

Oracle Hyperion Financial Data Quality Management, Enterprise Edition

TUNING GUIDELINES | OCTOBER 2014





Disclaimer

This document is provided for information purposes and should not be relied upon in making a purchasing decision. 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.

THIS DOCUMENT IS NOT PART OF A LICENSE AGREEMENT NOR CAN IT BE INCORPORATED INTO ANY CONTRACTUAL AGREEMENT WITH ORACLE CORPORATION OR ITS SUBSIDIARIES OR AFFILIATES.

Failure to adhere to these benchmarks does not constitute a breach of Oracle's obligations. 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.

| | |
|---|-------------------------------------|
| Table of Contents | |
| Disclaimer | 1 |
| Table of Contents | 2 |
| Introduction | 3 |
| Monitoring Tools | 4 |
| Windows Performance Monitor | 4 |
| Oracle Enterprise Manager Fusion Middleware Control | 4 |
| Tuning Parameters | 6 |
| VM Heap Registry Settings | 6 |
| Changing the Default JVM Heap Registry Entries | 6 |
| Weblogic Data Sources | 7 |
| Changing the default number of data sources | 7 |
| Batch Size | 9 |
| Changing the Default Batch Size | 9 |
| Oracle Data Integrator (ODI) | 9 |
| Database Indexes | 10 |
| Determining length of a process from the ProcessID logs | 11 |
| SQL Deadlocks | 12 |
| Load Balancing | Error! Bookmark not defined. |
| Checklist for a tuning check-up | 13 |
| Conclusion | 13 |

Introduction

Finance organizations continually enhance the quality of their internal controls and reporting processes. To meet these goals, a source-to-report view of financial data processes is required. Oracle Hyperion Financial Data Quality Management, Enterprise Edition (FDMEE) is a solution that allows business analysts to develop standardized financial data management processes and validate data from any source system—all while reducing costs and complexity. FDMEE puts the finance user in total control of the integration process to define source data, create mapping rules to translate data into the required target format, and to execute and manage the periodic data loading process.

Performance challenges can be a real showstopper during critical times of the year. This document focuses on general tuning guidelines for FDMEE, to help you avoid some of those challenges before they happen. It is not release specific, but generally applies to the 11.1.2.x releases. Your hardware and software configurations may result in outcomes other than what is described here. Management of data volumes, mapping rules, load balancing and database tuning may also be required to achieve the desired performance outcomes.

Users of this guide should be very familiar with Oracle Hyperion Financial Data Quality Management, database administration, and general operating system concepts. Before implementing any tuning changes, collect performance data on the default configuration, make incremental changes, and compare the performance data after the change.

Caution: Editing the Windows registry incorrectly can severely damage your system. Before making changes to the Windows registry, have a complete system backup available.

An assumption is made that all the latest patches have been applied to all the servers in your configuration.

Before making changes to your environment, review the following tuning, performance or sizing guidelines for any of the products installed in your configuration.

Table 1 Supporting documentation that you may need

| Task | Related Documentation |
|--------------------|---|
| HFM tuning | <i>Oracle® Hyperion Financial Management, Fusion Edition – Performance Tuning Guide</i> |
| Essbase ASO tuning | Tuning of Aggregate Storage Databases |
| General Tuning | <i>Oracle® Enterprise Performance Management System Installation – Start Here</i> |
| Load Balancing | <i>Oracle® Enterprise Performance Management System Deployment Options Guide</i> |

Monitoring Tools

Windows Performance Monitor

The recommended tool to monitor Windows performance is “Microsoft Windows Performance Monitor.” The Windows performance counters you should record are:

- » % Processor time
- » Private Bytes
- » Virtual Bytes

For the “Instances of selected object”, select the following:

- » HyS9aifWeb_epmsystemx
- » odi64

The data file produced by Windows Performance Monitor can then be analyzed for any resource shortages. To get more help on using Windows Performance Monitor, from the menu select **Help**, and then select **Help on Performance Monitor**.

For Linux, there are many options for monitoring/gathering system and process statistics including: dstat, vmstat and iostat to name a few. Consult your Linux documentation for help on this topic.

Oracle Enterprise Manager Fusion Middleware Control

To assist troubleshooting issues with JVM heap memory and data sources, the Oracle Enterprise Manager Fusion Middleware Control is available from your workspace server.

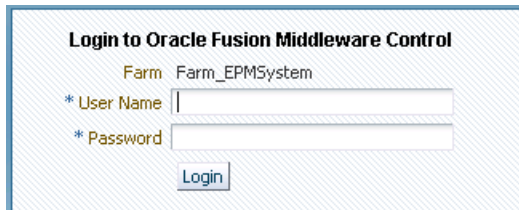
To access the Oracle Enterprise Manager Fusion Middleware Control:

- » From a browser, type:

http://servername:7001/em

Replace the “*servername*” with your WebLogic server *name* or *IP address*.

1. From the **Login to Oracle Fusion Middleware Control** screen, enter the WebLogic administrator **user name** and **password**.

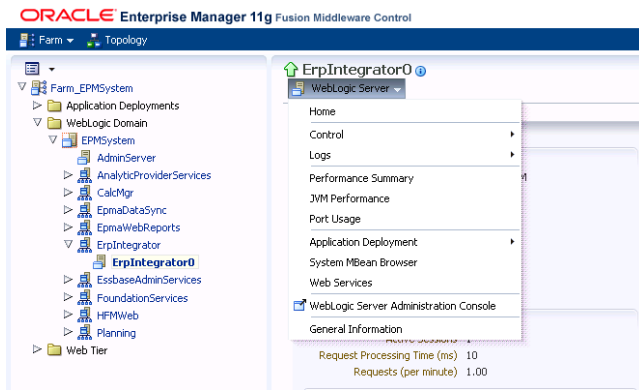


The screenshot shows a login form with the following elements:

- Title: Login to Oracle Fusion Middleware Control
- Farm: Farm_EPMSsystem
- * User Name: [input field]
- * Password: [input field]
- Login button

2. Click **Login**.

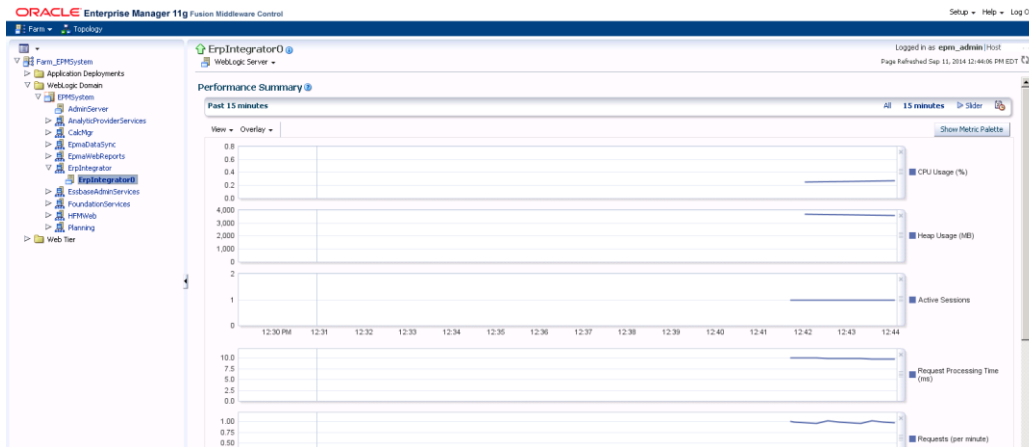
3. Navigate to ErpIntegrator from the menus on the left, expand an item, and click the right arrow.



4. Click **WebLogic Server**, and then select **Performance Summary**.

You can select the resource graphs to display by clicking **Show Metric Palette** button, and selecting the metrics by clicking the appropriate check boxes.

Monitoring the Heap Usage and data sources, in real time can identify any potential bottleneck during peak usage times.



Another option for monitoring your JVM heap usage is Oracle® JRockit Mission Control. Oracle JRockit Mission Control allows you to monitor the heap usage live through a visual console/dashboard.

For more information on JRockit Mission Control, reference either the Oracle Fusion Middleware Documentation Library:

- » http://docs.oracle.com/cd/E15289_01/index.htm
- or the Oracle Technology Network:
- » <http://www.oracle.com/technetwork/middleware/jrockit/mission-control/index.html>

Tuning Parameters

VM Heap Registry Settings

Heap Memory Error (OutOfMemoryError)

The JVM (Java Virtual Machine) allocates and manages the heap memory based on how much each Java application needs. When a Java application is running, the systematic usage and release of heap memory is managed through a “garbage collector”.

The two settings to change for heap memory are:

- » -Xms - the minimum size of the heap
- » -Xmx - the maximum size of the heap

If your application runs out of heap memory, you might see the log file message: “OutOfMemoryError”. For FDMEE we recommend setting the minimum and the maximum heap sizes to the same value. This prevents the JVM from having to allocate more memory every time it is needed, which affects performance. Finding the optimal heap settings for your application will require internal testing using various values.

To set heap sizes properly, consider how much total memory the server has (RAM), and what other processes/applications are running on that server. When you have the total memory information, collect the information on how much memory the processes/applications are currently using. As a general guideline, the memory being used by all applications should not be more than 80% of the Total RAM. Otherwise performance for all applications on the server may start to degrade.

To find out how much memory (RAM) the server has:

- » For Linux, use the “free” command this will show you Total, Used, and “Free” memory.
- » For Windows, the Windows Task Manger shows the Total memory, Available, and Free memory.

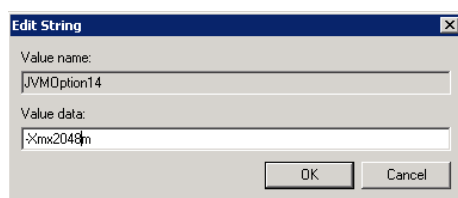
To open the Task Manager, right click on the server task bar and select **Start > Task Manager**, then select the **Performance** tab. You can increase these settings in increments of 512m or 1024m at your discretion. The change to these settings does not take effect until the “HyS9aifWeb” service is restarted.

Changing the Default JVM Heap Registry Entries

To change the JVM heap registry entries for Windows:

1. Open the Windows Registry.
2. Navigate to: HKEY_LOCAL_MACHINE\SOFTWARE\Hyperion Solutions\ErpIntegrator(n)\HyS9aifWeb_epmsystem(n)
Where ErpIntegrator(n) is something like ErpIntegrator0, ErpIntegrator1, depending on how many FDMEE web servers are deployed and which server you are updating.
3. Locate the two JVMOptions –Xms and –Xmx (Data Type is REG_SZ).
4. Double click the option to change.
(For example, JVMOptionnn)
5. In the Value Data, change the numerical value of –Xms512m to the new value,

Repeat this step for the –Xmx value.



6. Restart the **HyS9aifWeb** service.

-Xms and -Xmx values should be set to the same number

To change the JVM heap entries for Linux do the following:

1. Edit this file: <MIDDLEWARE_HOME>/user_projects/epmsystemx/bin/deploymentScripts/setCustomParamsErpIntegrator.sh.
2. Find the section `JAVA_OPTS="-Xms512m -Xmx512m..."`
3. Change the numerical values to the new desired heap memory value, save the changes.
4. Restart the "Oracle Hyperion FDM Enterprise Edition - Java Web App" service.

-Xms and -Xmx values should be set to the same number

WebLogic Data Sources

WebLogic Data source Error (weblogic.common.resourcepool.ResourceLimitException:)

WebLogic data sources are the number of pooled connections made available to FDMEE for a JDBC (database) data source. The application uses a connection, and then returns it to the pool when finished with the connection. If your application runs out of WebLogic data sources you might see in your log file, "ResourceLimitException", or "Reached maximum capacity of pool".

Changing the default number of data sources

To change this setting, use the WebLogic Server Administration Console. For FDMEE, "aif_datasource", and "odiMasterRepository" are the data source names.

To start Oracle WebLogic Server Administration Console:

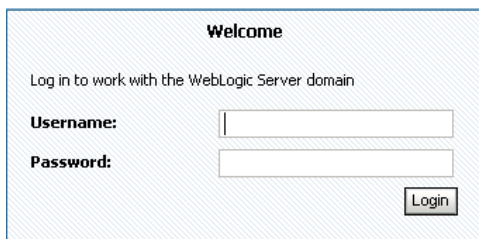
1. Select **Start**, then **Programs**, then **Oracle WebLogic**, then **User Projects**, then **EPMSystem**, and then **Start Admin Server for WebLogic Server Domain**.

When the Admin Server displays "RUNNING", select **Start**, then **Programs**, then **Oracle WebLogic**, then **User Projects**, then **EPMSystem**, and then **Admin Server Console**.

To access the WebLogic Server Administration Console from the browser type in the following URL:

- » `http://servername:7001/console`

Replace "servername" with your WebLogic server name or IP address. Your port number may be different, 7001 is the default port number.



2. Login to WebLogic Server Administration Console with the credentials used during the system configuration. (the default username is epm_admin)
3. From **Domain Structure**, select **Services**, and then **Data Sources**.



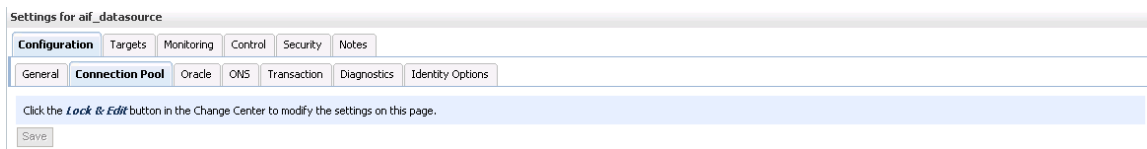
4. Select either **aif_datasource** or **odiMasterRepository** by selecting the hyperlink in the **Name** column.

Data Sources (Filtered - More Columns Exist)
Click the *Lock & Edit* button in the Change Center to activate all the buttons on this page.

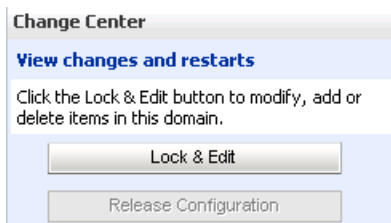
New Delete Showing 1 to 8 of 8 Previous | Next

| <input type="checkbox"/> | Name | Type | JNDI Name | Targets |
|--------------------------|-------------------------------------|---------|--------------------------|---------------|
| <input type="checkbox"/> | aif_datasource | Generic | jdbc/aif_datasource | ErpIntegrator |
| <input type="checkbox"/> | odiMasterRepository | Generic | jdbc/odiMasterRepository | ErpIntegrator |

5. Select the **Configuration** tab and then the **Connection Pool** tab.



6. Click **Lock & Edit** button to modify the “Maximum Capacity” setting.



7. In **Maximum Capacity**, type the new value for the data source connections.

| | | |
|--------------------------|----------------------------------|--|
| Initial Capacity: | <input type="text" value="1"/> | The number of physical connections to create when creating the connection pool in the data source. If unable to create this number of connections, creation of the data source will fail. More Info... |
| Maximum Capacity: | <input type="text" value="800"/> | The maximum number of physical connections that this connection pool can contain. More Info... |
| Minimum Capacity: | <input type="text" value="1"/> | The minimum number of physical connections that this connection pool can contain after it is initialized. More Info... |

8. Once the setting is changed, click **Save**. Repeat steps four through eight to change the `odiMasterRepository` data source connection pool.
Changing this setting does not require a restart of the `FDMEE` service.

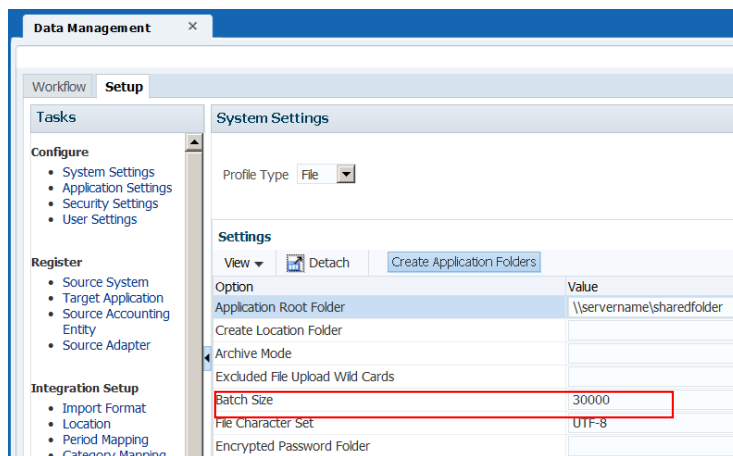
Batch Size

Batch Size refers to the number of import records that are “batched up” before a commit occurs on the database. A setting too large or too small can have adverse performance effects.

For example, if you import a large number of records, such as $\geq 500,000$ records, a Batch Size setting of 30000 could alleviate a bottleneck. Smaller imports, such as $< 10,000$ records, would suffice to use the default batch size. The “Batch Size” parameter is set at the system level. The default setting for “Batch Size” is 1000.

Changing the Default Batch Size

This setting is changed by selecting **Data Management**, then on the **Setup** tab, select **Configure, System Settings**. On the **System Settings** screen, type the new value in the **Batch Size** field and click **Save**.



Oracle Data Integrator (ODI)

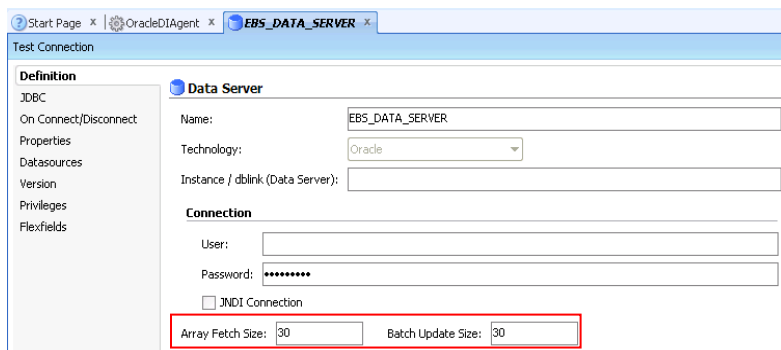
ODI is the engine that extracts data and metadata from the defined sources. It also populates the tables in `FDMEE` or populates the target applications with the extracted artifacts.

ODI has three significant tuning components to explore if you are having problems loading large data sets. The “Array Fetch Size”, “Batch Update Size”, and the “Maximum number of sessions” can be found on the “Topology” tab of ODI Studio in the Physical Architecture section for your data server. The Maximum number of sessions setting is found under the agent.

The “Array Fetch Size” value indicates how many rows are retrieved and stored in memory when doing a “Fetch” using the database driver. When you set this value too high and you are typically retrieving a small number of rows, adverse affects might result. Likewise if you specify a value too small when retrieving thousands of rows, multiple trips to service the request results.

The “Batch Update Size” value indicates the number of rows to insert/update when writing to the database, before a commit is done. For Example, If you are retrieving 500,000-1,000,000 rows from the general ledger, it would not be uncommon to set the Array Fetch Size to 30000, and set the Batch Update Size to 300.

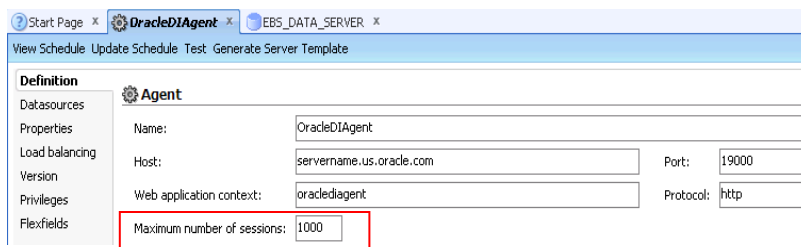
The default values for Array Fetch Size and Batch Update Size is “30”.



The “Maximum number of sessions” value depends on the capabilities of the machine running the agent.

Oracle Data Integrator allows you to load balance parallel session execution between physical agents. Each agent has a maximum number of sessions it can execute simultaneously from a work repository.

The default value for Maximum number of sessions is 1000.



Database Indexes

In some cases new database indexes are required for optimal performance. This is entirely dependent on your specific application mappings. To monitor the performance of processes use the FDMEE Process Log files, with log level set to 4 or 5. Your DBA can help you analyze the information from the logs and create the necessary indexes.

Database indexing can be a complicated topic, and this paper covers the key points relevant to this discussion. Indexes can increase the speed of execution. However, adding too many indexes can make insert performance slower.

To create indexes consider the level of permission to the table(s) for which you want to create indexes, and the amount of disk space required to store the index.

In FDMEE the import process is comprised of two steps: importing data from the source and then mapping the source data to the target. The area most affected by performance is the mapping step of the import process. The number and types of mappings could

adversely impact your performance. To determine the most beneficial indexes, run an import process that represents a typical load. Your DBA can examine the Process Log files to determine the dimension that is taking the most time and create indexes on the dimension column.

Process Log file setting – A Log Level of 4 provides summary information of the time taken for each dimension and each map. In some cases this information may not be enough to evaluate the indexing required. In that case Log Level 5 can be used. This provides a complete SQL statement for each mapping execution.

The Process Log Level is set on the “Setup” tab within “Data Management”, under “System Settings”. If you select Profile Type “Other” from the drop down you will see the setting for “Log Level”. Change the log level value and click the Save button to save your changes.

| | |
|-----------|---|
| Log Level | 5 |
|-----------|---|

This is an example of the syntax to add an index to your table.

- » CREATE INDEX *indexname* ON *tablename* (UD1);
 - » Example: CREATE INDEX TDATASEG_T_C1 on TDATASEG_T(UD1); (, *tablename* is usually TDATASEG_T)

Determining length of a process from the ProcessID logs

In your FDMEE Application root folder, within the Outbox > Logs folder, the ProcessID log files are written for each execution. The file name has a format that resembles the following: “*targetappname_processidnumber.log*”.

With log level set to 4 or 5, the start and end step indicators are available for determining the amount of time taken for the various steps in the process execution.

Examples of the entries in the log file are shown here:

Based on the Version of EPM you have the message may vary slightly

```
YYYY-MM-DD 04:22:09,305 INFO [AIF]: FDMEE-105011:Message - Total time taken for the import in ms = 2347
YYYY-MM-DD 12:55:46,851 INFO [AIF]: ERPI-105011:EPMERPI- Total time taken for the import in ms = 14863
YYYY-MM-DD 04:22:36,744 INFO [AIF]: -----START IMPORT STEP-----
YYYY-MM-DD 04:22:50,307 INFO [AIF]: -----END IMPORT STEP-----
YYYY-MM-DD 04:22:50,331 INFO [AIF]: -----START VALIDATE STEP-----
YYYY-MM-DD 04:22:50,477 INFO [AIF]: -----END VALIDATE STEP-----
YYYY-MM-DD 04:22:50,542 INFO [AIF]: -----START EXPORT STEP-----
YYYY-MM-DD 04:22:52,159 INFO [AIF]: -----END EXPORT STEP-----
YYYY-MM-DD 04:22:52,323 INFO [AIF]: ***** BEGIN CONSOLIDATION STEP INIT *****
YYYY-MM-DD 04:22:52,350 INFO [AIF]: ***** END CONSOLIDATION STEP INIT *****
YYYY-MM-DD 04:22:52,396 INFO [AIF]: -----START CHECK STEP-----
YYYY-MM-DD 04:22:52,456 INFO [AIF]: -----END CHECK STEP-----
```



SQL Deadlocks

Creation of indexes for key tables may also be necessary to avoid SQL deadlock issues when using a Microsoft SQL Server database. If you are experiencing a deadlock issue, you might receive an error resembling the following:

“Transaction (Process ID xxx) was deadlocked on lock resources with another process and has been chosen as the deadlock victim.”

The recommended setting for Transaction Isolation Level is Read Committed (Snapshot). The default for Microsoft SQL Server READ_COMMITTED_SNAPSHOT is set to OFF.


Consult the most current FDMEE patch documentation for any index recommendations to help correct this issue.

Scaling to Increase Capacity

In cases of large application deployments with many users and high user load, consider scaling your application server.

The following documentation describes the steps involved in scaling of your application.

» [Oracle® Enterprise Performance Management System Deployment Options Guide](#)



Checklist for a tuning check-up

If any of the following changes, consider re-tuning your system for optimal performance:

- ✓ Change in number of mappings
- ✓ Change in custom scripts (import, mapping, etc.)
- ✓ Change in number of end-users
- ✓ Change in Hardware
- ✓ Change in number of applications and their structure
- ✓ Change in major software release

Conclusion

Tuning is an important activity that needs to be addressed throughout the lifecycle of your applications. It should be considered after a configuration is installed and be re-evaluated over time. Evaluation of changes such as an increase in end-users, additional applications, hardware changes, just to name a few, could impact the performance of your system and require re-tuning. Implementing end-to-end monitoring for mission critical applications reduces the risk of bottlenecks and downtime during critical business cycles. Of course there are a host of other situations that degrade the performance of a configuration. A sample list of other things to consider is:

- Network latency
- Disk Space
- Fragmentation
- DNS and Firewalls

For more help, contact Oracle Support: <http://www.oracle.com/us/support/contact/index.html>




Or the Oracle Technology Network: <http://www.oracle.com/technetwork/index.html>



Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

-  blogs.oracle.com/oracle
-  facebook.com/oracle
-  twitter.com/oracle
-  oracle.com

Hardware and Software, Engineered to Work Together

Copyright © 2014, Oracle and/or its affiliates. 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 and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. 1014

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