

Achieving a Superior Ownership Experience in Manageability and Quality for Siebel CRM

An Oracle Whitepaper
August 2008

INTRODUCTION

Oracle's Siebel CRM provides class-leading capabilities for managing sales, services, and marketing activities. To help you maximize the value of this mission critical application, and to achieve a Superior Ownership Experience in application manageability and quality, Oracle provides a set of tools that facilitate top-down application management and covers the entire application lifecycle.

TOP-DOWN APPLICATION MANAGEMENT

A key requirement for managing Siebel Application is the ability to manage the entire application stack, which includes Siebel-specific components such as Siebel Application Server and Gateway Server, as well as infrastructure components such as database and operating system. All these components must work optimally together in order to deliver availability and performance required of the Siebel Applications. Therefore, it is important that all these components be managed together.

Traditional system management tools tend to focus on a silo approach of management – handling each component individually, and then try to piece together information about the health of the application environment from the bottom up. Oracle Enterprise Manager goes beyond this bottom up approach by providing a top-down perspective also, which delivers a business-centric view of the application environment and helps you manage your Siebel Applications according to their business requirements, and allows drill down to the underlying infrastructure for further information.

This top-down approach starts with Application Management Pack for Siebel, which extends Oracle Enterprise Manager to manage Siebel Applications. Through the service level management (SLM) capabilities of the pack, your administrators can model the availability and performance requirements that your Siebel Application needs to satisfy, and then monitor according to these requirements automatically. This approach helps you focus your organization's resource on issues that are truly important – those that actually impact your business.

Application Management Pack for Siebel is complemented by other Oracle products such as Siebel Server Manager, Siebel Application Deployment Manager, Oracle Application Testing Suite, Oracle Real User Experience Insight, Oracle Database Management Packs, Oracle Middleware Management Packs, Oracle Provisioning Pack and System Monitoring Plug-in's for third party technologies to provide management coverage for your entire system environment, and support for each phase of the application lifecycle.

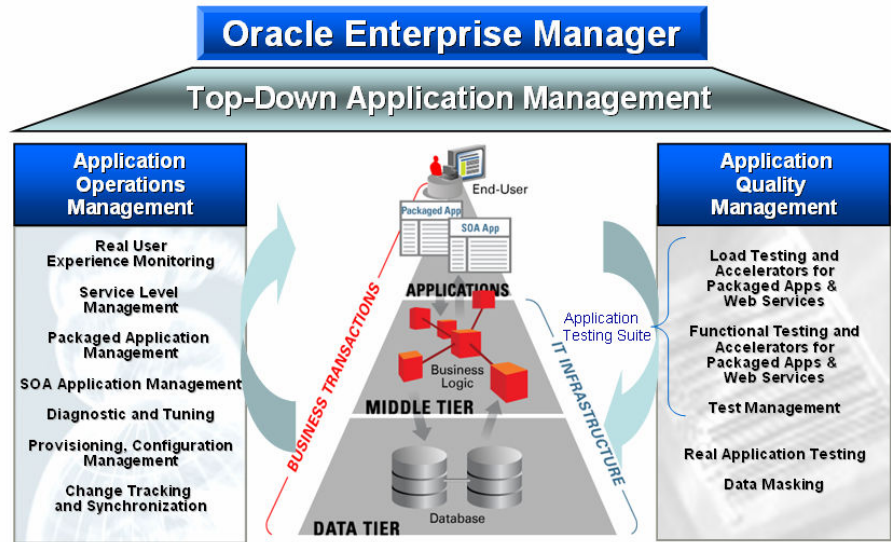


Figure 1 – Siebel Application Management Solution

COMPLETE APPLICATION LIFECYCLE COVERAGE

The deployment of Siebel CRM application goes through three distinct phases – Implement, Manage, and Optimize, and each phase presents its own set of challenges. In the very first implementation cycle, your developers may take an out-of-the-box Siebel application and make functional configuration changes. In subsequent cycles, your developers may take an already deployed Siebel application and make further functional changes, deploy a new module, or upgrade to a new release of Siebel CRM. Throughout the implementation process, your testers need to constantly test the application to make sure that it performs properly. Your team would also be constantly migrating configuration changes from development to test to staging environments. Ultimately, when you are ready to go live with your Siebel applications, your administrators would deploy your tested configuration from staging to production environment.

As you enter production, the focus shifts to management. Your administrators need to monitor the performance and availability of the application from both end user and system component perspectives. If any problem is detected, your administrators and support analysts need to triage the problem quickly in order to engage the right expert to locate problem root cause. In addition, your administrators need to monitor operational changes that are made to the environment on an on-going basis to ensure that these changes do not introduce problems into your environment.

Lastly, you need to fine tune your environment in order to achieve further optimization. The starting point of this process is a set of service level and capacity utilization reports that provide insight on the performance, availability and resource

utilization of your application. Your administrators may use the information provided by these reports to decide whether to apply software patches from Oracle, tune the database, or make other functional adjustments to the Siebel application in order to improve application end user experience. You may want to test the optimization of your application in a test environment using actual production data, in which case you need to mask sensitive information so that it is not exposed unnecessarily. As you make these optimization changes, you would also need to track the changes that you make, and maintain an audit trail for compliance purpose.

In the following pages, we'll describe how you may use various Oracle technologies to accomplish tasks in each of the three application lifecycle phases.

IMPLEMENT

Challenge 1 – Ensuring Functional Conformance to Business Requirements

Running functional tests to ensure an application's implementation conforms to business functional requirements is critical to the successful adoption of the application. However, relying on redundant manual testing is an inefficient use of quality assurance (QA) resources, especially for regression tests that need to be run over and over whenever changes are made. This is particularly true for Siebel CRM given the wide array of functionality these applications provide.

Most automated testing solutions on the market carry a steep learning curve and require that your testers become programmers in order to test. Siebel applications present even greater challenges due to the High Interactivity components utilized in the Siebel user interfaces that make many application functions impossible to automate and validate as well. As a result, most automated testing solutions available on the market fail to provide an adequate solution for automating Siebel functional testing.

Oracle Functional Testing for Web Applications combined with the Functional Testing Accelerator for Siebel provides an efficient and accurate way to automate functional testing of Oracle's Siebel CRM applications. A powerful, easy-to-use functional testing solution, Oracle Functional Testing for Web Applications accurately reproduces and validates complex end-user transactions through an intuitive record-and-playback model. The Functional Testing Accelerator for Siebel extends these capabilities by integrating with Siebel Test Automation to effectively support automation of Siebel's rich client user interface components. The core functionality in Oracle Functional Testing combined with integration of Siebel's Test Automation's CAS interface simplifies the scripting process for both Standard Interactivity and High Interactivity controls and enables users to add custom checkpoints into their script to validate Siebel application content during script playback. Oracle Functional Testing also enables automated testing at the Web Services level as an additional option, which enables users to automate testing of SOA interfaces.

Oracle Test Manager for Web Applications, another component of ATS, manages the entire test process, including test requirements, manual and automated test cases, and defects identified during testing. Oracle Test Manager for Web Applications maximizes the return on investment for your testing tools by providing a centralized repository for storing all of your test assets, which is accessible through a simple and intuitive Web-based interface and can be completely customized to fit your test process.

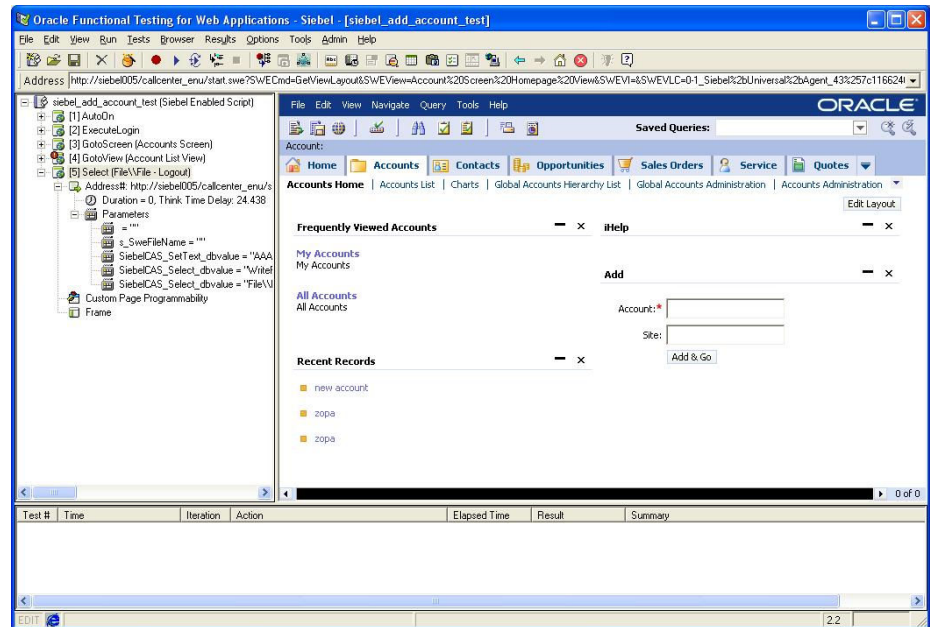


Figure 2 – Oracle Functional Testing for Web Applications

Challenge 2 – Projecting Capacity Utilization and Achieving Scalability

Load test is important to ensure that the application will scale and consume resources efficiently when it is used by a large number of users. Since Siebel CRM provides business critical functionality deployed to potentially thousands of users in an organization, ensuring application performance prior to deployment is critical. Manual testing is not an alternative for load testing since you can't realistically simulate the loads needed and it's impossible for testers to provide objective performance results. However, automated load testing for Siebel is very difficult, given the highly dynamic nature of Siebel application requests which must be accurately parameterized in order to create working load test scripts. Furthermore, as load tests are run, your testers need to be able to work with your application developers to quickly analyze performance problems that are identified during the tests in order to satisfy application performance requirements and optimize capacity utilization.

A powerful, easy-to-use performance testing solution, Oracle Load Testing can automate Siebel CRM transactions and then simulate thousands of virtual users accessing the application simultaneously to measure the effect of user load on application performance. This information is critical to inform decisions about application design, system hardware architecture and tuning options.

Oracle Load Testing integrates with Siebel Test Automation's Siebel Correlation Library to allow users to create performance testing scripts in minutes by recording real interactions with Siebel application. Siebel Correlation Library automatically identifies dynamic Siebel request parameters and converts them into Oracle Load Testing script variables. This eliminates the need to identify dynamic parameters and correlate these requests manually, which results in a dramatic reduction in scripting time. Testers can then configure one or more scripts to run with hundreds or thousands of concurrent users simulating the load that their Siebel applications would experience in production to assess performance. Oracle Load Testing for Web Applications not only stresses your application to simulate the impact of end-user workloads, but also enables rigorous validation that protocol-based legacy client/server testing tools cannot provide.



Figure 3 – Oracle Load Testing for Web Applications

As load tests are run, you can also use Application Management Pack for Siebel to observe the application's behavior under various load profiles. You may capture Siebel Application Server statistics, Server Component statistics and performance and utilization metrics of the underlying server machines. All these captured metrics would be stored in Oracle Enterprise Manager's repository, and can be used to establish performance baselines that provide context for production monitoring.

In addition, your testers and application developers may use Application Management Pack for Siebel's integrated transaction diagnostic tool to troubleshoot slow running transactions generated from the load tests. This diagnostic tool integrates with Siebel Application Response Measurement (SARM) natively, and uses the full SARM data set to provide detail insights on the various Siebel modules, screens and views, scripts, business components, and workflow invoked in processing transaction requests.

Challenge 3 – Orchestrating Controlled Deployment of the Tested Application

After functional and load tests confirmed the functional compliance and performance characteristics of the application, you are ready to have your administrators deploy the application into production. It is very important to make sure that the application that is deployed into production represents the exact configuration that was tested in functional and load tests. Otherwise, the application might not behave as it is expected to. Manually deploying an application, with its various functional artifacts such as metadata objects, web UI templates, database schema objects, list-of-values, workflow definitions, etc... can be very error prone and time consuming. After deployment is carried out, your administrators may also want to validate the proper configuration of your production environment by comparing it against the tested environment, and it is very tedious to perform this task manually.

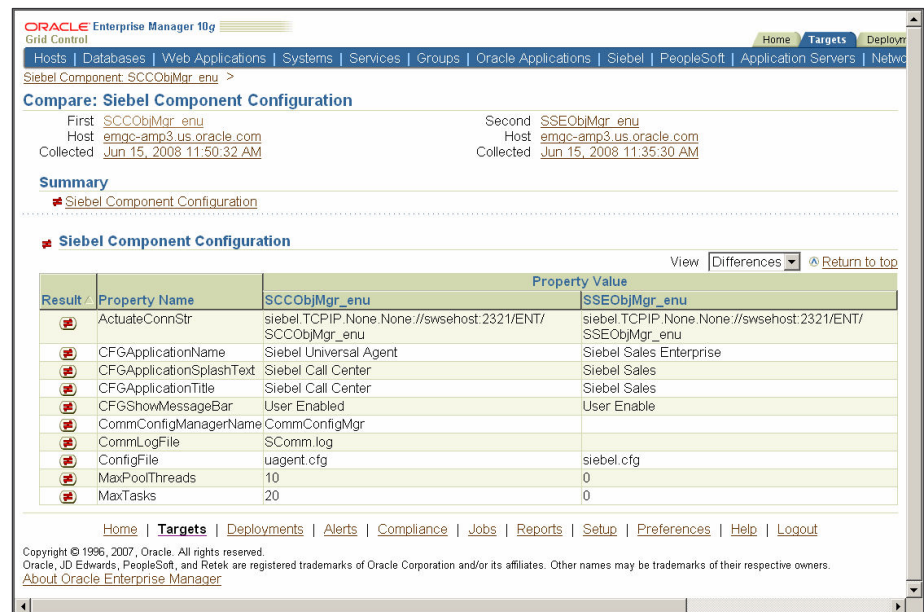


Figure 4 – Configuration Comparison

Siebel Application Deployment Manager (ADM) provides a unified framework to package up the functional contents of the application, transport and deploy them

into a production environment automatically. ADM has broad coverage of over 70 different Siebel functional configuration artifacts, eliminating the need to write separate scripts to deploy these changes. Using ADM's packager, your administrators may group changes for particular fix and deploy them as a single package. ADM also includes backup and restore capabilities that preserve settings before changes are applied, so that new changes could be rolled back if necessary.

To ensure that the Enterprise, Server and Server Component configurations in your production environment is consistent with your staging or test environments, you administrators may also use Application Management Pack for Siebel's configuration analysis tool to compare the production environment against the test or stage environments to make sure that the production setup is done according to the tested configuration.

MANAGE

Challenge 4 – Aligning IT Priorities with Business Demands

A common dilemma in organizations is balancing business needs with IT spending. IT management constantly needs to satisfy business owners while keeping a lid on spending and increasing IT efficiency. Key questions that need to be answered include:

- What are the IT dependencies of a business process? When business problems arise, how to determine if they are caused by IT issues?
- When changes are made to the application environment, what is the potential impact on the business?
- How to prioritize IT activities according to business needs?
- What is the impact of IT on business? Some key performance indicators needed to answer the question may be traditional IT system-based indicators while others may need to be derived from the business applications.

Application Management Pack for Siebel's service level management capabilities helps you define service level objectives (SLO) based on business requirements, model the end-to-end service down to the system components it depends on, monitor performance against these goals, and report on service level agreement (SLA) compliance to key stakeholders.

Service Level Objectives can be specified not only in terms of the system-level metrics for the components supporting the service, but also in terms of end user experience metrics and business KPIs imported from other systems. Application Management Pack for Siebel is unique in allowing all these classes of metrics to be used in measuring service levels. The basis for the service level management capability is a modeling facility that allows you to define a business service to be composed of component services and supporting infrastructure.

To visualize the information, a services dashboard provides real-time views into service level agreements (SLAs) and, along with other custom reports, is invaluable in communicating SLA compliance to business customers.

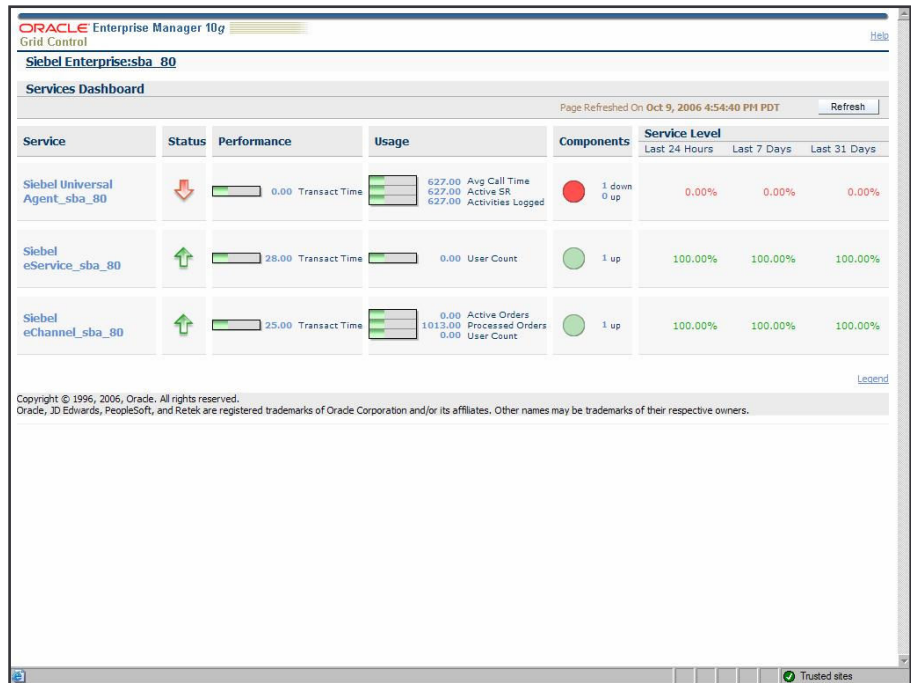


Figure 5 – Service Dashboard

Challenge 5 – Proactive Monitoring of the Complete Application Environment

In order to deliver the application service level required by your business, your administrators need to monitor your entire application environment proactively. This requires them to monitor all the components that make up your Siebel environment, including the gateway server, web server, application server, database server, server machines, network and storage devices. The key metrics that your administrators need to monitor include component up/down status, load, resource utilization, performance, exceptions such as errors and warnings etc... The monitoring needs to be carried out in a “lights out” manner with the monitoring tool alerting the administrators only when a problem occurs so that administrators are freed to concentrate on their other duties when the application is functioning normally.

Application Management Pack for Siebel anchors the solution for proactively monitoring a Siebel application environment. Using the pack, your administrators may monitor the health of your Siebel Gateway Server, Application Server, Database Server, and Web Server. Thresholds may be defined against server and component statistics such as CPU utilization, the current number of component

tasks running, and up/down status of servers and components. Log files that are associated with the various Siebel servers and components can be monitored by specifying Siebel error codes, or by defining regular expressions that match the log messages. Besides relying on information that Siebel outputs, synthetic service tests may be defined against the web servers, Siebel Server Components, the SOAP interfaces that Siebel exposes, and against third party components that Siebel relies on such as Actuate Report Server in order to monitor them actively.

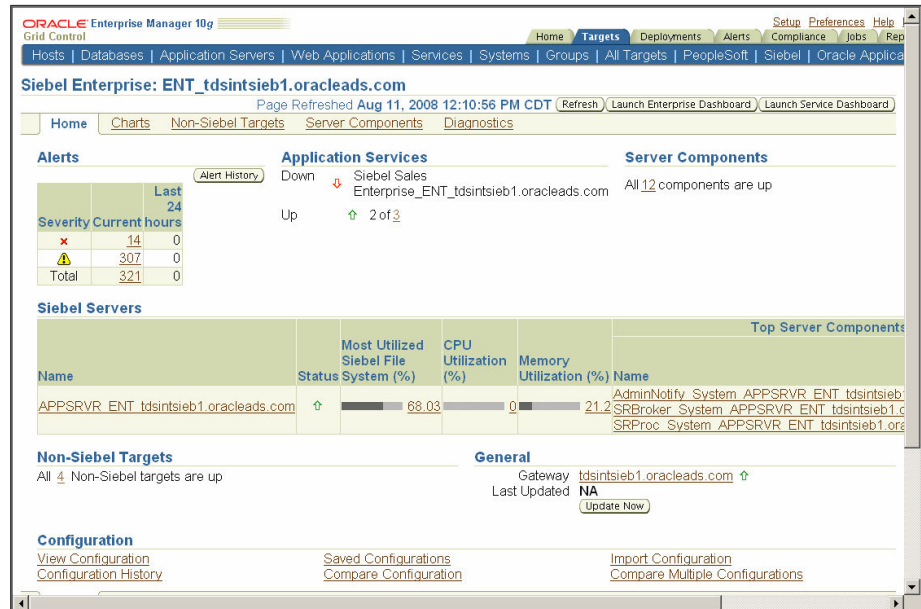


Figure 6 – Siebel Enterprise Dashboard

When monitoring the various statistics, you may rely on Application Management Pack for Siebel's built-in event management capabilities. Notification methods could be defined to send email, trigger SNMP traps to forward alerts to third party management tools, or to kick off custom scripts. Notification may be defined according to a schedule, so that different administrators who are on duties at different times would get the alerts during their shifts.

To reduce the possibility of false alarms, Application Management Pack for Siebel uses several tactics to throttle the raising of alerts. First, you may define an alert to go off only if a certain condition persists for a certain number of sampling interval. This approach prevents a singular rogue event such as a spike from triggering unnecessary alert. Second, you may define notification rule to stop sending alert after a certain number of attempts so that you don't get alerted over and over if a condition persists and you already know about it. Furthermore, you may define threshold alerts against metric snapshots so that the alerts are based on deviation from observed behavior of the components.

Besides managing the Siebel application components, Oracle Enterprise Manager provides a range of management packs and system monitoring plug-in's to cover the infrastructure components that support the Siebel applications. You may mix

and match these additional packs and plug-in's to complement the core application monitoring provided by Application Management Pack for Siebel.

If you use Oracle Database, use Oracle Database Diagnostic Pack for deep monitoring of database's functions such as tablespace, buffer pool, memory, CPU and I/O. If you use Microsoft SQL Server or IBM DB2, use System Monitoring Plug-in for Non-Oracle Database to perform similar type of monitoring.

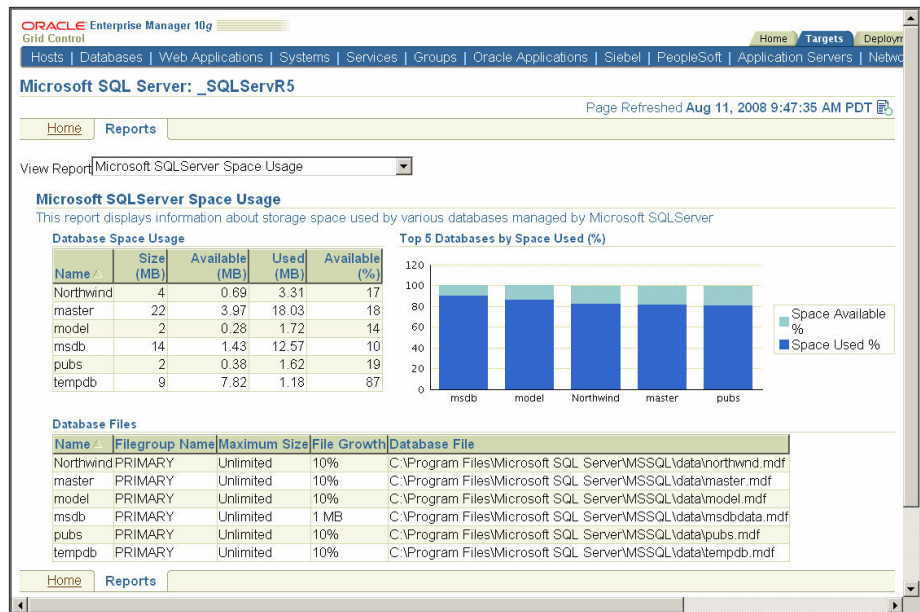


Figure 7 – Microsoft SQL Server Monitoring

If you have integrated Siebel with other software using Oracle SOA Suite or Application Integration Architecture (AIA), use SOA Management Pack to monitor the BPEL processes that orchestrate business processing across Siebel and other applications and monitor the partner links that connect BPEL with the applications.

Lastly, to monitor infrastructure technologies such as F5 Big-IP Load Balancer, EMC Storage Arrays and NetApp Filers, Oracle offers System Monitoring Plug-in for Network Devices and System Monitoring Plug-in for Storage Devices. Management data collected through these plug-in's as well as from database, middleware and SOA management packs can be combined with system and end user experience data collected from Siebel on the same Oracle Enterprise Manager instance to give Siebel administrators a holistic, top-down and end-to-end view of the entire Siebel environment and the extended infrastructure.

Challenge 6 – Monitoring End User Experience

No matter how well tuned the application is during testing, production performance problems may still occur because of unforeseen usage or interdependencies with other components of the IT infrastructure. Studies indicate

that most application performance issues are still reported first by application end users before IT administrators find out about them. Unfortunately, this delay means that business operations have been impacted.

Your administrators need to proactively identify the end user issues before the end user community is impacted by a performance problem. First step in guaranteeing end-user satisfaction is to learn about the end-user performance experience. Some of the questions that your IT staff needs to answer related to the end-user performance experience are:

- Are end-users satisfied with the application performance?
- Are end-users able to complete key business transactions successfully?
- Is the application performance problem impacting all the users or limited to a geographical region?
- How to ensure that key business transactions have consistent performance and do not have any server-side performance issues?

There are two ways to monitor your end users' experience. The first method is by using Application Management Pack for Siebel's synthetic service test. These tests are designed to simulate key end user activities such as logging in, navigating to the customer screen, and querying customer records. The tests are run via "beacons" from locations within your network to actively measure the performance and availability of your Siebel applications from end user perspectives. Because these are controlled tests and they do not rely on actual end users being present, they can be used to collect consistent data that are useful for performance trending analysis more easily.

Complementing Application Management Pack for Siebel's synthetic service test is Oracle Real User Experience Insight (RUEI). RUEI enables you to maximize the value of your application by delivering insight into real end user experiences. It can help identify lost revenue from frustrated users, reduce support costs by lowering call center volumes, accelerate problem resolution of poorly performing applications, and help businesses adapt to changing needs by providing insight into business trends and user preferences. It integrates performance analysis and usage analysis into a single offering, enabling business and IT stakeholders to develop a shared understanding into their application user experience.

Oracle RUEI's passive monitoring capability is built using state of the art Network Protocol Analysis technology, which does not require any modification, changes, or instrumentation of the application. Its passive monitoring approach allows enterprises to deploy in production, without requiring costly test/QA environment validations.

Oracle RUEI provides you with powerful analysis of network and application infrastructure. You can monitor the real-user experience, set Key Performance Indicators (KPIs) and Service Level Agreements (SLAs), and trigger alert notifications for incidents that violate them. RUEI comes with a library of powerful

reports that provide both business-orientated and technical-orientated users with the information they need to make effective decisions. Reports generated by RUEI can be consumed by line of business (LOB) owners who review and optimize the business performance, IT managers who are responsible for availability and performance of an application and IT operators who run day –to-day operations such as monitoring and diagnosing application performance. Here are RUEI’s capabilities for different usage scenarios:

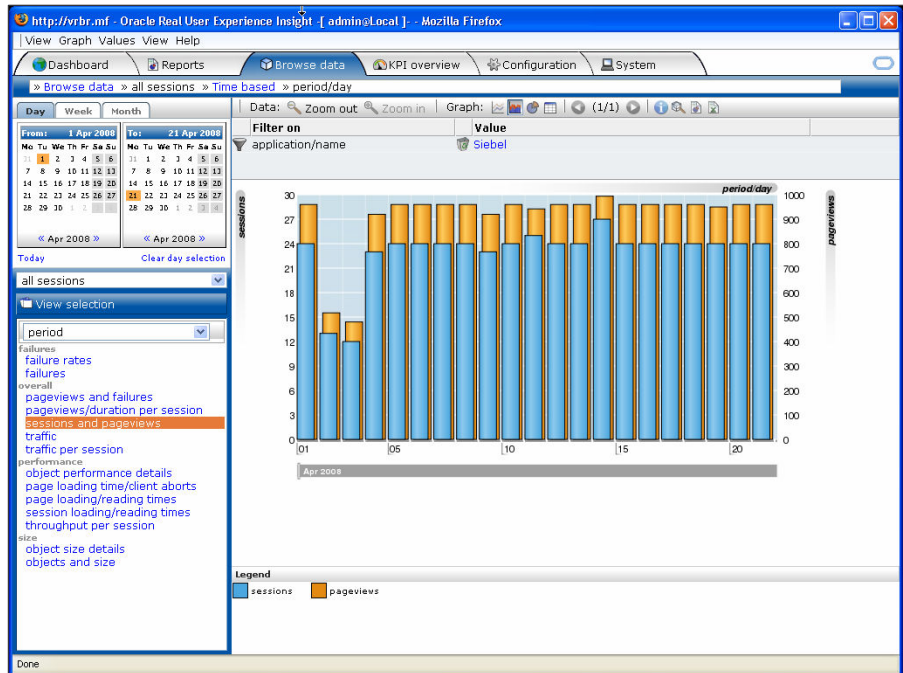


Figure 8 – Oracle RUEI’s Application Usage Analysis

Monitor Real End-user Performance

- Monitor the response times of transactions on various screens and views. View response time breakout between server time and network time end-user transactions
- Determine which parts of application are having performance problems
- Analyze how page components and objects that are contributing to overall page response time
- Satisfaction reports: end-user satisfaction reports such as frustrated page views, tolerating page views, satisfied page views and also failed views.

Application Usage Analysis

- Business transaction funnel indicates how many users were able to move from step1 of a transaction to the final step of a business transaction.
- View end user geographical regions and view the application response times by regions

- Identify transactions with failures and replay end-user transactions to view application errors
- Define and monitor KPIs and SLAs on dashboards

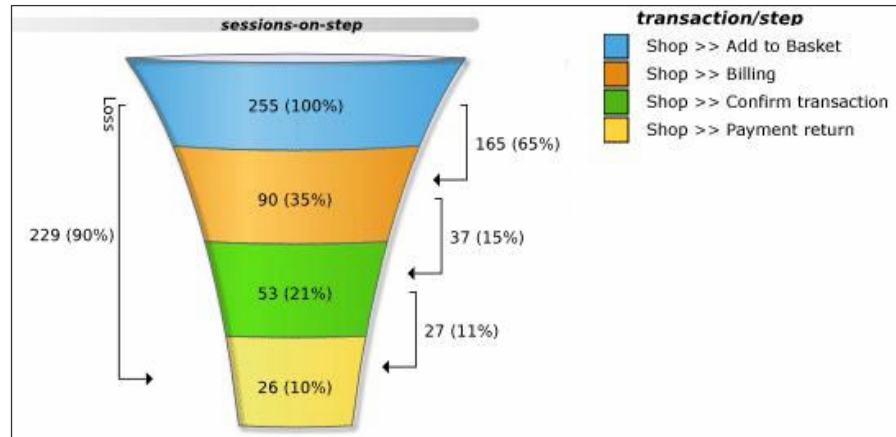


Figure 9 – Business Transaction Funnel

Using information captured by RUEI, you may determine who your users are, what parts of the applications are being used, the response times that actual end users experience, and whether they are having any trouble using the application. You may then make decisions to adjust the application or its infrastructure accordingly.

Challenge 7 – Diagnosing Production Problems Quickly

When problems are detected, you need to fix them quickly in order to minimize impacts to your end users. Problem diagnostic can be a very tedious task often involving guesswork because of difficulties in accessing pertinent diagnostic information. Diagnostic is also difficult because of the large number of components involved. As a result, diagnostic often require multiple people who manage the application, database, server, network and storage to get together to determine the problem, making the task very expensive and time consuming to perform.

Application Management Pack for Siebel simplifies diagnostics by presenting relevant diagnostic information in dashboards and providing tools to analyze information from the different parts of the application environment. The pack simplifies initial problem triage so that the task can be done quickly and with fewer people. It also provides deep diagnostic capabilities to identify problems that are rooted in the Siebel specific code.

The starting point of a diagnostic effort is the Siebel Enterprise dashboard. The dashboard provides a one page summary on the health of your entire Siebel environment, showing aggregated information on the number of servers and components having problems, number of errors and warnings raised, and number

of application services that are down. This dashboard helps you achieve an overall perspective on the environment before you proceed further to deeper investigation.

From the dashboard, you may drill down to the application services to assess whether the problem has impacted service levels. Then, begin the triage process by examining service test data to see whether the problem is network location specific. If it is network specific, you may then engage the network administrator to resolve the problem. If not, you may want to bring up metric history information of the various servers and components to see if the problem is due to over utilization or lack of resource. Application Management Pack for Siebel automatically saves all the metrics that it collected from your application and its environment, so you can go back to a point in time to examine the state of the system when the problem occurred.

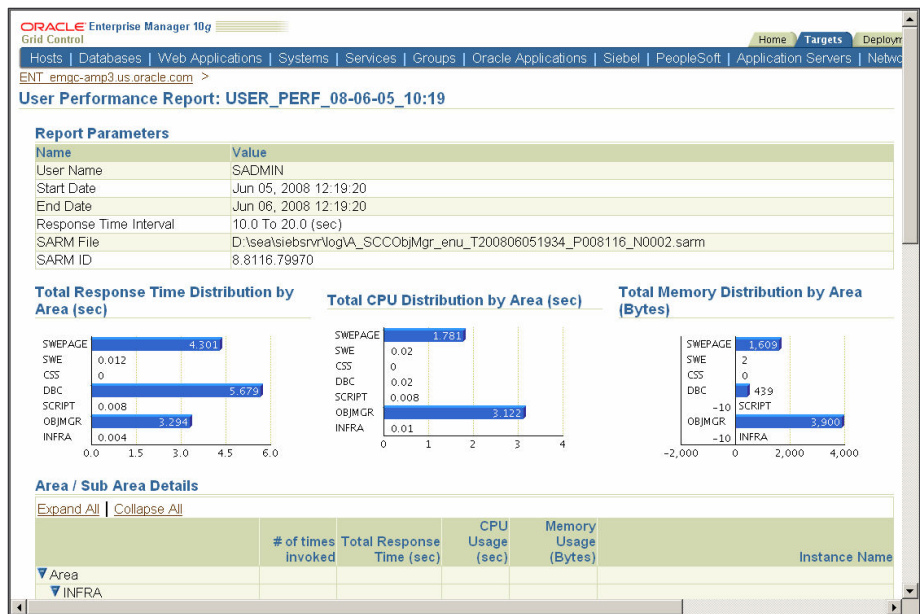


Figure 10 – Siebel Transaction Diagnostics

For problems that are more intermittent or are tied to specific transaction requests or users, use Application Management Pack for Siebel’s transaction diagnostic tool to get detailed information at the transaction request level. You may retrieve transaction requests from a particular end user or a particular server, identify the requests that are slow running, and drill down to a particular request to find out which screen and view the user accessed, the name of the business component, and the script or workflow that was the bottleneck.

For problems that may be system configuration related, use Application Management Pack for Siebel’s configuration analysis tool to locate the cause. You may query against Oracle Enterprise Manager’s configuration management database (CMDB) to find out whether any Siebel Enterprise, server or component parameter has changed. You may also compare configuration settings across different server

components, between servers, or even against different Siebel Enterprises to find out why there are discrepancies in behavior amongst different environments.

OBJMGR_B SVC_INVOKE	1	0.0	0.0	-27	Web Engine Interface
OBJMGR_B SVC_INVOKE	4	0.0	0.0	0	Web Engine UI Preferences
OBJMGR_B SVC_INVOKE	1	0.0	0.0	0	Workflow Process Manager
OBJMGR_B SVC_INVOKE	1	0.0	0.0	0	AMP Transform Order
OBJMGR_B SVC_INVOKE	1	0.0	0.0	0	AMP Load Order
OBJMGR_B SVC_INVOKE	1	0.0	0.0	0	AMP Submit Order
OBJMGR_B SVC_INVOKE	11	0.0	0.0	0	Task UI Service (SWE)
OBJMGR_B SVC_INVOKE	40	0.0	0.0	0	Web Engine State Properties
OBJMGR_B SVC_INVOKE	3	0.0	0.0	0	Web Engine User Agent
▼ SWE					
SWE_CMD_SWEUAID	1	0.0	0.0	12	Syeyj82Rj02SHzb5B0egwRobLL8r-S uY4hDDmHztNs_
▼ SWEPAGE					
SWEPAGE_VIEW_LAYOUT	1	0.0020	0.0	18	Sales Order-Browse Catalog View
SWEPAGE_APPLET_SHOW	1	0.0030	0.0	16	Catalog Order Header Form Applet
SWEPAGE_APPLET_SHOW	1	0.0	0.0	3	UMF Message List Applet - SI
SWEPAGE_APPLET_SHOW	1	0.0	0.0	3	Sales Order Catalog List Applet
SWEPAGE_APPLET_SHOW	1	0.0	0.0	5	COM ActiveX Placeholder Applet - Sales Order
SWEPAGE_APPLET_SHOW	1	0.0	0.0	7	Order Entry - Line Item List Applet (Sales) - Short
▼ WORKFLOW					
WORKFLOW_ENGNE_INVOKE	1	0.0	0.0	0	
WORKFLOW_ENGNE_INVOKE	1	0.0	0.0	0	AMP Submit Order
WORKFLOW_CORDR_EXECUTE	1	0.0	0.0	0	AMP Submit Order
WORKFLOW_STEPS_EXSTEP	5	0.011	0.01	-375	AMP Submit Order
▼ SCRIPT					
SCRIPT_J_EXECUTEALL	2	0.0	0.0	-1	Service [AMP Transform Order]
SCRIPT_J_EXECUTE_EVENT	1	9.999	9.934	128	Service [AMP Transform Order]
SCRIPT_J_EXECUTEALL	2	0.0	0.0	-1	Service [AMP Load Order]
SCRIPT_J_EXECUTE_EVENT	1	3.0	2.814	127	Service [AMP Load Order]
SCRIPT_J_EXECUTEALL	2	0.0	0.0	-1	Service [AMP Submit Order]
SCRIPT_J_EXECUTE_EVENT	1	1.998	1.922	128	Service [AMP Submit Order]

Figure 11 – Detailed Visibility into Transaction Requests

If you use Oracle Database, you may use Oracle Database Diagnostic Pack to carry out deep database level diagnostics. The pack includes a self-diagnostic engine built right into Oracle Database kernel, called Automatic Database Diagnostic Monitoring (ADDM). ADDM periodically examines the state of the database, automatically identifies potential database performance bottlenecks, and recommends corrective actions. Oracle Database Diagnostic Pack presents ADDM’s findings and recommendations in a convenient and intuitive fashion, and guides administrators step-by-step to quickly resolve performance problems by implementing ADDM’s recommendations. 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 to determine the root cause of problems. The problem classification tree used by ADDM encapsulates decades of performance tuning experience of Oracle’s own performance experts and it has been specifically designed to accurately diagnose the most frequently seen problems, such as CPU and I/O bottlenecks, poor connection management, undersized memory, resource intensive SQL statements, lock contention, etc... Each ADDM finding has an associated impact and benefit measure to enable prioritized handling of the most critical issues. To better understand the impact of the findings over time, each finding has a descriptive name that facilitates search, a link to number of previous occurrences of the finding in the last 24 hours, and affected instances.

If you have augmented your Siebel application with Java or SOA technologies, use Oracle Middleware Diagnostic Pack to troubleshoot OC4J or Weblogic containers, or Non-Oracle Middleware Diagnostic Pack to diagnose your IBM Websphere or JBoss servers. For either sets of servers, you may use Oracle Advanced Diagnostic for Java (AD4J), which is part of both middleware diagnostic packs, to troubleshoot performance problems. These problems may include inefficient database locks, SQL statements, slow performing Java methods, memory leaks, or invocations to the Siebel Java DDK.

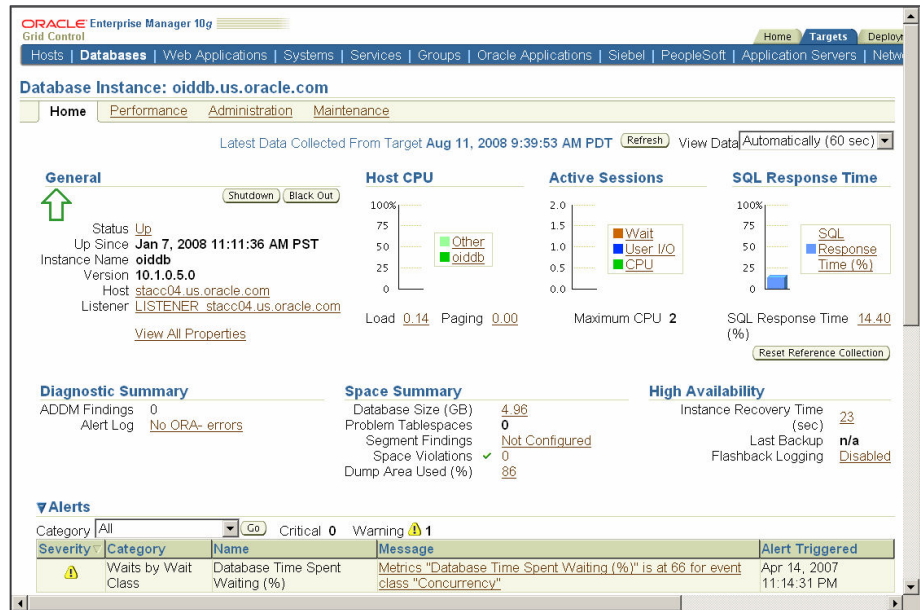


Figure 12 – Oracle Database Diagnostics

OPTIMIZE

Challenge 8 – Making Fact-Based Optimization Decisions

Optimizing an application is a time consuming task often surrounded by myths and legends, few of them based on facts. Like diagnostics, application optimization is very hard to do unless you have access to the right information. Application Management Pack for Siebel, along with Oracle Database Tuning Pack, provide the information that you need to make fact-based optimization decisions.

The starting point of the optimization process is Application Management Pack for Siebel's service level management reports. Based on service level indicators collected from the application over a period of time, these reports indicate whether Siebel applications have provided the performance and availability needed to support critical business operations. These reports are further complemented by capacity utilization reports of the underlying system components, and by RUEI's application usage reports that show the usage patterns of the application.

With these information, you may then decide whether you need to invest in further optimization, which may include tasks such as adjusting the functional configuration of your application, applying patches from Oracle, tuning Siebel Application Server and Server Components, or tuning the database.

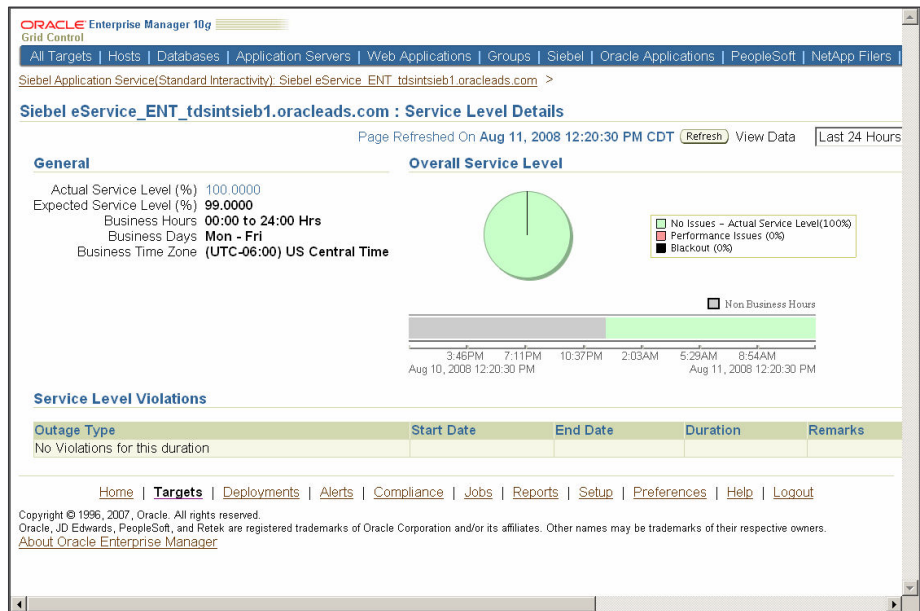


Figure 13 – Service Level Report

To optimize Siebel Server Components, you need to consider several statistics collected during run-time. These statistics are gathered by Application Management Pack for Siebel and are stored in Oracle Enterprise Manager’s repository. You may retrieve them in reports that show the graph of these metrics over time to understand how the application behaves or compare the metrics across different servers to see if your servers are load balanced properly. In addition, on Siebel 8, you may analyze SARM data by server component, and find out how CPU and memory are consumed per transaction request, and the amount of database activities each transaction request generates. Using these information, you may work with your application developers to modify your application’s functional configurations if they prove to be too resource intensive, or use Siebel Server Manager to adjust component definitions or server parameters accordingly.

For tuning the database, use Oracle Database Tuning Pack if you run Siebel on an Oracle database. Even though Siebel dynamically generate SQL statements for database access, you may still effect changes to database performance through the use of indices, database system component tuning, and SQL profiling to tune the execution plans.

Manual SQL tuning is a complex process that presents many challenges. It requires expertise in several areas, is very time consuming, and requires an intimate knowledge of the schema structures and the data usage model of the application.

All these factors make manual SQL tuning a challenging and resource intensive task that is ultimately very expensive for businesses.

SQL Tuning Advisor is Oracle's answer to all the pitfalls and challenges of manual SQL tuning. It automates the SQL tuning process by comprehensively exploring all the possible ways of tuning a SQL statement. The analysis and tuning is performed by the database engine's significantly enhanced query optimizer.

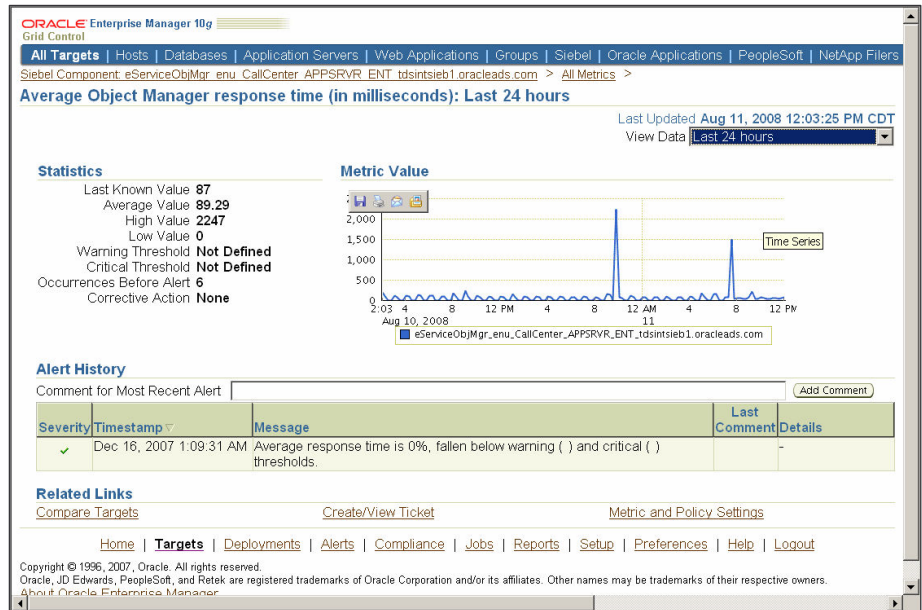


Figure 14 – Siebel Component Statistic History

These analysis performed by SQL Tuning Advisor are applicable to Siebel:

- **Statistics Analysis:** The query optimizer needs up-to-date object statistics to generate good execution plans. In this analysis objects with stale or missing statistics are identified and appropriate recommendations are made to remedy the problem.
- **SQL Profiling:** This feature, introduced in Oracle Database 10g, revolutionizes the approach to SQL tuning. Traditional SQL tuning involves manual manipulation of application code using optimizer hints. SQL Profiling eliminates the need for this manual process and tunes the SQL statements without requiring any change to the application code. This ability to tune SQL without changing the application code also helps solve the problem of tuning packaged applications. Packaged application users now no longer need to log a bug with the application vendor and wait for several weeks or months to obtain a code fix for tuning the statement. With SQL profiling the tuning process is automatic and immediate.
- **Access Path Analysis:** Indexes can tremendously enhance performance of a SQL statement by reducing the need for full table scans. Effective

indexing is, therefore, a common tuning technique. In this analysis new indexes that can significantly enhance query performance are identified and recommended.

The output of this analysis is in the form of recommendations, along with a rationale for each recommendation and its expected performance benefit. The recommendation relates to collection of statistics on objects, creation of new indexes, restructuring of the SQL statement, or creation of a SQL Profile. A user can choose to accept the recommendation to complete the tuning of the SQL statements.

Oracle Database Tuning Pack 11g also provides the ability to reorganize objects. Managing the space usage of your tablespaces efficiently by removing wasted space is not only a good space management practice but it also enhances performance by reducing unnecessary disk I/Os. Reorganization is used for:

- Rebuilding indexes and tables that are fragmented
- Relocating objects to another tablespace
- Recreating objects with optimal storage attributes

Oracle Database Tuning Pack 11g provides a wizard that can perform reorganization at schema and tablespace levels, and gives the option for both online and offline reorganization. The wizard also provides an impact analysis report as well as a review script that contains the exact operations that will be performed. This helps users to precisely understand the implications of the operation before implementing it.

Challenge 9 – Masking Production Data to Test Optimization Changes

In carrying out optimization tasks, it is often beneficial to use real production data in order to assess optimization impacts accurately. However, using real data may raise information security and privacy concerns. Safeguarding production data and preventing leaks of confidential or sensitive information to non-production users has become a corporate imperative for all organizations – thanks to the slew of global regulations governing data privacy. The Sarbanes Oxley Act of 2002 in the US or the Financial Instruments Exchange Law (FIEL) of Japan (also called J-SOX) provides enhanced standards on internal controls for corporate information. The Health Insurance Portability and Accountability Act (HIPAA) of 1996 in the US or the European Union’s Data Protection Directive are a part of the global laws governing the privacy of personal data related to individuals. Even credit card payment processors have adopted Payment Card Industry (PCI) standards regarding the use and sharing of credit card information.

If you use Oracle Database, you may use Oracle Database Masking Pack to overcome this problem. Data Masking Pack helps you obfuscate sensitive data selectively, preserving the realism of test data set while protecting sensitive

information at the same time. Data masking rules are highly configurable, and you may control the algorithm for masking the data in order to preserve relevant data semantics that are useful to creating realistic test scenarios.

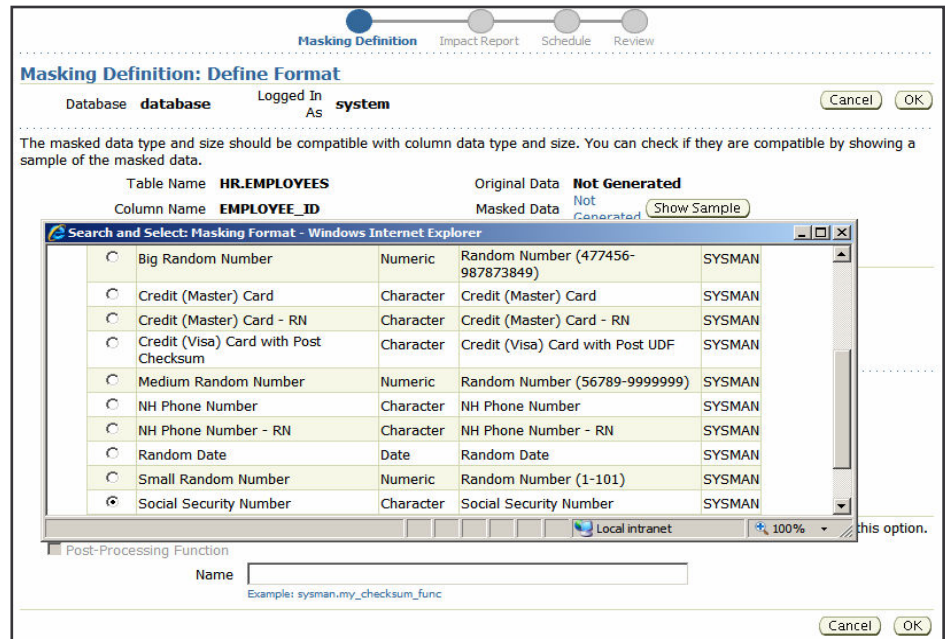


Figure 15 – Data Masking Definition

Data Masking Pack uses an irreversible process to replace sensitive data with realistic-looking but scrubbed data based on masking rules and ensures that the original data cannot be retrieved, recovered nor restored. The Data Masking Pack helps maintain the integrity of the application while masking data.

The Data Masking Pack provides out-of-the-box mask primitives for various types of data, such as random numbers, random digits, random dates, constants.

Organizations can also use other built-in masking routines, such as shuffling, which shuffle the value in a column across different rows. This is useful when the range of values in a column is often not known and the shuffling of values in the same table provides a sufficient degree of privacy protection. For organizations require that the masked value be realistic but not based on the original data, the Data Masking Pack can replace the original data, such as names and addresses, with data containing fictitious names and addresses derived from external data sources.

Organization with specialized masking requirements can also add user-defined mask formats to the collection of the mask formats. These user-defined formats, defined using PL/SQL, provide an unlimited degree of flexibility in generating mask format appropriate to the business or the industry segment that the organization operates in. Information security administrators can then create complex and composite masks based on combinations of various masking formats - both standard and user-defined. For example, a mask for common credit card

numbers can be defined as unique sixteen digit numbers that begin with 4 or 5, which are then verified for check-sums to be compliant with PCI standards.

Data Masking Pack uses a highly efficient and robust mechanism to create masked data. The Data Masking perform bulk operations to rapidly replace the table containing sensitive data with an identical table containing masked data while retaining the original database constraints, referential integrity and associated access structures, such as INDEXes, PARTITIONs, and access permissions, such as GRANTs. Unlike masking processes that are traditionally slow because they perform table updates, the Data Masking Pack takes advantage of the built-in optimizations in the database to disable database logging and run in parallel to quickly create a masked replacement for the original table. The original table containing sensitive data is dropped from the database completely and is no longer accessible.

Challenge 10 – Managing Configuration Changes and Achieving Compliance

As optimization changes are made against server and component parameters, it is important to be able to keep track of the changes for diagnostic and compliance purpose. Traditionally, people have relied on manual methods of maintaining change history, often keeping the information in spreadsheets. The manual approach is very tedious, and is often inaccurate. Sometimes, changes are made temporarily for testing purpose, but end up becoming permanent as the person who made the change forget to back it out, and this causes what is known as configuration drifts that can impact application performance and availability over time.

Application Management Pack for Siebel's configuration management capabilities automates configuration management activities. It provides a view of configuration items and their dependencies within and across each other. Manage configuration drift through scheduled comparison with "gold configuration" baselines. Administrators can track, analyze and report on configurations while capturing and storing configuration data that is used for the automation of the entire change management process.

Application Management Pack for Siebel's configuration management support is the foundation to all Service Support processes, enabling effective incident management, problem management, change management, release management, service level management and availability management. It enables faster mean-time-to-repair through root cause analysis by isolating and correlating problems to the exact infrastructure or application component that is causing failure and by auditing change history for all targets and parameters. Reduces the risks involved in rolling out changes to production environments by identifying the impact of changes on deployed applications and users.

Here are some of the key features of configuration management capabilities:

- Automated discovery of Siebel servers such as application server and gateway server, and their association with the underlying host and operating system
- View and analyze Siebel Enterprise, application server and server component configurations
- Out-of-box and customizable configuration searches
- Compare configurations
- Historical change tracking
- Configuration reports

Configuration History
 Enterprise Manager automatically collects configuration information for targets such as hosts and databases. Changes to these configurations are recorded and may be viewed from this page. Page Refreshed Aug 11, 2008 10:03:08 AM PDT [Search Using SQL](#)

Category: Siebel Component
Changing the category clears any existing results.

Search

Target Name: is exactly SCCObjMgr_enu_CallCenter... Change Discovered after: [] [] AM C PM

Target Property: Deployment Type [] Change Discovered before: [] [] AM C PM

On Host: contains []

Member Of: contains []

Type of Change: is All []

View History Records: Grouped

Change Discovered	Target Name	On Host	Category	History Records
Jan 9, 2008 4:25:59 AM CST	SCCObjMgr_enu_CallCenter_APPSRRV_ENT_emgc-amp3.us.oracle.com	emgc-amp3.us.oracle.com	Siebel Component: Siebel Component Configuration	4
Dec 12, 2007 2:04:14 AM CST	SCCObjMgr_enu_CallCenter_APPSRRV_ENT_emgc-amp3.us.oracle.com	emgc-amp3.us.oracle.com	Siebel Component: Siebel Component Configuration	1
Dec 7, 2007 12:40:27 AM CST	SCCObjMgr_enu_CallCenter_APPSRRV_ENT_emgc-amp3.us.oracle.com	emgc-amp3.us.oracle.com	Siebel Component: Siebel Component Configuration	4
Nov 12, 2007 12:21:31 AM CST	SCCObjMgr_enu_CallCenter_APPSRRV_ENT_emgc-amp3.us.oracle.com	emgc-amp3.us.oracle.com	Siebel Component: Siebel Component Configuration	1
Nov 11, 2007 5:14:00 AM CST	SCCObjMgr_enu_CallCenter_APPSRRV_ENT_emgc-amp3.us.oracle.com	emgc-amp3.us.oracle.com	Siebel Component: Siebel Component Configuration	1
Aug 28, 2007 8:22:29 PM CDT	SCCObjMgr_enu_CallCenter_APPSRRV_ENT_emgc-amp3.us.oracle.com	emgc-amp3.us.oracle.com	Siebel Component: Siebel Component Configuration	2
Aug 26, 2007 2:09:01 PM CDT	SCCObjMgr_enu_CallCenter_APPSRRV_ENT_emgc-amp3.us.oracle.com	emgc-amp3.us.oracle.com	Siebel Component: Siebel Component Configuration	2

Figure 16 – Configuration Change History

SUMMARY

Through Oracle Application Management Pack for Siebel and other Oracle management and testing products, you can start centralizing the management of your Siebel CRM applications on Oracle Enterprise Manager. These products complement bundled application tools, such as Siebel Server Manager and Siebel Application Deployment Manager, which provide tactical administrative functions. The management packs leverage Oracle Enterprise Manager's top-down application management capabilities to facilitate proactive management and ITIL best practices implementation that cover the complete application lifecycle. You can use Oracle Enterprise Manager as the unified console to manage your entire application infrastructure, including all your application instances, the SOA-based fabric that

you use to connect your applications, both Oracle and non-Oracle databases and middleware, as well as your servers, storage and network devices, all of which impact your application's performance and availability. Through these tools, you can achieve a Superior Ownership Experience in manageability and quality for your applications, and deliver the application service level required to meet your business needs.



Achieving a Superior Ownership Experience in Manageability and Quality for Siebel CRM
August 2008

Authors: Chung Wu, Rajagopal MARRIPALI

Contributing Authors: Rajiv Taori, Joe Fernandes

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

Copyright © 2008, Oracle. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice.

This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission. Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.