CHAPTER 24

Performance Tuning for Oracle R12 Financials
ne of the greatest challenges of large corporate information systems—aside from security and disaster recovery—lies in the performance of the mission-critical applications that run as part of the Oracle E-Business Suite. Performance optimization is a large and complex subject that requires a book in itself to explain in depth. We will provide a set of best practices and tips to help you gain the most performance from your Oracle E-Business Suite environment. We will explain how to tune the overall environment with a focus on the following key areas:

- methodologies for performance tuning,
- a holistic tuning approach,
- system performance optimization, and
- application and database performance tuning.

First we will discuss how to implement an effective method to optimize performance with the Oracle E-Business Suite of applications, since without a solid framework, you will be constantly in reactive “firefighting,” mode which is both ineffective and counterproductive over the long term.

**Methodology for Performance Tuning the Oracle E-Business Suite**

Establishing a method of best practices for performance tuning the Oracle E-Business Suite environment requires that you adopt a framework in concert with the key stakeholders of your IT organization. At a minimum, it is wise and prudent to examine the following topics in developing a roadmap for performance analysis with Oracle EBS:

- load testing before, during, and after go-live;
- functional requirements and testing;
- map the business and functional environments to the technical environment;
- the thinking of businesspeople compared to that of technical people;
- the goal of unify to resolve tuning issues (Method R);
- stress testing;
- integration testing; and
- unit testing.

**Load Testing for Oracle E-Business Suite**

Unfortunately, the subject of load testing is rarely if ever given consideration during the implementation of Oracle R12 EBS environments in most organizations. It is an afterthought—that is, until major performance issues appear to adversely affect the operational aspects of the Oracle Financials production systems. At this point, it places the organization at risk and forces the IT staff to react to the issues at hand, which causes stress, frustration, and an end result of just a temporary solution rather than a root-cause analysis and solution to the fundamental problem. It is wise to implement load testing at all phases of the project, from initial deployment of the Oracle E-Business Suite through go-live.
Chapter 24: Performance Tuning for Oracle R12 Financials

Why Do We Load Test?

Load testing provides numerous benefits as part of a performance effort to develop a proactive approach to highly optimized Oracle EBS systems. It allows you to generate metrics before, during, and after go-live to optimize performance and availability for Oracle EBS applications. For example, you can understand how long the Oracle GL batch jobs take to run on average at peak hours versus during the night. It also helps to create an accurate baseline of performance so that when changes are made to the environment, you can assess why performance has changed. For instance, if you determine baseline performance via load testing and then apply a new patch to Oracle Financials, you can truly know whether or not the patch has adversely affected performance for the Oracle EBS environment.

**TIP**

*You really do not know the true performance until you test!*

In order to measure load performance for Oracle EBS, you can use third-party load-generator tools such as HP LoadRunner to create custom scripts to simulate a real user environment and measure application response time based on a set number of users logged into the Oracle EBS environment. You can also use the Oracle 11g Real Application Testing Suite, which is excellent for database-tier load-testing scenarios, or the Oracle Application Management Pack, which is useful for measuring application-tier performance metrics.

Now let's discuss how to apply a methodology for functional and technical aspects of performance-test analysis.

Functional to Technical Specifications and Testing

Oracle database administrators have the challenging task of mapping business requirements onto technical results to optimize performance for the Oracle E-Business Suite environment. Part of the difficulty is due to the gap in understanding between the various parties and stakeholders within the IT organization who are responsible for the care and feeding of the Oracle E-Business Suite. Businesspeople such as financial analysts, project managers, and CFOs do not think like technical people. They live in a completely different world, apart from the technical staff. This creates unintended consequences leading to conflict, confusion, and unique problems in the operational aspects of maintaining a high-performance environment with Oracle E-Business Suite applications. The first step in the mapping is to eliminate the disconnect between the functional, development, and technical teams as soon as possible in the deployment process, to reduce bottlenecks in performance.

We recommend that you do the following to bridge this gap in knowledge and communication:

- Establish stakeholders to review and test the business requirements from the functional and business units mapped out to the technical teams. Involve all parties from the users, developers, and functional users in regularly held staff meetings and review sessions.
- Document and review the business requirements for modules deployed (e.g., Accounts Payable, Accounts Receivable, and General Ledger) to really understand what to test and measure for performance tuning.

Now that we have given consideration to the functional aspects of performance for the Oracle E-Business Suite, the next logical step is to decide exactly what we need to tune.
What to Tune?

Understanding where performance tuning needs to begin is paramount to successful optimization of the Oracle E-Business Suite. As the saying goes, “Knowing is half the battle.”

There are two situations that affect how and when to tune. We recommend that you start with the initial business problem at hand and work out an action plan toward remediation. For instance, if the Oracle EBS database administrator receives a phone call from a user complaining about slow response time for invoices processed with Oracle Financials, then it warrants a closer look at tuning efforts regarding the Accounts Payable and Accounts Receivable modules within the Oracle E-Business Suite. It is useful to drill down from a macro level to micro areas of the environment to identify the low-hanging fruit for optimizing performance in such a situation.

Never assume that it is the database or another technical component until you have completed a root-cause analysis. We will use a holistic approach to performance tuning as the capstone for performance optimization of the Oracle E-Business Suite.

Figure 24-1 shows the architecture for the Oracle Release 12 E-Business Suite.

![Figure 24-1. Oracle R12 EBS architecture and file-system layout](image)

Holistic Approach to Performance Tuning

There are various methods to performance tuning the Oracle E-Business Suite. By adopting the holistic method, you will use best practices in a proactive manner to alleviate the majority of performance issues before they become major headaches. This will save the technical and functional staff many sleepless nights and ensure a healthy environment. Figure 24-2 shows the overall picture of holistic tuning for Oracle.

By understanding the real problem first, you can drill down from the macro level to micro layers and arrive at root-cause solutions. Experts in the Oracle field, from Cary Millsap and Method R to the OakTable Network and the Battle Against Any Guess movement, agree that it is critical to identify the real business problem first before leaping to illogical conclusions. This avoids the fallacy in the fable of the blind men and the elephant, which would otherwise
convolute the issue at hand. A holistic approach to tuning allows you to drill down to the root cause and perform useful and interesting analyses while filtering out any white noise. It is also important to develop test cases so that you can duplicate the identified performance problem. We are focused on methodology and not silly “silver bullets.” In fact, Oracle recommends that you apply the holistic approach to tuning Oracle E-Business Suite via the My Oracle Support Note 69565.1: A Holistic Approach to Performance Tuning Oracle Applications Systems.

**System-Performance Optimization**

Many Oracle database professionals jump to application and database performance tuning before looking into system-performance analysis. This is a fatal mistake because many issues that cause performance degradation occur at the hardware, system, network, and storage layer. For instance, the Oracle application tier requires solid network connectivity to access the database-tier server. Therefore, if network latency is severe, application performance will suffer because the application services will hang during batch processing. In addition, if the disk setup is flawed on the storage-area network, read and write performance will be poor, causing slow application performance as a result. In this case, it behooves the application and database team to establish a close working relationship with the system and network infrastructure team to perform effective design and proactive tuning activities that can remediate performance issues before they occur. In summary, you should consider the following areas carefully for performance optimization with Oracle infrastructure:

- hardware server configuration,
- storage configuration, and
- network design and configuration.

First let’s take a look at hardware performance.

**Hardware Performance Considerations**

One area that is frequently overlooked in sizing Oracle R12 EBS environments is the hardware server components. Most vendors such as IBM, Dell, Cisco, and HP have sizing tools available to plan for new Oracle R12 EBS environments. These give a starting point for sizing hardware for a
new deployment of the Oracle E-Business Suite. We recommend that you obtain sizing recommendations from different vendors and compare benchmarks for performance to arrive at the best decision for your particular Oracle EBS implementation. One of the most frequently asked questions regarding hardware is whether to use a big centralized server or smaller blade-type servers. For the application tier, we recommend multiple smaller servers with a hardware load balancer to best distribute performance across the tech stack. For the database tier, we recommend a large single server or multiple large servers in clustered configuration for scale-out requirements via Oracle Real Application Cluster. More memory is not always better, but you do want to ensure that you have enough. How do you know how much memory and CPU resources to use for the servers? Initial load testing is your best bet to assess if you have sufficient memory and CPU resources, along with sizing estimates from hardware vendors to determine the sweet spot for your hardware server configuration. Tools such as top, sar, vmstat, iostat, and Oracle Enterprise Manager are useful yardsticks to measure resources and consumption during load testing and operations after go-live. As a parting statement regarding hardware server sizing, multiple cores (quad-core configurations) are excellent for CPU performance, but as a downside, they may have additional Oracle licensing requirements.

Operating-System Tuning
Tuning the operating system (OS) is an area that is commonly neglected during Oracle EBS implementation. As a result, performance problems occur because key operating-system parameters are not tuned. The approach for performance tuning the OS varies based on the platform. For example, Linux, Unix, and Windows have different requirements and items that are unique to each platform and must be tuned correctly. We recommend that you consult the My Oracle Support Note 224176.1: How to Use OS Commands to Diagnose Database Performance Issues? to identify the specific tools for measuring OS performance. As a general rule of thumb, for Linux and Unix platforms we recommend that you start with the following steps:

- Measure overall system performance with top and iostat.
- Check memory and configuration for kernel settings with dmesg and sysctl commands.
- Drill down if necessary with the strace utility.

Linux Performance Tuning
Linux is a popular operating system used in many large Oracle E-Business Suite deployments. Oracle has migrated from development on Solaris Unix to Linux recently as the chosen platform for testing applications. Since a complete treatment of performance tuning with Linux is beyond the scope of this chapter, we recommend that you consult the following My Oracle Support notes (http://support.oracle.com) to tune Linux issues:

Network Performance

- Note 274953.1: Tuning TCP/IP Parameter in Linux Box for SQLNET
  Use netstat and ping to identify latency issues with networks.
- Note 560590.1: How and When to Use the net.ipv4.tcp_rmem and net.ipv4.tcp_wmem Linux Kernel Parameters
Chapter 24: Performance Tuning for Oracle R12 Financials

Kernel Tuning

- Note 434351.1: Linux Kernel: The SLAB Allocator

Disk Tuning

- Note 175980.1: Tuning Disk Throughput Using hdparm in Linux
- The sar and iostat tools are your friends for identifying disk I/O contention issues.
- Also check for disk I/O issues with SAN tools (varies by vendor, such as EMC, Hitachi, etc.).

Unix is one of the most popular operating systems in production today for large mission-critical Oracle E-Business Suite systems. In general, Unix shares many of the same tools and design in common with its close cousin Linux as a multitasking and multithreaded enterprise operating system. Performance tuning of Unix is a complex and large subject also beyond the scope of this book. The following My Oracle Support (http://support.oracle.com) notes provide insight into usage of the tools and techniques for performance optimization of Oracle EBS on UNIX platforms.

Unix Performance Tuning

- Note 144638.1: Relationship Between Common Init.ora Parameters and Unix, Linux Kernel Parameters
- Verify that kernel parameters are set correctly and verify with the vendor platform.
- Assess performance with top, iostat, vmstat, and sar.
- OS Watcher tool
  - Note 301137.1: OS Watcher User Guide

**TIP**

Find the performance bottleneck with top and ps to drill down to the root cause.

- Solaris: use truss and DTRACE for tuning system calls and processes
  www.sun.com/bigadmin/features/articles/dtrace_truss.jsp
- HP-UX: use the same utility for tuning
  http://docs.hp.com/en/B2355-90692/sam.1M.html
- IBM AIX: SMIT (System Management Information Tool) is useful tool for performance tuning

**NOTE**

You may need root OS privileges to run these tools or elevated privileges.
Figure 24-3 shows an example of how to use the top command for Linux to view Oracle system performance.

The Linux and Unix top command is an excellent starting point to understand the current performance of a running Oracle EBS environment. You can then quickly drill down into storage and disk performance by using the iostat tool.

$ iostat
Linux 2.6.18-194.8.1.0.1.el5 (app01.ben.com)         09/19/2010
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          0.29    0.00    0.13    0.37    0.00   99.22
Device:            tps   Blk_read/s   Blk_wrtn/s  Blk_read   Blk_wrtn
sda               0.08         0.29         1.79  21019459  131483716
sda1              0.00         0.00         0.00      2496       28
sda2              0.08         0.29         1.79  21016659  131483688
sdb               0.00         0.00         0.00     23141     5072
sdb1              0.00         0.00         0.00     22253     5072
dm-0              0.23         0.29         1.79  21000250  131361616
dm-1              0.00         0.00         0.00    16120     122072

$ iostat -x
Linux 2.6.18-194.8.1.0.1.el5 (app01.ben.com)         09/19/2010
avg-cpu:  %user   %nice %system %iowait %steal  %idle
          0.29    0.00    0.13    0.37    0.00   99.22
Device:  rrqm/s  wrqm/s  r/s  w/s  rsec/s  wsec/s  avgrq-sz  avgqu-sz
         await  svctm  %util
sda    0.00  0.15  0.01  0.07  0.29  1.79  26.25  0.00
3.77   2.05  0.02
sda1   0.00  0.00  0.00  0.00  0.00  0.00  23.37  0.00
3.22   2.70  0.00
sda2   0.00  0.15  0.01  0.07  0.29  1.79  26.25  0.00
3.77   2.05  0.02

FIGURE 24-3. Using top for Linux performance and Oracle R12 EBS

Figure 24-3 shows an example of how to use the top command for Linux to view Oracle system performance.

The Linux and Unix top command is an excellent starting point to understand the current performance of a running Oracle EBS environment. You can then quickly drill down into storage and disk performance by using the iostat tool.
Virtualization and Cloud-Computing Considerations

Cloud computing and virtualization are the hot technology in vogue today with data centers looking to consolidate hardware and reduce operating costs (CAPEX/OPEX) for Oracle applications. However, while the benefits of the cloud are magnificent, they also come with additional challenges and risks to address in order to avoid pitfalls. These risks include the time and opportunity cost of training existing and new staff on how to implement and administer the new virtualized cloud based environments. In addition, substantial time is required to migrate legacy environments to the virtual data center. The two major players in the cloud marketplace for Oracle EBS are VMware and Oracle. VMware is the market leader in virtualization, with VMware vCenter and vSphere for virtualization of Oracle applications. Oracle has a hypervisor product called Oracle Virtual Machine (VM) which consists of the Oracle VM Manager and Oracle VM Server. VMware has the advantage over Oracle in that its vCenter and vSphere hypervisor products can run nearly any application, including Oracle, in a virtualized cloud environment on server hardware. Oracle VM is limited to running only Oracle applications at this time. For customers with mixed workloads and mixed-application environments—such as a business requirement to run additional applications like Microsoft Exchange and SAP on the same server environment with Oracle—we recommend that you implement VMware for your private–public cloud solution.

Listed here are some key considerations to keep in mind when virtualizing Oracle EBS.

- Virtual servers with VMware and Oracle VM require additional hardware resources to optimize performance compared to physical bare-metal implementations.
- Oracle and VMware have best practices you should be familiar with. For example, both offer DRS and clustering to optimize virtual-machine performance.
- VMware provides tools available with vCenter for provisioning, cloning, and deploying virtual machines as well as migrating physical servers to virtualized environments. Oracle provides similar features for these tasks with Oracle VM as well as template design with Oracle Virtual Assembly Builder.

Storage and Disk Considerations

- Solid-state disks offers the best overall performance, but are more expensive.
- Performance can be optimized with Disk I/O with RAID 0+1 or RAID 1+0 and Avoid RAID 5 for best overall disk performance.
- Automatic Storage Management should be implemented for Oracle 10g and Oracle 11g on the database tier for performance and availability.
- Usually vendor centric depends on storage vendor, e.g., EMC, SUN, HP, Hitachi, etc.

Areas to Tune—Start Here!

1. Oracle R12 EBS Technology Stack
   - Concurrent Manager Tuning
   - Oracle R12 EBS Module tuning: AR, AP, GL, HR, etc.
Application and Database Performance Tuning

Now that we have discussed methodologies and initial performance-tuning considerations, let’s delve into core tuning areas for the Oracle E-Business Suite. Application and database performance tuning is the heart of challenges for Oracle professionals tasked with optimizing performance for Oracle Financials applications.

Application Tech-Stack Tuning

The Oracle EBS technology stack is a robust and complex set of application servers embedded within the Oracle E-Business Suite. In contrast to a non-Oracle EBS system, it poses unique challenges to optimizing performance and availability. The Oracle EBS application tier consists of multiple application JVM containers, an HTTP Web server based on Apache called Oracle HTTP Server, and the Concurrent Managers. Figure 24-4 shows the architecture of the application technology stack.

Tuning Concurrent Managers

Best practices:

- Useful My Oracle Support Notes: 104452.1: Troubleshooting Concurrent Manager
- Note 1057802.1: Best Practices for Performance for Concurrent Managers in E-Business Suite
- Myth: More Concurrent Managers are better
- Keep it simple and use only a few Concurrent Managers
- Tune queue size
- Tune sleep cycle (PMON)
- Tune number of processes
- Check for high number of requests

TIP

Do not mistake the forest for the trees and avoid tunnel vision!
Use the following script to find high numbers of concurrent requests.

```
SELECT v.USER_CONCURRENT_QUEUE_NAME, COUNT(PHASE_CODE) v1
FROM  APPS.FND_CONCURRENT_QUEUES_V1 v,
     APPS.FND_CONCURRENT_WORKER_REQUESTS r
WHERE r.queue_application_id = 0
  AND r.PHASE_CODE = 'P'                -- Pending Concurrent Requests
  AND r.HOLD_FLAG != 'Y'                -- Concurrent Requests not on hold
  AND r.REQUESTED_START_DATE <= SYSDATE -- No Future Concurrent jobs
  AND r.CONCURRENT_QUEUE_ID=v.CONCURRENT_QUEUE_ID
```

**FIGURE 24-4.  Oracle R12 EBS tech-stack architecture**
Use Oracle Application Manager (OAM) charts to measure performance.

OAM is an excellent starting point to find performance bottlenecks for Concurrent Managers and other tuning issues with Oracle R12 EBS. Figure 24-5 shows the main tuning screen for examination of Oracle R12 EBS performance.

Figure 24-6 shows how to check for long-running concurrent requests for potential bottlenecks by using Oracle Application Manager.

Figure 24-6. Finding long-running concurrent requests with OAM
You can use OAM to measure performance of Concurrent Managers for throughput to get the big picture. Figure 24-7 shows how to use OAM to examine database and concurrent batch-processing performance.

**Tuning PMON Sleep Cycle**

**How to set value for PMON sleep cycle?**

- Start-up parameter in adcmctl.sh script for Oracle R12 EBS
  Located under $INST_TOP/admin/scripts/ directory in applications tier.
- Parameters
  adcmctl.sh {start|stop|abort|status} [<APPs username/APPs password>]
  [sleep=<seconds>] [restart=<N|minutes>] [pmon=<iterations>] [quesiz=<pmon_iterations>] [diag=Y|N] [wait=Y|N]
- Tune by modifying the pmon and sleep parameters.
- You can also use OAM to set these values.

---

**FIGURE 24-7.** Oracle R12 EBS performance analysis with OAM charts
Finding Value for PMON Sleep Cycle

Log in to OAM as SYSADMIN and systems-administrator Responsibility, then navigate to System Administrator | Concurrent:Manager | Define | WorkShifts.

Figure 24-8 shows how to locate the sleep value for the PMON process.

Cache Size and Performance for Concurrent Processing

Tune by setting in OAM for Oracle R12 environment.

The cache size can be tuned via the OAM menu as shown in Figure 24-9.

Tuning Number of Processes for Concurrent Managers

- Examine the number of processes for Concurrent Managers.
- Check the columns for actual, target, and running in OAM.
- Log in as SYSADMIN and then navigate the path Concurrent:Manager | Administer. See Figure 24-10.

Specialization Rules for Concurrent Manager Performance

- Use INCLUDE and EXCLUDE rules to optimize performance, since Concurrent Managers are batch-type processes.
- Log in as SYSADMIN and navigate System Administrator Concurrent:Manager | Define in OAM. See Figure 24-11.

The above menu item is located under Specialization Rules section in OAM.
Chapter 24: Performance Tuning for Oracle R12 Financials

**FIGURE 24-9.** Setting cache size for Oracle R12 EBS in OAM

**FIGURE 24-10.** Tuning the number of concurrent processes for Oracle R12 EBS in OAM
Setting Values for the Number of Processes with Concurrent Managers

- Located in OAM under the navigation path System Administrator | Concurrent:Manager | Define | WorkShifts. See Figure 24-12.
- Use combined specialization rules for complex tasks. See Figure 24-13.
Work Shifts for Performance with Concurrent Processing

- Set up different work shifts for balancing performance load with different critical tasks.
- For example, you can schedule AP to run during day and nightly processing on different work shifts.
- Log in to OAM as SYSADMIN and then navigate to System Administrator | Concurrent:Manager | WorkShifts. See Figure 24-14.
The following are some key scripts for monitoring the Concurrent Managers:

- `afcmsg.sql`: Shows all defined managers, capacity, process ID, etc.
- `afimchk.sql`: Gives status for Internal Concurrent Manager and PMON
- `afcmcreq.sql`: Shows Concurrent Manager basics
- `afrqwait.sql`: Shows pending, held, and scheduled requests
- `afreqstat.sql`: Lists concurrent-request execution times and status
- `afqpmrid.sql`: Identifies the process IDs for FNDLIBR processes
- `afimlock.sql`: Finds concurrency and locking issues

These scripts are located in the $FND_TOP/sql directory. You can execute them while connected to SQL*PLUS as the APPS schema.

Below are some additional scripts that can be used for tuning the Oracle EBS concurrent managers.

```sql
select
    f.application_short_name app,
    substr(p.user_concurrent_program_name,1,55) description,
    substr(p.concurrent_program_name,1,20) program,
    r.priority,
    count(*) cnt,
    sum(actual_completion_date - actual_start_date) * 24 elapsed,
    avg(actual_completion_date - actual_start_date) * 24 average,
    max(actual_completion_date - actual_start_date) * 24 max,
    min(actual_completion_date - actual_start_date) * 24 min,
    stddev(actual_completion_date - actual_start_date) * 24 stddev,
    stddev(actual_start_date - requested_start_date) * 24 wstddev,
    sum(actual_start_date - requested_start_date) * 24 waited,
    avg(actual_start_date - requested_start_date) * 24 avewait,
    c.request_class_name type
from fnd_concurrent_queues fcq,
     fnd_concurrent_queue_content fcqc,
     fnd_concurrent_request_class c,
     fnd_application f,
     fnd_concurrent_programs_vl p,
     fnd_concurrent_requests r
where r.program_application_id = p.application_id
and r.concurrent_program_id = p.concurrent_program_id
and r.status_code in ('C','G','E')
and p.application_id = f.application_id
and r.concurrent_request_class_id = c.request_class_id(+)
and r.request_class_application_id = c.application_id(+)
and fcqc.concurrent_queue_id = fcq.concurrent_queue_id(+)
and fcqc.type_application_id = fcq.application_id(+)
and fcqc.type_id = fcq.type_id(+)
```
```sql
SELECT c.request_class_name, 
f.application_short_name, 
p.concurrent_program_name, 
p.user_concurrent_program_name, 
r.priority 
FROM c_request_class c, f_package f, p_concurrent_program p, r_priority r 
WHERE c.request_class_id = f.concurrency_id 
AND f.package_short_name = p.package_name 
AND p.user_concurrent_program_name = r.user_concurrent_program_name 
AND r.priority = p.priority
```

**Statistics Collection for Oracle R12 EBS**

There are two fundamental reasons why you need to update statistics for Oracle EBS environments.

- Cost-based optimization requires the best execution path with current statistics.
- Second, it improves overall application and database performance because the execution plans are stabilized for queries issued against the database tier. Below is a reference point on how to collect statistics:

  My Oracle Support (http://support.oracle.com)  
  Note 419728.1 How to Gather Statistics on Oracle Applications 11.5.10 (and above)—Concurrent Process, Temp Tables, Manually, which provide details on how to collect stats for the Oracle EBS environment.

Do not use DBMS_STATS with Oracle R12 EBS to collect statistics! The reason why you want to use the FND_STATS or concurrent program to update and collect statistics is because of the integrated nature of ERP systems of which Oracle R12 EBS is a part. By using the FND_STATS method for collecting statistics for Oracle EBS, you will ensure that all of the application schemas receive the best execution plan. If you run DBMS_STATS it will only update the internal database stats such as those for SYS and SYSTEM and will fail to grab the latest optimizer statistics for application schemas such as APPS.

Be sure to gather table statistics on CM tables:

- FND_CONCURRENT_PROCESSES
- FND_CONCURRENT_PROGRAMS
The above Oracle EBS concurrent manager (CM) tables need to be updated frequently in terms of gathering statistics to maintain performance. Below are the two key methods to gather statistics for Oracle R12 EBS:

- Manual
- Concurrent job for gather stats

FND_STATS for schema statistics collection

**Example**

Collect statistics for AP schema:
```
exec fnd_stats.gather_schema_statistics('AP');
```

Collect stats for all schemas for Oracle R12 EBS:
```
exec fnd_stats.gather_schema_statistics('ALL');
```

How to gather stats for temp tables with Oracle R12 EBS:
```
exec fnd_stats.gather_table_stats('<schema>','<temp_table_name>');
```

Collect stats for temp tables in AR schema:
```
exec fnd_stats.gather_table_stats('AR','temp_ap');
```

Using OAM to gather stats for Oracle R12 EBS.
```
Use the following navigation path within the Oracle R12 EBS menu from OAM: System Administrator | Concurrent Requests |
```

See Figure 24-15.

**Application Tech-Stack Tuning**

- Tune values for timeouts to prevent and resolve errors in performance for network delays for forms, oacore, and OC4J.
- Avoid and resolve the infamous “Uninterrupted Exception 150 error” when starting and stopping application-tier services.
- Values to check in Oracle R12 EBS Content file
- Consider shared APPL_TOP and staged APPL_TOP for performance.
- Implement Parallel Concurrent Processing to distribute the performance load for concurrent processing.
- Configuration settings for JVM with OC4J under both the Oracle R12 EBS Context file for application tech stack and opmn.xml values.
Additional Considerations for Tuning the Application Tier with Oracle R12 EBS

- Performance metrics for OC4J groups
- Use server for concurrent processing server, application server in split-tier configuration to provide more performance and availability.
- Consider load-balancing hardware or software to improve performance and availability.
- Consider deploying Oracle RAC for scale-out and scale-up in performance if you need it for large deployments.
- Oracle 10g Application Server tuning issues
- Oracle HTTP Server: Apache Web server
- Timeout settings for Apache
- Under http.conf file under IAS_ORACLE_HOME and Oracle 10g AS home directory
Load Balancing for Application-Tier performance

- Hardware load-balancing options: Cisco and Big IP some vendors that offer solutions
- Software load balancing
- Load balancing is complex and requires coordination with network, security, and systems-administration teams, as well as the database team, for successful implementation.
- Reference point:
  - My Oracle Support Note 727171.1: Implementing Load Balancing on Oracle E-Business Suite - Documentation for Specific Load Balancer Hardware

Software Load Balancing for Oracle R12 EBS Application-Tier Performance

- Not as robust as hardware load balancers but more cost effective
- Uses Web cache option for Oracle 10g AS application server
- My Oracle Support Note 380486.1 Installing and Configuring Web Cache 10g and Oracle E-Business Suite 12

Java Tuning for Oracle R12 E-Business Suite

- Tuning Java components is essential for performance with Oracle R12 EBS.
- Oracle R12 EBS is dependent on Java technology.
- Default values are too low for Java with OC4J settings.
- Watch for issues with garbage collection.
- Avoid full garbage collection, as this impacts performance.
- My Oracle Support Note 567551.1: Troubleshooting: Configuring Various JVM Tuning Parameters for Oracle E-Business Suite 11i and R12
- My Oracle Support Note 362851.1: Guidelines to Set Up the JVM in Apps E-Business Suite 11i and R12
- http://java.sun.com/javase/technologies/hotspot/vmoptions.jsp#BehavioralOptions

Tools for Measuring JVM Performance with Oracle R12 EBS

- Excellent free tool to graph performance for JVM with Oracle R12 EBS and garbage collection
- www.tagtraum.com/gcviewer.html

See Figure 25-16.
Chapter 24: Performance Tuning for Oracle R12 Financials

Monitoring Java Performance: Oracle R12 EBS

- Oracle 10g Grid Control (OEM) with Application Management Pack (AMP)

NOTE
These items require additional licenses.

- My Oracle Support Note 557194.1 provides a script called monitor_jdbc_conn.sql to monitor JDBC connections.
- Collect metrics to analyze current performance
- Good information from ADDM and AWR in 11g. Below is the output from an AWR report from the database tier for Oracle R12 EBS:

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Total</th>
<th>per Second</th>
<th>per Trans</th>
</tr>
</thead>
<tbody>
<tr>
<td>java call heap collected bytes</td>
<td>683,300,160</td>
<td>1,300.0</td>
<td>877.8</td>
</tr>
<tr>
<td>java call heap collected count</td>
<td>3,693,358</td>
<td>7.0</td>
<td>4.7</td>
</tr>
</tbody>
</table>
Database Tuning for Oracle R12 E-Business Suite

- Similar to tuning regular Oracle 11g database, but unique in how statistics collected
- Remember to use FND_STATS rather than DBMS_STATS to collect and update statistics.
- Watch out for concurrency and locking issues for runaway user processes.
- Schedule backup jobs during off hours to avoid impact on business users.
- Suggested values for database initialization parameters for Oracle R12 EBS
- My Oracle Support Note 396009.1: Database Initialization Parameters for Oracle Applications Release 12
- Oracle R12 by default installs either a 10gR2 or an 11gR1 database for each fresh (new) installation.
- For example: Oracle 12.0 to 12.04 uses 10.2.x database
- Oracle 12.1.1 and later install Oracle 11gR1 database (11.1.x).
- Watch out for upgrade issues from 11i to R12 EBS for database.
- Suggested values for database initialization parameters for Oracle R12 EBS

Database Parameters for Oracle R12 EBS Sizing Chart
Table 24-1 shows the recommended values for the Oracle database-initialization parameters to use with Oracle R12 E-Business Suite.

Upgrade Performance Considerations for Oracle R12 EBS

- When you upgrade from older releases such as 10.x and 11i to Oracle R12, you must account for performance items.
- Rule based Optimizer (RBO) in 10.x and early 11i releases to the Cost Based Optimizer (CBO) in R12
- Different way of tuning for stats and database
Consider changes for SGA and PGA sizing.
- Migrate from old tablespace model to Oracle Application Tablespace Model (OATM)
- My Oracle Support Note 761570.1: Database Preparation Guidelines for an E-Business Suite Release 12.1.1 Upgrade

### Conclusion

Oracle performance tuning is a massive discipline that crosses multiple business and technology areas. Tuning begins with a holistic approach to correctly implementing key business processes that map to the technical aspects of hardware, storage, network, and application processes, so that performance analysis is conducted in a proactive rather than a reactive manner.