the database, discovers the possible compression ratios that could be achieved, and recommends optimal compression settings.

Fault Diagnostics

Oracle Database 11*g* includes an advanced fault diagnostic infrastructure for preventing, detecting, diagnosing, and resolving problems. The problems that are targeted in particular are critical errors that can affect the health of the database. When a critical error occurs, it is assigned an incident number, and diagnostic data for the error (traces, dumps, and more) are immediately captured and tagged with this number. The data is then stored in the Automatic Diagnostic Repository (ADR)—a file-based repository outside the database—where it can later be retrieved by incident number and analyzed. The extensive improvement of the fault diagnostics infrastructure in Oracle Database 11*g* aims to provide the following benefits:

- Respond proactively to small problems and prevent catastrophic system failure by alerting DBAs using health checks.
- Limiting damage and repair and interruptions after a problem is detected using the Data Recovery and SQL Repair Advisor.
- Reducing problem diagnostic time through ADR and Test Case Builder.
- Simplifying customer interaction with Oracle Support using the Incident Packaging Service (IPS) and Oracle Configuration Support Manager.

The following are the key components of the fault diagnostic infrastructure:

Automated Health Checks

A health checker framework has been added in Oracle Database 11*g* for the purposes of performing proactive checks on system health. Upon detecting a critical error, the fault diagnostic infrastructure can run one or more health checks to perform deeper analysis of a critical error. The results of a health check are stored in a report that can be viewed as a text file or as formatted HTML in a browser. The report can be added to other diagnostic data collected for the error. Separate individual health checks look for data corruptions, undo and redo corruptions, data dictionary corruption, and more.

SQL Test Case Builder

For many application problems, obtaining a reproducible test case is an important factor in problem resolution speed. The SQL Test Case Builder allows a user to automatically gather all the necessary information needed to reproduce the problem such as SQL text, PL/SQL, DDL, execution environment information, etc. The information gathered can then be transmitted to Oracle Support to help reproduce the problem.

Automatic Diagnostic Repository

The Automatic Diagnostic Repository is a file-based repository for database diagnostic data such as traces, dumps, the alert log, health monitor reports, and more. It has a unified directory structure across multiple instances and components of the Oracle database and it replaces the USER_DUMP_DEST, BACKGROUND_DUMP_DEST, and CORE_DUMP_DEST of previous releases. The diagnostic data in ADR is self-managing and is purged automatically based on predefined data retention setting. ADR also maintains meta-data for all critical errors on the database such that a user can run queries against ADR to determine what and how many critical problems occurred on the system over the last few days, months or even years.

Incident Packaging Service

The Incident Packaging Service automates the process of collecting all necessary diagnostic data related to one or more problems. Users no longer have to search in different directory locations trying to gather all the relevant trace files and dump files needed for problem diagnosis by Oracle Support. By invoking IPS, all diagnostic data (traces, dumps, health check reports, SQL test cases, and more)

pertaining to a critical error are automatically packaged into a zip file which can then be shipped to Oracle Support.

Support Workbench

The Support Workbench is a facility in Oracle Enterprise Manager that enables you to interact with the new fault diagnostic infrastructure of Oracle Database 11*g*. With it you can investigate, report, and where appropriate, repair problems, all with an easy-to-use graphical interface. The Support Workbench provides a self-service means for you to package diagnostic data using IPS, obtain a support request number, and upload the IPS package to Oracle Support with a minimum of effort and in a very short time, thereby reducing time-to-resolution for problems.

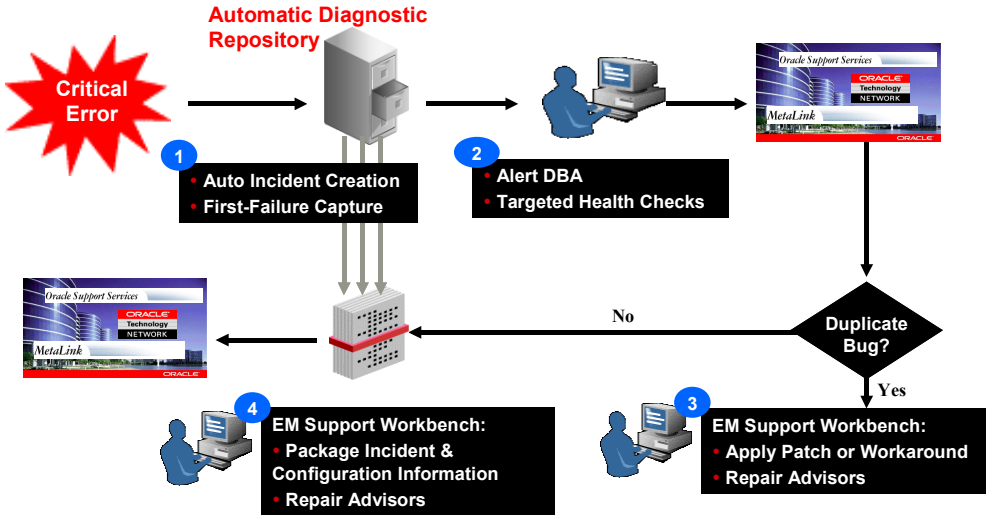

Figure 7. Support Workbench Workflow

The Support Workbench workflow consists of the following steps:

1. Create an incident in the database automatically based on the first occurrence of a failure.
2. Alert the DBA of the failure and run health checks in the areas where the failure was reported.
3. If it is a known issue, then recommend and apply any necessary patch to solve the problem.
4. Otherwise, package up incidents and relevant configuration information and upload to Oracle Support and run repair advisors to recover from failure.

There are a number of different kinds of problems that can occur in an Oracle database and the right remedy for each problem may be different. The Support Workbench has extensive workflows that guide the user to take action that is appropriate for the problem encountered.

Exadata Management and Cloud Consolidation

As enterprises increasingly look to consolidate their disparate databases onto the Oracle Exadata infrastructure, Oracle Enterprise Manager Cloud Control 12*c* can help administrators manage the Exadata Database Machine using a holistic approach to and can provide comprehensive lifecycle management from monitoring to management and ongoing maintenance for the entire engineered system.

Integrated System Monitoring

Oracle Enterprise Manager provides comprehensive monitoring and notifications to enable administrators to proactively detect and respond to problems with Oracle Exadata Database Machine and its software and hardware components. Administrators can easily adjust these monitoring settings to suit the needs of their datacenter environment. When notified of these alerts, administrators can easily view the history of alerts and associated performance metrics of the problem component, such as the network performance of an Infiniband port or the disk activity of an Exadata storage cell, to identify the root cause of the problem. With direct connectivity into the hardware components of Exadata, Oracle Enterprise Manager can alert administrators to hardware-related faults and log service requests automatically through integration with Oracle Automatic Service Requests (ASR) for immediate review by Oracle Support.

Problems that would have required a combination of database, system and storage administrators to detect in traditional systems can now be diagnosed in minutes because of integrated systems monitoring for the entire Exadata Database Machine.

Manage Many as One

Oracle Enterprise Manager provides a unified view of Oracle Exadata hardware and software where you can view the health and performance of all components such as compute nodes, Infiniband switches, Exadata storage cells, Oracle databases, ASM, etc.

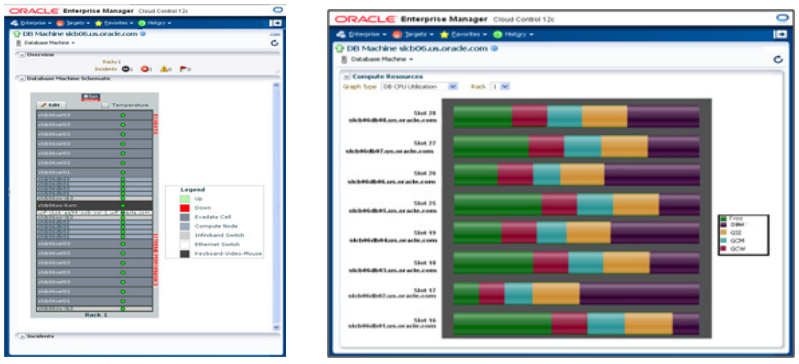

Figure 8: Monitoring Exadata using Oracle Enterprise Manager Cloud Control 12*c*

Oracle databases run transparently on Oracle Exadata Database Machine without any changes. However, there are times when a DBA needs to drilldown from the database to the storage system to identify and diagnose performance bottlenecks or hardware faults. Enterprise Manager's integrated view of the hardware and software of Exadata allows the DBA to navigate seamlessly from the database performance pages to the associated Exadata storage server to isolate the problem, whether they may be caused by a hardware component or other databases running on the same storage subsystem. The SQL monitoring capability that analyzes the performance of SQL executions in real time is Exadata aware and can pinpoint the plan operations of the execution plan that are being offloaded onto the Exadata storage servers, giving DBAs visibility into the efficiency of the SQL statement.

The Exadata management capabilities in Enterprise Manager are provided in-line with the health and performance features of the specific component being managed. For example, in addition to monitoring the performance of the Infiniband network, administrators can also alter the port settings if Enterprise Manager detects port degradation. On the Exadata storage cell, administrators can configure and activate I/O resource manager plans within Enterprise Manager if they see excessive I/O resource consumption by one particular database affecting the performance of other databases on the same set of storage cells.

Consolidation Planning

As enterprises increasingly look to consolidate their disparate databases onto the Oracle Exadata infrastructure, administrators can use Consolidation Planner in Oracle Enterprise Manager to determine optimal consolidation strategies for different Exadata configurations. Using the actual hardware configurations and the server workload history stored in Enterprise Manager, Consolidation Planner analyzes the workloads of the source systems and computes the expected utilization for the consolidation plan on the target Exadata systems. Equipped with a rich library of hardware configurations, Consolidation Planner can guide administrators to define consolidation scenarios for phantom Exadata servers, ranging from the different versions of X2-2 to X2-8. Now, businesses can make smarter and optimal decisions about the exact configurations of Exadata that is right for their database consolidation needs.

Database as a Service

Oracle Cloud Management for Oracle Database delivers capabilities spanning the entire database cloud lifecycle. It lets cloud administrators identify pooled resources, configure role-based access, define the service catalog, and the related chargeback plans. It allows cloud users to request database services, and consume them on-demand. It also allows for users to scale-up and scale-down their platforms to adapt to changes in application traffic. Finally, it lets both parties to understand the costs of the service delivered, and establish accountability for consumption of resources.

Oracle Enterprise Manager ships with an out-of-box self-service portal that allows developers, testers, DBAs, and other self service users to log on and request new single instance and RAC databases, as well as perform lifecycle operations like start/stop, status and health monitoring, etc. on them. One

can also deploy virtual assemblies containing databases on an Oracle VM virtualized server infrastructure. The portal provides access to a service catalog which lists various published service templates for standardized database configuration and versions. Users can review their past and outstanding requests, resource quotas, and current utilization as well as chargeback information for the databases they own.

Metering and Chargeback

A critical aspect of cloud delivery is the ability to establish usage cost for consuming cloud resources, and metering actual usage to deliver chargeback reports. Enterprise Manager provides tools for defining detailed chargeback plans spanning different metrics collected for each type of resource as well as defining cost centers for grouping costs across multiple developers. Chargeback plans can use not only usage based costs, but also configuration-based costs (e.g., version of the platform) or fixed costs (e.g. flat-rate management fee).

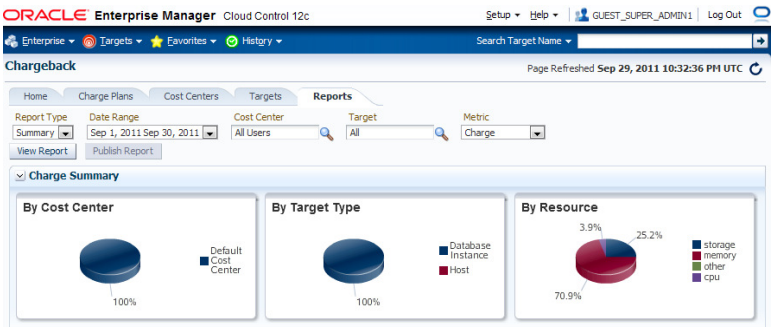

Figure 9. Chargeback for Databases

What does it mean to you?

Change and consolidation are relentless in today's rapidly evolving IT environments but it does not have to be difficult for data center managers and administrators. Thanks to the manageability features in Oracle Database 11*g* managed using Oracle Enterprise Manager Cloud Control 12*c*, database administrators can keep their systems performant and available while providing higher quality of service to their users through testing and consolidation

Conclusion

Modern enterprises are aggressively adopting new technology solutions to enhance their competitiveness and profitability. As a result, management challenges continue to rise. Oracle Database 11*g* addresses these critical challenges by enabling database administrators to maintain database performance at peak levels, adopt new technology rapidly and without risk, and increase DBA productivity and system availability by automating routine administrative tasks. Oracle Database 11*g* managed by Oracle Enterprise Manager Cloud Control 12*c* offers next-generation database management for the next-generation DBA.

Oracle Database 11*g* Release 2:
Database Manageability Overview
September, 2011
Author: Kurt Engeleiter
Contributing Authors: Jagan Athreya, Mughees Minhas, Debaditya Chatterjee

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