



LECTRA FASHION PLM

Database Performance guide

Target: ORACLE on Windows

Date of last update: November 2014

Edition: 3.2

Reference: 521658

PREFACE

Lectra PLM Fashion: Copyright ©2006-2014, Lectra and third parties. All rights reserved. This software is owned by Lectra and Lectra's applicable suppliers, and is protected by intellectual property and copyright laws. All rights, title and interest in and to this software, including without limitation all copyrights, patents, trademarks and trade secrets remain at all times exclusively with Lectra, or its applicable suppliers. Clarity: Copyright©1998-2005 Niku Corporation and third parties. This software is owned by Niku and its suppliers and is protected by United States copyright laws and international treaty provisions.

Trademarks

Niku and the Niku logo are registered trademarks, and Clarity, the Clarity logo, Clarity Studio, Global 2000 Proven, Precision Security, PowerMods, Best Practice Accelerators, PMO Accelerator, CPIC Accelerator, Service Connect, Schedule Connect, and XOG are trademarks of Niku Corporation in the United States and certain other countries. All other trademarks, trade names, and/or product names are used solely for the purpose of identification and are the property of their respective owners.

Lectra® and Lectra Systèmes® are registered trademarks of Lectra. Internet Explorer is a registered trademark of Microsoft Corporation.

Windows NT®, Windows® 2000, and Windows® XP are registered trademarks of Microsoft Corporation. Microsoft® is a registered trademark of Microsoft Corporation. Any integrated Actuate product remains the property of Actuate.

License

The software is for limited use only. The software is subject to a limited, non-exclusive and non-transferable license of use, for the licensee's own internal business purposes only. The conditions and restrictions of such license are described in Lectra's end-user license of use.

Guarantees

Lectra reserves the right to modify information relating to its products etc., without prior notification, with the aim of improving their reliability and operation.

Publication does not imply that this information is free of all intellectual copyright and does not grant any license over these rights. Furthermore, Lectra shall not be held liable for any consequences arising from the use of this information, for whatever purpose.

The performance measurements and other data referred to in this documentation are approximate and have no contractual value.

Contacts

Europe Call Center: [mailto: callcenter-europe@lectra.com](mailto:callcenter-europe@lectra.com)

North America Call Center: [mailto: Callcenter.Americas@lectra.com](mailto:Callcenter.Americas@lectra.com)

Asia Pacific Call Center: [mailto: callcenter.asia.pacific@lectra.com](mailto:callcenter.asia.pacific@lectra.com)

Italy Call Center: [mailto: callcenter.italia@lectra.com](mailto:callcenter.italia@lectra.com)

Spain Call Center: [mailto: callcenter.sp@lectra.com](mailto:callcenter.sp@lectra.com)

Contents

Trademarks	2
License	2
Guarantees.....	2
Contacts	2
1. GENERAL OVERVIEW	10
1.1 Scope of the Document.....	10
1.1.1 Explain minimal best practice to get scalability on Windows in a PLM context.....	10
1.1.2 Explain what to do if a scalability problem occurs	10
1.1.3 PLM scope: all PLM versions	10
1.2 Intended Audience	10
1.3 Clarification of responsibilities between Lectra and Customers	10
1.4 Clarification of Lectra support team organization.....	11
1.5 Related Documentation.....	11
1.6 Organization	12
1.7 Conventions	12
1.7.1 Naming conventions:	13
2. COMMON REQUIRMENTS FOR ALL STEPS	13
2.1 Recommended to get the most recent database bundle	13
2.2 Lectra database bundle provided scripts to save time to configure OS.....	13
3. OPTIMIZATION TASKS BEFORE INSTALLING PLM DATABASE.....	14
3.1 Understand what is Scalability	14
3.2 Need to work on capacity planning	14
3.2.1 Definition	14
3.2.2 Recommended steps.....	14
3.2.2.1 Step 1: Determining Performance Goals and Objectives.....	14
3.2.2.2 Step 2: Measuring Your Performance Metrics.....	15
3.2.2.3 Step 3: Identifying Bottlenecks in Your System.....	15
3.2.2.4 Step 4: Implementing a Capacity Management Plan	15
3.2.3 Lectra recommendations on database instance memory.....	16
3.2.3.1 PLM old version (version <v3)	16
3.2.3.2 PLM Versions >= v3	16
3.3 CHECK OPERATING SYSTEM CONFIGURATION (WINDOWS).....	17
3.3.1 Most of the configuration task can be done using database bundle script.....	17
3.3.2 Memory and processor usage parameters.....	17
3.3.2.1 Adjusting Paging File Size	18
3.3.3 Windows 32 bits only: set memory /3GB and /PAE	18

3.3.3.1	Objective: avoid out of memory errors (ORA-4030)	18
3.3.3.2	Example on XP/Windows2003 server 32 bits.....	19
3.3.4	Adjusting network configuration (Windows 2003 Server).....	19
3.3.5	Storage: About raid sub-system	20
3.3.6	Storage: use SSD when intensive IO	20
3.3.6.1	PLM v2r3 32 bits example: not enough memory can generate intensive TEMP IO 20	
3.3.7	Storage: check defragmentation.....	21
3.3.7.1	Steps to defragment databases.....	21
3.3.7.2	Antivirus configuration	21
4.	POST-INSTALLATION OPTIMIZATION TASKS	22
4.1	Change the memory target from default to recommended value.....	22
4.1.1	Adjust Oracle instance memory (Refer to capacity planning chapter)	22
4.2	Summary of recommended regular tasks to automates	22
4.2.1	Check statistics are run at regular interval in an optimal period of time.....	22
4.2.1.1	Checking using sqlplus	22
4.2.1.2	Checking schedule using database console	22
4.2.1.3	Checking using database bundle STD_DB_tuning script.....	23
4.2.1.4	Apply statistics, if not applied, using database bundle script	23
5.	SOLVE PERFORMANCE PART1: COLLECT GENERAL FUNCTIONNAL CUSTOMER CONTEXT..	24
5.1	Describe the Customer context.....	24
5.1.1	What is the environment type? (PROD/UAT/DEV/TRAINING?)	24
5.1.2	Collect PLM schema information like functional version (ex: v3r3sp3)	24
5.1.3	PLM V3: Collect WLP schema information that can have impact on performance.....	24
6.	SOLVE PERFORMANCE PART2: COLLECT RELEVANT FUNCTIONNAL INFORMATION.....	24
6.1	Objective: describe the loss of service in a standard way	24
6.2	Get all relevant functional information.....	24
6.2.1	What is the frequency and time interval of the error/loss of service?.....	24
6.2.1.1	Not specified or unknown=> try to get the information	24
6.2.1.2	Consistently (need timestamp or time interval)	24
6.2.1.3	Intermittent (need timestamp or time interval)	24
6.2.1.4	One-time (need timestamp or time interval)	24
6.3	Explain the gravity of the problem.....	25
6.3.1	Are you experiencing a mission or business critical loss of service?	25
6.3.1.1	Requires immediate and continuous effort to solve? Y/N	25
6.3.1.2	Easy to recover, bypass or workaround? Y/N	25

6.3.1.3	Does the application work normally after the problem occurs? Y/N.....	25
6.3.2	Describe the functional impact:	25
6.3.2.1	Only on one user, on site, on team?.....	25
6.3.2.2	What functional part is impacted?.....	25
6.3.2.3	Describe the scenario to reproduce.....	25
6.4	Explain the last application usage or configuration changes	25
6.4.1	Recent upgrades?	25
6.4.2	Big data load? (ETL....)	25
6.4.3	New customer development (CBR, reports,ETL...) deployed.....	25
6.4.4	New application usage part, new users.....	25
6.4.5	Application configuration changes (security...)	25
7.	SOLVE PERFORMANCE PART3: FIND RELEVANT TECHNICAL INFORMATION	26
7.1	Objective	26
7.2	Get logs	26
7.2.1	Objective: to be able to understand what part of the architecture has a problem.....	26
7.2.2	Get application server logs	26
7.2.2.1	Get persistence.log (not empty when problems comes from database tier)	26
7.2.2.2	Get apache logs (needed to understand application usage).....	26
7.2.3	Get database minimum server logs.....	26
7.2.3.1	Get and compress database alert file and last month traces files.....	26
7.2.3.2	Get and compress listener.log extract (to check connections problems).....	26
7.3	Get operating system and database global vision	26
7.3.1	Using database bundle get STD_DB_TUNING result.....	26
7.3.2	Using database console (if available) find interval with loss of services	27
7.3.2.1	Connect to the database console	27
7.3.3	Go to the performance dashboard and find interval with resource usage.....	27
7.3.3.1	Check CPU and IO resource used and note the time interval.....	27
7.3.3.1	Check top activity.....	27
7.3.4	Using database bundle get snapshots selecting loss of service time interval.....	27
7.3.4.1	Check if customer has enabled control_management_pack_access	27
7.3.4.2	Temporary enable control_management_pack_access if not yet enabled	28
7.3.4.3	Do a database snapshot and get the html report result for each snapshot.....	28
7.3.4.4	Once done disable control_management_pack_access if initially disabled.....	29
7.4	Check the last technical changes.....	29
7.4.1	Check each OS/database/application server/database configuration changes	29
7.4.2	Check if a new tier process is impacting resource usage	29

8.	SOLVE PERFORMANCE PART4: CENTRALIZE ALL INFORMATION TO ANALYZE	29
8.1	Prepare a centralized compressed folder with all relevant information to analyze	29
8.1.1	Objectives:	29
8.1.1.1	Centralize in a standard way all relevant information to diagnose a customer	29
8.1.1.2	Be able when too complex to analyze to send to support team	29
8.1.2	Suggested check list and naming rules	29
8.1.2.1	Root folder: Audit_<customer>_<environment type>_<timestamp>	29
8.1.2.2	Mandatory items	29
8.1.2.3	Not mandatory except to reproduce	30
9.	SOLVE PERFORMANCE PART5: METHODE TO ANALYZE.....	31
9.1	Objective: understand the root cause of the performance problem	31
9.2	Critical success factor to succeed to diagnose	31
9.2.1	Check you have all needed input like described in previous steps	31
9.2.2	Recommended to have database administration skills	31
9.3	Step1: Sort by customer loss of service event and assign a time interval.....	31
9.3.1.1	Objective: correlate error logs, resource usage with final user loss off services..	31
9.3.1.2	Example: sort problems by gravity and assign on time interval	31
9.4	Step2: Reduce the technical scope analysis	32
9.4.1	Objective: For each event type and time interval find what parts of PLM architecture as problems.....	32
9.4.1.1	Application server part	32
9.4.1.2	Database server part	32
9.4.1.3	Database instance part.....	32
9.4.2	If available get a global vision using diagnostic tools like Foglight, Dynatrace , Oracle database console	32
9.4.2.1	Objective: understand DB tier, application server, network impact on performances.....	32
9.4.2.2	Foglight example where database is not the root cause: resource part used by a PLM service	32
9.4.2.3	Example: database tier is the root performance cause	32
9.4.3	Analyze PLM application server logs: check if exist persistence errors.....	32
9.4.4	Analyze database part:	32
9.4.4.1	Check the database alert file	32
9.4.4.2	Extract a global vision from AWR snapshots of Instance/db sever waits and limits	32
9.4.4.3	Extract a global vision from STD_DB_TUNING of database server operating system	33

9.4.4.4	Extract a global vision from STD_DB_TUNING of database instance	33
9.4.5	Focalize on part having problems and assign interval time.....	33
9.4.5.1	When database part is the root cause of loss of services	33
9.4.5.2	When database part is not the root cause => not in the scope of this document	33
9.4.6	Check changes with possible impact on instability or connections	33
9.4.6.1	Check main functional or application usage changes or aggravating changes ...	33
9.4.6.2	Check technical main or aggravating changes (db server example).....	33
9.5	Step3: Correlate all events/time intervals: Root and aggravating causes!	34
9.5.1.1	Typical example on 32 bits	34
10.	APPENDIX: typical Oracle ORA-XXXX CRITICAL errors	35
10.1	ORA-00060 errors: Deadlocks (contentions) due to missing index on foreign key	35
10.1.1.1	Needed database skills: advanced	35
10.1.1.2	Understand deadlock type	35
10.1.1.3	Step1: Detect deadlock from alert file and get the trace file	35
10.1.1.4	Step2: Extract trace file main information.....	35
10.1.1.5	Step2: get and start database bundle STD_LOCK script to manage LOCKS	36
10.1.1.6	Step3: Choose STD_LOCK.cmd option 3 to get locked objects	37
10.1.1.7	Step4: Investigate if there is a foreign key between locked objects using STD_FIND.cmd	37
	Objects are CONTENTMONITORINGINFO and.....	37
10.1.1.8	Step5: Investigate if there is a not a missing index on the foreign key column....	38
11.	APPENDIX: typical STD_DB_TUNING ANALYSIS.....	38
11.1	Remainder: how to get results	38
11.1.1	Starting.....	38
11.1.2	Typical input.....	38
11.1.3	Output: Two files (OS and Oracle instance information) with timestamp.....	39
11.2	Get operating system information on windows (tmp_STD_DBtuning_systeminfo)	39
11.2.1.1	Check OS version, free physical and virtual memory.....	39
11.3	Get instance information on windows (tmp_STD_DBtuning_<hostname>).....	40
11.3.1	Most used global information useful for performance analysis	40
11.3.1.1	Time since ORACLE instance has been started	40
11.3.1.2	How to compute needed memory according to usage	40
11.3.1.3	Global waits	40
11.3.1.4	Top 10 IO consumers objects by Tablespace	42
11.3.2	Most used detailed information useful for performance analysis	42

11.3.2.1	Check if statistics are computed at regular interval	42
11.3.2.2	Schema: number of objects (missing indexes can impact performances)	42
11.3.2.3	Count rows in each schemas: check bigger tables	43
11.3.2.4	Most consumers statements extract: top 10 by type (elapse, CPU, exec, IO, get, parsing...)	43
12.	APPENDIX: typical Oracle AWR snapshot analysis	44
12.1	Requirements	44
12.1.1	Check snapshot interval match with final user loss of service	44
12.2	Step1: get a global vision of database server Instance/OS resource usage and limit.....	44
12.2.1	Objective: Check what kind of “wait class”: DB CPU/ DB IO or other?	44
12.2.2	IO: how to detect abnormal IO usage and waits	44
12.2.2.1	Check Top foreground wait class	44
12.2.2.2	Check IO speed (Slow if IO avg read >=10 ms)	44
12.2.2.3	Example of Intensive TEMP usage instead of memory.....	45
12.3	Step2: find consumers.....	45
12.3.1.1	Go to the SQL statistics link.....	45
12.3.1.2	Example: Find most elapse consumer	45
13.	APPENDIX: ORACLE DATABASE console how to	47
13.1	Since 11GR1: How to make PERFORMANCE tab available	47
13.1.1	About Performance tab Oracle licensing or SQL PROFILE usage considerations.....	47
13.1.2	11GR2 example using SQL*Plus	47
13.1.2.1	Temporary enable control_management_pack_access if not yet enabled	47
13.1.3	11GR2 example using database console.....	47
13.2	How to enable a SQL profile computed by Automatic SQL Tuning	48
13.2.1	Caution: Each potential SQL profile should be tested before in Pre-production and validated by the R&D	48
13.2.2	Steps.....	48
13.2.2.1	Step: Select a statement and push the “View implementation” button.....	50
13.2.2.2	Step: If a new database console recommend new indexes wait for R&D agreement.....	51
13.3	How to enable a new SQL profile by running SQL tuning advisor	51
13.3.1	Caution:	51
13.3.1.1	SQL Tuning Advisor use a lot of resource: not recommended during application usage	51
13.3.1.2	Each potential SQL profile should be tested before in Pre-production and validated by the R&D	51
13.3.2	Steps:.....	51

13.3.2.1	Run your statement or choose a statement already running.....	51
13.3.2.2	Schedule a SQL Tuning Advisor from database console.....	51
13.4	How to find and kill a big consumer impacting the PRODUCTION	52
13.4.1.1	Connect to the database console	52
13.4.1.2	Select Performance Tab	52
13.4.1.3	Select top activity link	52
13.4.1.4	Find the most consumer session ID	52
13.4.1.5	Go to Top consumer list.....	53
13.4.1.6	Order by PGA Memory and search for the Session ID.....	53
13.4.1.7	Select the session you want to kill and press Kill Session	53
13.4.1.8	Return to the top activity to check if your session has been killed	53
13.5	How to find and trace a big consumer (DEV environment).....	54
13.5.1.1	Clean your Oracle diag log folder	54
13.5.1.2	Connect to the database console	54
13.5.1.3	Select Performance Tab	54
13.5.1.4	Select top activity link	54
13.5.1.5	Find the most consumer session ID	54
13.5.1.6	Go to Top consumer list.....	55
13.5.1.7	Order by PGA Memory and search for the Session ID.....	55
13.5.1.8	Check it is the good session with good module and machine.....	55
13.5.1.9	Enable SQL trace.....	55
13.5.1.10	Select "Trace with bind information"	55
13.5.1.11	Check the SQL trace is enabled.....	55
13.5.1.12	Caution: Once your session ended, disable the trace.....	55
13.5.1.1	Once your session ended check the trace under the diag folder	55
14.	APPENDIX: Definition of Components and concepts	56
14.1.1	Oracle database server, Oracle instance	56
14.1.2	Oracle schema, Oracle database user	56
14.1.3	Oracle database console (Oracle Enterprise Manager)	56
14.1.4	Lectra Database bundle.....	56
15.	APPENDIX: PLM TYPICAL PERFORMANCE PROBLEMS.....	57
15.1	On Oracle 11GR1 only (PLM <v3)	57
15.1.1	PLM search SKU can use 100% CPU and is slow:.....	57
15.1.1.1	Cause: Oracle bug 6438892 (11GR1 only)	57
15.1.2	Fix	57



Modifications made to the document since its last publication are highlighted in [blue](#).

1. GENERAL OVERVIEW

1.1 Scope of the Document

1.1.1 Explain minimal best practice to get scalability on Windows in a PLM context

To do that, explain:

- Operating system configuration
- Database configuration

1.1.2 Explain what to do if a scalability problem occurs

1.1.3 PLM scope: all PLM versions

1.2 Intended Audience

To use this document, you need to be familiar with the following:

- Windows servers 2003/2008R2/2012 and have installed and tested them on your computer system
- General concepts and basic Oracle database administration tasks
- Lectra standards and best practices for databases

1.3 Clarification of responsibilities between Lectra and Customers

- It is the Customer responsibility to manage the Production environment, not Lectra. The objective is to help Customer able to manage by itself by explaining what the Lectra PLM constraints are and not to do the customer job.
- Lectra can help by providing best practices, by helping the customer to test on UAT environment before applying in Production but not doing the Customer job on the Production environment like doing the daily management (backup, diagnose ...) and sub-systems (IO subsystem, ...) management tasks

1.4 Clarification of Lectra support team organization

To save time and guarantee a good level of services for all customers Lectra support team follow some rules:

- Will ask a good level of context definition (see chapter “How to diagnose a context and send information to support”), this to be able to diagnose, reproduce in a short delay, and compare. This is possible because Lectra professional services team has the needed skills to be able to do a first level of database diagnostic and able to send all relevant information to support team
- will try to first use remote diagnostic tools as possible like Foglight , Dynatrace or others , this to be able to diagnose and detect any resource or application usage change during time
- Avoid as possible sessions that use a lot of resource time like WebEx (Most of the time WebEx usage come when the context has not been well defined) preferring VPN usage. This should be an exceptional event that needs a punctual Production access to make some more complex investigations.

1.5 Related Documentation

Source	Description	Localization/name
Lectra documentation	LECTRA FASHION PLM Database troubleshooting guide Target: ORACLE on Windows	Oracle_Windows_Troubleshooting_en.docx
	LECTRA FASHION PLM Database Non-interactive Installation and Configuration How To	Oracle_Windows_HowTo_en.docx
	LECTRA FASHION PLM Database Non-interactive Installation and Configuration Quick Guide target: Oracle on Windows	Oracle_Windows_Quick-Installation_en.docx

1.6 Organization

Chapter 2: "COMMON REQUIREMENTS FOR ALL STEPS"

Explain all basic delivered scripts or tools needed to be able to diagnose and fix.

Chapter 3: "OPTIMIZATION TASKS BEFORE INSTALLING PLM DATABASE"

Explain minimal operating system configuration to guarantee performance

Explain what is scalability and capacity planning

Explain Lectra recommendations on database instance memory

Chapter 4: "POST-INSTALLATION OPTIMIZATION TASKS"

Explain optimization task to be done after silent installation

Chapter 5: "SOLVE PERFORMANCE PART1: COLLECT GENERAL FUNCTIONNAL CUSTOMER CONTEXT"

To solve performance: How and what can of Customer information we need to diagnose

Chapter 6: "SOLVE PERFORMANCE PART2: COLLECT RELEVANT FUNCTIONNAL INFORMATION"

To solve performance: How and what can of functional information we need to diagnose

How to qualify the loss of service

Chapter 7: "SOLVE PERFORMANCE PART3: COLLECT RELEVANT TECHNICAL INFORMATION"

To solve performance: How and what can of technical information we need to diagnose

How to use database bundle scripts, database console to get information

Chapter 8: "SOLVE PERFORMANCE PART4: CENTRALIZE ALL INFORMATION TO ANALYZE"




To solve performance: How to centralize and package all information needed to be able to diagnose, reproduce. With this package support team can do a full analysis

Chapter 9: "SOLVE PERFORMANCE PART5: METHODE TO ANALYZE"

To solve performance: Explain the method used by Lectra support team to analyze

With this method it is possible to do a root cause analysis

1.7 Conventions

	Note
	Tips and hints
	Warning

1.7.1 Naming conventions:

PLM	=	Product Life Management
PDM	=	Product Development Module
WLP	=	Workflow Management and Line Planning modules
<ORA_ VERSION>	=	10GR2 SEO (Oracle 10.2.0.1 Standard Edition One)
		11GR1 SEO (Oracle 11.2.0.1 Standard Edition One)
		11GR202 SEO (Oracle 11.2.0.2 Standard Edition One)
		11GR204 SEO (Oracle 11.2.0.4 Standard Edition One)
<PLM_VERSION>	=	PLM Minor compatible version (used for patches)
<DATABASE_BUNDLE_PATH>	=	path\<Database bundle root folder>
<Database bundle root folder>	=	PLM < v3: ORA<ORA_VERSION>_SE_WIN
	=	PLM >= v3: ORA_SE_WIN
UAT	=	User Acceptance Testing environment

2. COMMON REQUIRMENTS FOR ALL STEPS

2.1 Recommended to get the most recent database bundle

The most recent scripts are more powerful to diagnose and administrate

Since PLM V3 script have a good level of functionalities

Unzip database bundle under a path without blanks

2.2 Lectra database bundle provided scripts to save time to configure OS

Since PLM v2r3, database bundle provides scripts to help for pre-installation tasks under
<DATABASE_BUNDLE_PATH>\<APP_ALIAS>\silent_install\

STD_prepare_operating_system.cmd

This will start a menu with a list of choice that matches all steps to be done, starting Windows wizards and suggesting good configuration values when needed.

STD_prepare_operating_system_help.txt

More details on how to configure (Can be called using choice 11: help)

Some optimization can be done easier with provided scripts available since PLM V2R2ga database bundle

For old PLM version (V1R5, V2RM or lower than PLM V2R2ga), you need to get PLM V2R2ga database bundle

Since PLM V2R2ga (database bundle 2.2.0.31.1) the Oracle database bundle contains all needed scripts.

3. OPTIMIZATION TASKS BEFORE INSTALLING PLM DATABASE

3.1 Understand what is Scalability

Scalability is a system's ability to process more workload, with a proportional increase in system resource usage. In other words, in a scalable system, if you double the workload, then the system uses twice as many system resources. This sounds obvious, but due to conflicts within the system, the resource usage might exceed twice the original workload.

Examples of poor scalability due to resource conflicts include the following:

- Applications requiring significant concurrency management as user populations increase
- Increased locking activities
- Increased data consistency workload
- Increased operating system workload
- Transactions requiring increases in data access as data volumes increase
- Poor SQL and index design resulting in a higher number of logical I/Os for the same number of rows returned
- Reduced availability, because database objects take longer to maintain

3.2 Need to work on capacity planning

3.2.1 Definition

Capacity Planning is the process of determining what type of hardware and software configuration is required to meet application needs. Like performance planning, capacity planning is an iterative process. A good capacity management plan is based on monitoring and measuring load data over time and implementing flexible solutions to handle variances without impacting performance.

3.2.2 Recommended steps

3.2.2.1 Step 1: Determining Performance Goals and Objectives

- The anticipated number of users.
- The number of concurrent sessions.
- The number of SSL connections required.

- The number and size of requests.
- The amount of data and its consistency.
- Determining your target CPU utilization.

Performance objectives are limited by constraints, such as

- The configuration of hardware and software such as CPU type, disk size versus disk speed, sufficient memory.
- The ability to interoperate between domains, use legacy systems, support legacy data.
- The security requirements and use of SSL. SSL involves intensive computing operations and supporting the cryptography operations in the SSL protocol can impact the performance of the Web Server.
- Development, implementation, and maintenance costs.

3.2.2.2 Step 2: Measuring Your Performance Metrics

3.2.2.3 Step 3: Identifying Bottlenecks in Your System

3.2.2.4 Step 4: Implementing a Capacity Management Plan

3.2.3 Lectra recommendations on database instance memory

3.2.3.1 PLM old version (version <v3)

PLM version	OS Type	Oracle version	Concurrent sessions	Configuration type	Recommended Memory_target or sga if 10GR2	Other OS configuration	Status/Remark
<v3	Windows 32 bits Standard Edition	11.1.0.6	1<= users < 30	SMALL	1032MB (Default)		Certified
			30<= users < 150	MEDIUM	2100MB	/3GB	Certified
			>150	LARGE	2100MB<= <=OS limit by process	/3GB /AWE	Recommended to switch on Windows Enterprise edition

3.2.3.2 PLM Versions >= v3

PLM version	OS Type	Oracle version	Concurrent sessions	Configuration type	Minimal recommended Memory_target	Other OS configuration	Status
>=v3	Windows 64 bits	11202 /11204	1<= users < 30	SMALL	Default: 3GB Min: 2GB	NA	Certified
			30<= users < =100	MEDIUM	Default: 3GB Min: 3GB	NA	Certified
			> 100 users	LARGE	Default: 3GB Min: 3GB + add 3GB for each additional 200 users	NA	Certified

Of course previous recommendations are starting values which need to be adjusted according to each Customer real application usage during time.

3.3 CHECK OPERATING SYSTEM CONFIGURATION (WINDOWS)

3.3.1 Most of the configuration task can be done using database bundle script

Since PLM v2r3, database bundle provide scripts to help for pre-installation tasks like managing host file, virtual memory, network configuration

Under

<DATABASE_BUNDLE_PATH>\<APP_ALIAS>\silent_install\prepare_operating_system

[STD_prepare_operating_system.cmd](#)

This will start a menu with a list of choice that matches all steps to be done, starting Windows wizards and suggesting good configuration values when needed.

[STD_prepare_operating_system_help.txt](#)

More details on how to configure (Can be called using choice 11: help)

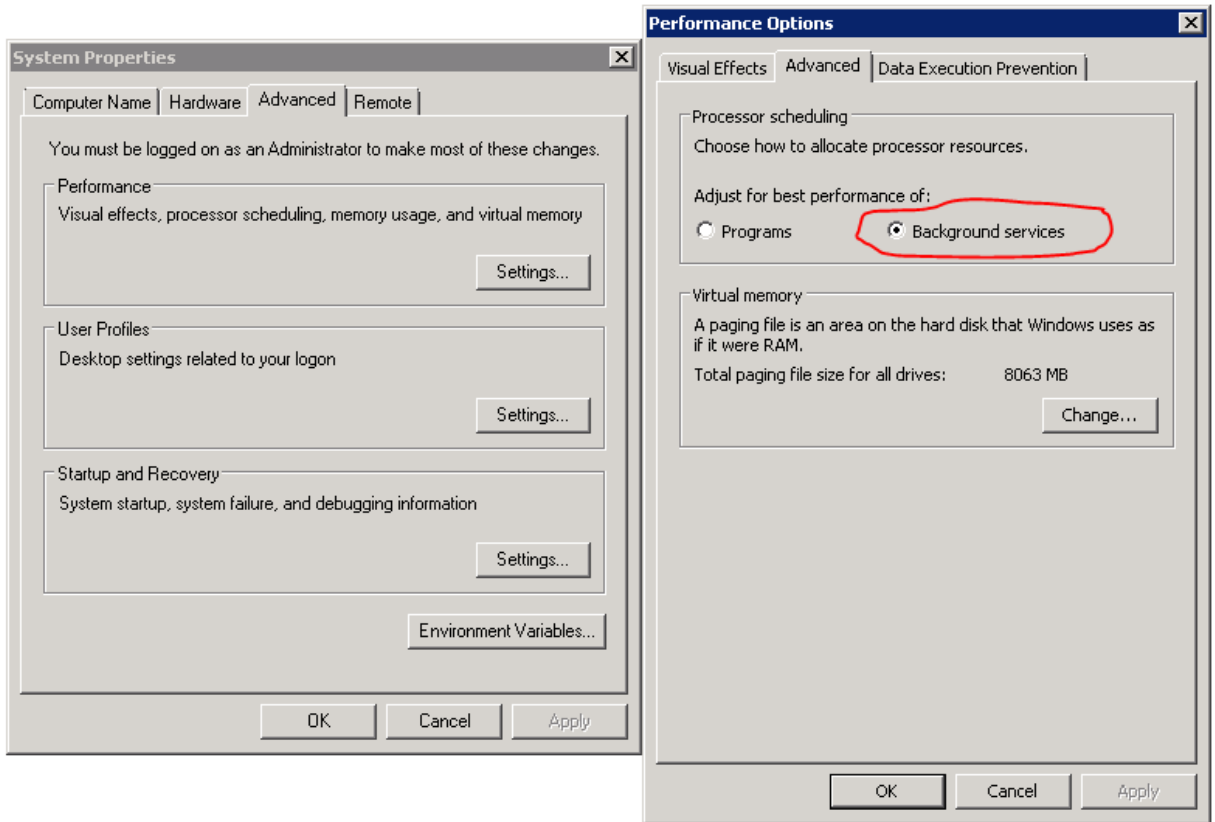
3.3.2 Memory and processor usage parameters

Start STD_prepare_operating_system.cmd

Choose the “Show Physical and virtual memory configuration” menu option

```
Available Page File Space (KB): 998,776000
Do you want to change virtual memory (Y/N) [N]?Y
Press any key to continue . . . _
```

- Advanced system settings > Advanced
- Performance > Settings
- Example on Windows 2008R2 server:



3.3.2.1 Adjusting Paging File Size

Example of rules on Oracle 11G:

If RAM = 1024MB to 2048Mb then

1.5 times RAM

Else if

RAM > 2048MB and < 8192MB

then match RAM

else RAM > 8192MB then

.75 times RAM

3.3.3 Windows 32 bits only: set memory /3GB and /PAE

3.3.3.1 Objective: avoid out of memory errors (ORA-4030)

Windows Server 2003 includes support for a startup switch that lets you tune the allocation of use of memory and memory address space. Regardless of the amount of physical memory in your system, Windows uses a virtual address space of 4 GB, with 2 GB allocated to user-mode processes (for example, applications) and 2 GB allocated to kernel-mode processes (for example, the operating system and kernel-mode drivers).

On systems that have 1 GB or more of physical memory, these two startup switches can be used to allocate more memory to applications (3 GB) and less memory to the operating system (1 GB).

This additional virtual address space helps reduce the amount of memory fragmentation in the virtual address space of the Exchange information store process.

The /3GB switch is used to effect this allocation change.

The switch is entered in the system's boot.ini file and takes effect after a restart.

PAE is the added ability of the IA32 processor to address more than 4 GB of physical memory

For more information on how to set in your boot.ini the /3GB and PAE, refer to following links:

About 3GB:

[http://technet.microsoft.com/fr-fr/library/bb124810\(EXCHG.65\).aspx](http://technet.microsoft.com/fr-fr/library/bb124810(EXCHG.65).aspx)

About PAE:

<http://support.microsoft.com/kb/283037/en-us>

<http://www.microsoft.com/whdc/system/platform/server/pae/paedrv.msp>

3.3.3.2 Example on XP/Windows2003 server 32 bits

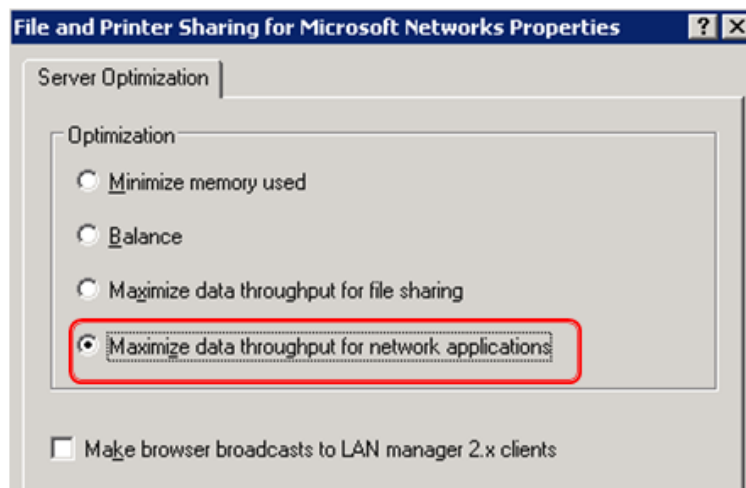
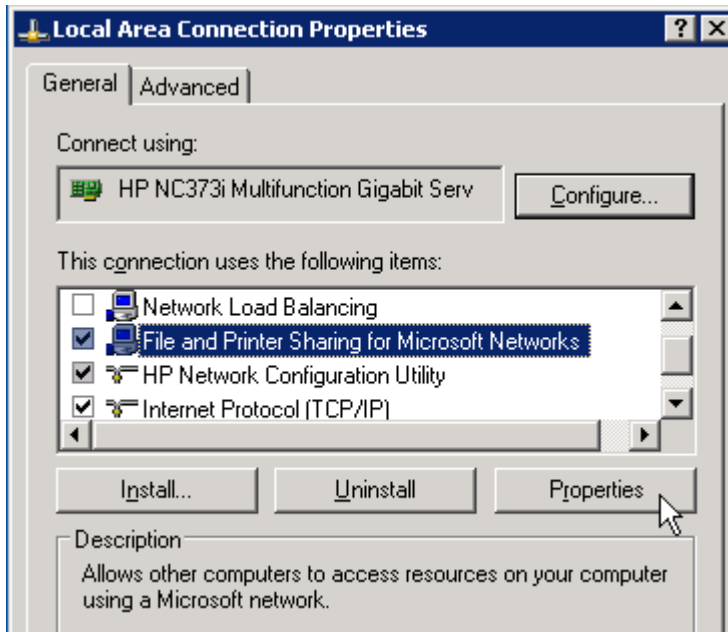
System properties>startup and recovery> push the "Edit" button to edit the boot.ini and add the /3GB switch

Example: multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Windows Server 2003, 3GB" /fastdetect /NoExecute=OptOut /3GB

3.3.4 Adjusting network configuration (Windows 2003 Server)

Network configuration > properties> File and Printer Sharing to Microsoft Network properties

=> Check "Maximize data throughput for network applications"



3.3.5 Storage: About raid sub-system

Always place log files on RAID 1+0 (or RAID 1) disks, never RAID5.

3.3.6 Storage: use SSD when intensive IO

3.3.6.1 PLM v2r3 32 bits example: not enough memory can generate intensive TEMP IO

PLM v2r3 is certified only on Windows 2003 server 32 bits which is limited to 3GB by process (need to configure /3GB)

Performance problems can occurs when the following conditions are true:

- Great number of users
- Great usage of Workflow management

When Oracle has not enough memory, Oracle use TEMP to be able to do big sorts

Moving TEMP tablespace on SSD disk can workaround too much IO

3.3.7 Storage: check defragmentation

The physical fragmentation of files on disks can cause a performance hit.

It is therefore advisable to perform a defragmentation when the need appears.

CAUTION: Do not defragment the disks which are files databases.

3.3.7.1 Steps to defragment databases

- Stop the database.
- Move the files physically on another logical path
- Defragment
- Move files to their original location

3.3.7.2 Antivirus configuration

Never scan dbf, log, trc file extensions

4. POST-INSTALLATION OPTIMIZATION TASKS

4.1 Change the memory target from default to recommended value

4.1.1 Adjust Oracle instance memory (Refer to capacity planning chapter)

After PLM silent installation Oracle the Oracle instance is set to a minimal default memory target which is PLM version dependant

To see the current Oracle instance memory and modify it is recommended to use database provided script

PLM V3 example:

Under <DATABASE_BUNDLE_PATH>\PLM\admin\manage_schema

Run **STD_configure_instance.cmd**

4.2 Summary of recommended regular tasks to automates

4.2.1 Check statistics are run at regular interval in an optimal period of time

4.2.1.1 Checking using sqlplus

```
Sqlplus <schema_owner>/<schema_owner_pwd>@<service_name>
```

```
Set pagesize 1000
```

```
Select table_name, to_char(last_analyzed,'YYYY/MM/DD HH24 :MI')
```

```
From USER_TABLES
```

```
Order by 1 ;
```

Example :

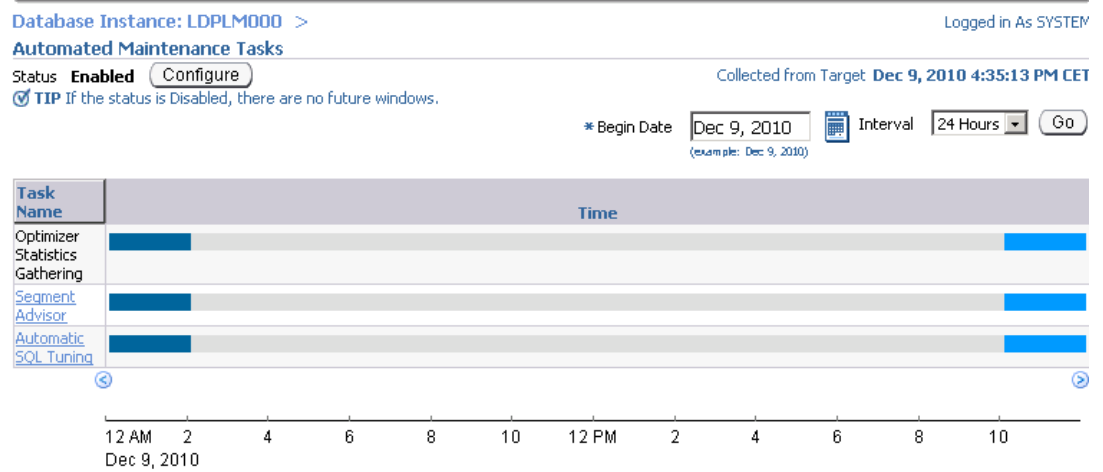
TABLE_NAME	HOUR
ABSTRACTIMAGE	2010/12/01 08 :48
ABSTRACTMULTIMEDIAFILE	2010/12/01 08 :48
ACT	2010/12/01 08 :48
...	

4.2.1.2 Checking schedule using database console

Connect to the database console

(Refer to “Lectra Fashion PLM Non-interactive Installation and Configuration How To “ chapter “HOW TO ACTION LIST”, ACTION 0002 : HOW TO CONNECT TO THE DATABASE CONSOLE)

Go to « SERVER » tab > «Automated Maintenance Tasks »



4.2.1.3 Checking using database bundle STD_DB_tuning script

Under

<DATABASE_BUNDLE_PATH>\<APP_ALIAS>\ORA_SE_WIN\PLM\admin\manage_schema\script\advanced\tuning

Start **STD_DB_tuning.cmd**

Search in the result the string "LAST ANALYZED TABLES"

Example: check MIN_LAST_ANALYZED, MAX_LAST_ANALYZED column values

- LAST ANALYZED TABLES FOR APPLICATION USERS AND SCHEMA STATUS

OWNER	TABLE_COUNT	MIN_LAST_ANALYZED	MAX_LAST_ANALYZED	ACCOUNT_STATUS	CREATION_DATE
PLM_01	416	2014/03/06 21:50	2014/04/10 22:00	OPEN	2014/04/09 14:52
PLM_01_ADM	3	2014/04/24 22:00	2014/05/05 22:00	OPEN	2014/04/24 16:06
PLM_02_ADM	3	2014/05/09 22:00	2014/05/09 22:00	OPEN	2014/05/09 16:39
PLM_08_ADM	3	2014/05/09 22:00	2014/05/09 22:00	OPEN	2014/05/09 15:34
PLM_ADMIN	7	2014/04/14 22:00	2014/04/14 22:00	OPEN	2014/04/14 17:10

- LAST ANALYZED INDEXES FOR APPLICATION USERS

OWNER	INDEX_COUNT	MIN_LAST_ANALYZED	MAX_LAST_ANALYZED
PLM_01	1000	2014/03/07 14:55	2014/04/09 22:00
PLM_01_ADM	4	2014/04/24 22:00	2014/05/05 22:00
PLM_02_ADM	4	2014/05/09 22:00	2014/05/09 22:00
PLM_08_ADM	4	2014/05/09 22:00	2014/05/09 22:00
PLM_ADMIN	7	2014/04/14 22:00	2014/04/14 22:00

4.2.1.4 Apply statistics, if not applied, using database bundle script

Refer to « Lectra Fashion PLM Non-interactive Installation and Configuration How To », « MANAGE SCHEMA HOW TO » chapter

5. SOLVE PERFORMANCE PART1: COLLECT GENERAL FUNCTIONAL CUSTOMER CONTEXT

5.1 Describe the Customer context

5.1.1 What is the environment type? (PROD/UAT/DEV/TRAINING?)

5.1.2 Collect PLM schema information like functional version (ex: v3r3sp3)

Although functional version can be seen during PLM application logging since PLM V3, a better way to obtain schema information is to use database bundle script

PLM V3 example:

Under <DATABASE_BUNDLE_PATH>\PLM\admin\manage_schema

Run **PLM_CHECK.cmd**

5.1.3 PLM V3: Collect WLP schema information that can have impact on performance

Under <DATABASE_BUNDLE_PATH>\PLM\admin\manage_schema

Run **WLP_CHECK.cmd**

6. SOLVE PERFORMANCE PART2: COLLECT RELEVANT FUNCTIONAL INFORMATION

6.1 Objective: describe the loss of service in a standard way

6.2 Get all relevant functional information

6.2.1 What is the frequency and time interval of the error/loss of service?

6.2.1.1 Not specified or unknown=> try to get the information

6.2.1.2 Consistently (need timestamp or time interval)

6.2.1.3 Intermittent (need timestamp or time interval)

6.2.1.4 One-time (need timestamp or time interval)

(1) Timestamp is important to be able to correlate with technical components errors

6.3 Explain the gravity of the problem

6.3.1 Are you experiencing a mission or business critical loss of service?

6.3.1.1 Requires immediate and continuous effort to solve? Y/N

6.3.1.2 Easy to recover, bypass or workaround? Y/N

6.3.1.3 Does the application work normally after the problem occurs? Y/N

6.3.2 Describe the functional impact:

6.3.2.1 Only on one user, on site, on team?

6.3.2.2 What functional part is impacted?

6.3.2.3 Describe the scenario to reproduce

6.4 Explain the last application usage or configuration changes

6.4.1 Recent upgrades?

6.4.2 Big data load? (ETL....)

6.4.3 New customer development (CBR, reports,ETL...) deployed

6.4.4 New application usage part, new users

6.4.5 Application configuration changes (security...)

7. SOLVE PERFORMANCE PART3: FIND RELEVANT TECHNICAL INFORMATION

7.1 Objective

Get all relevant technical information to be able to make support team or a database expert able to analyze the problem and find the tree cause of the loss of service

7.2 Get logs

7.2.1 Objective: to be able to understand what part of the architecture has a problem

7.2.2 Get application server logs

7.2.2.1 Get persistence.log (not empty when problems comes from database tier)

7.2.2.2 Get apache logs (needed to understand application usage)

7.2.3 Get database minimum server logs

7.2.3.1 Get and compress database alert file and last month traces files

Get and compress files under

<drive>\app\oracle\diag\rdbms\<INSTANCE_NAME>\<INSTANCE_NAME>\trace

Ex: C:\app\oracle\diag\rdbms\ldplm000\ldplm000\trace

=> alert_ldplm000.log and last month *.trc trace files

7.2.3.2 Get and compress listener.log extract (to check connections problems)

Get and compress file:

- <drive>\app\oracle\diag\tnlsnr\<HOSTNAME>\listener\trace\listener.log

Remark:

If too big, before compress, extract the 100 last rows using:

<DATABASE_BUNDLE_PATH>\<APP_ALIAS>\silent_install\tool\tail.exe

tail listener.log -n 100 > extract_listener100rows.txt

7.3 Get operating system and database global vision

7.3.1 Using database bundle get STD_DB_TUNING result

Under <DATABASE_BUNDLE_PATH>\PLM\admin\manage_schema\script\advanced\tuning\

Run **STD_DB_TUNING.bat**

⇒ Get the two result files (**STD_DB_Tuning_xxxx.lst result files**)

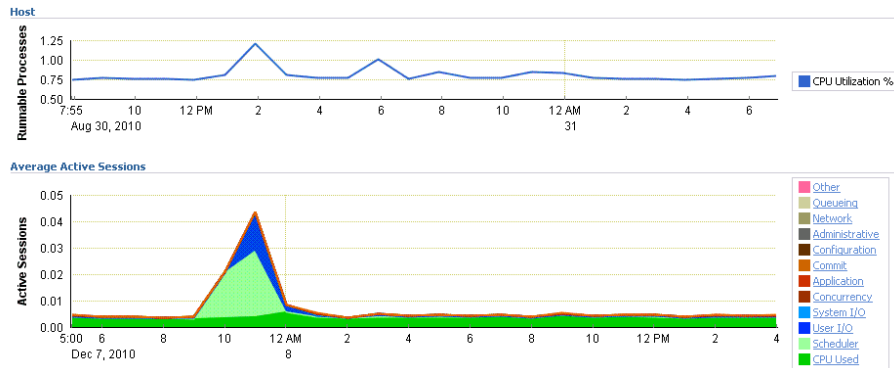
7.3.2 Using database console (if available) find interval with loss of services

7.3.2.1 Connect to the database console

(Refer to “Lectra Fashion PLM Non-interactive Installation and Configuration How To “ chapter “HOW TO ACTION LIST”, ACTION 0002 : HOW TO CONNECT TO THE DATABASE CONSOLE)

7.3.3 Go to the performance dashboard and find interval with resource usage

7.3.3.1 Check CPU and IO resource used and note the time interval



For more information refer to Oracle database documentation

7.3.3.1 Check top activity

For more information refer to Oracle database documentation or refer to APPENDIX “ How to find and kill a big consumer impacting the PRODUCTION”

7.3.4 Using database bundle get snapshots selecting loss of service time interval

7.3.4.1 Check if customer has enabled control_management_pack_access

In order to produce Oracle tuning reports with measures, access AWR views or use the diagnostic information from any part of the Automatic Workload Repository the Diagnostic Pack License is required.

To check if customer has enabled it:

Set ORACLE_SID=LDPLM000

Sqlplus sys/**** as sysdba

Show parameter control_management_pack_access

SHOW PARAMETER control_management_pack_access

NAME	TYPE	VALUE
control_management_pack_access	string	DIAGNOSTIC+TUNING

7.3.4.2 Temporary enable control_management_pack_access if not yet enabled

```
alter system set control_management_pack_access="DIAGNOSTIC+TUNING"
scope=both;
```

7.3.4.3 Do a database snapshot and get the html report result for each snapshot

Choose a slow period in the day (minimum 2 Hours), do a database snapshot on each slow period:

Under

<DATABASE_BUNDLE_PATH>\PLM\admin\manage_schema\script\advanced\tuning\

Run **STD_DBtuning_AWR_ASH.cmd**

=> Choose option 2: "Generate AWR report from begin to end snapshot"

Output example:

```
Enter Database Service [LDPLM000]?
Enter Sys Password [LDPLM000]?
Enter on the next question: "Enter value for report_type" => this will choose
default value
Enter value for num_days:1
```

output example for next questions:

Listing the last day's Completed Snapshots

Instance	DB Name	Snap Id	Snap Started	Snap Level	
ldplm000	LDPLM000	1779	02 Aug 2013 00:00	1	
		1780	02 Aug 2013 01:00	1	
		1781	02 Aug 2013 02:00	1	
		1782	02 Aug 2013 03:00	1	
		1783	02 Aug 2013 04:00	1	
		1784	02 Aug 2013 05:00	1	
		1785	02 Aug 2013 06:00	1	
		...			
		1793	02 Aug 2013 14:00	1	
		1794	02 Aug 2013 15:00	1	

Specify the Begin and End Snapshot Ids

~~~~~

Enter value for begin\_snap: 1789

Begin Snapshot Id specified: 1789

Enter value for end\_snap: 1792

End Snapshot Id specified: 1792

Specify the Report Name

~~~~~

The default report file name is awrrpt_1_1789_1792.html. To use this name press <return> to continue, otherwise enter an alternative.

Enter value for report_name:

=> let the default

7.3.4.4 Once done disable control_management_pack_access if initially disabled

alter system set control_management_pack_access=**NONE** scope=both;

7.4 Check the last technical changes

7.4.1 Check each OS/database/application server/database configuration changes

7.4.2 Check if a new tier process is impacting resource usage

8. SOLVE PERFORMANCE PART4: CENTRALIZE ALL INFORMATION TO ANALYZE

8.1 Prepare a centralized compressed folder with all relevant information to analyze

8.1.1 Objectives:

8.1.1.1 Centralize in a standard way all relevant information to diagnose a customer

8.1.1.2 Be able when too complex to analyze to send to support team

8.1.2 Suggested check list and naming rules

8.1.2.1 **Root folder:** Audit_<customer>_<environment type>_<timestamp>

With <environment type>= PROD, UAT, DEV, and TRAINING

Example: Audit_MYCUSTOMER_PROD_20140530

8.1.2.2 Mandatory items

- Functional
 - Describe scenario to reproduce
 - (not mandatory: WebEx record)
 - Functional description (refer to chapter items)
- Database_logs
 - Trace
 - alert_ldplm000.log and last month *.trc trace files
 - listener.log 100 last row extract
 - Tuning
 - two STD_DB_Tuning.cmd result files (STD_DB_Tuning_xxxx.lst files)
 - all AWR snapshots reports you have done

- Schema information
 - PLM_CHECK.cmd results
 - WLP_CHECK.cmd result (PLM v3)
- Application_logs
 - Apache logs, persistence.log...

8.1.2.3 Not mandatory except to reproduce

- Database_dump
 - Compressed PLM schemas Export (or datapump)
- File_repository (when not too big)
 - PLM compressed file repository
- PLM_custo

9. SOLVE PERFORMANCE PART5: METHODE TO ANALYZE

9.1 Objective: understand the root cause of the performance problem

9.2 Critical success factor to succeed to diagnose

9.2.1 Check you have all needed input like described in previous steps

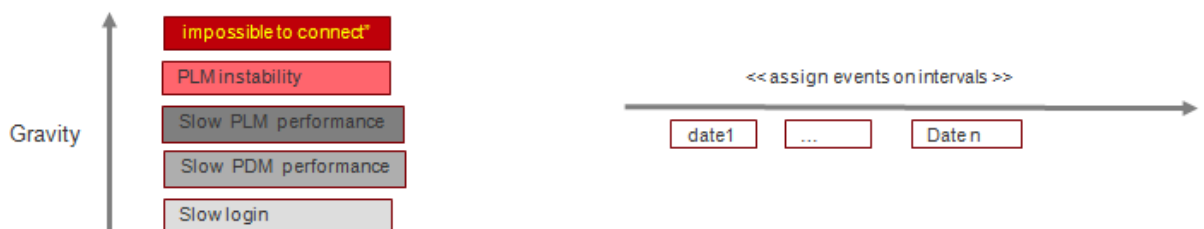
- Impossible to distinguish root cause from aggravating causes without all PLM and Database minimal logs (alerts, traces, global vision scripts) need timestamps to correlate scenarios and logs on all tiers,...
- Impossible to diagnose root cause without tools and manual scripts to run in parallel (database bundle) to simulate load and measure once consumer isolated
- Need tools to have a global vision (diagnostic tools like Foglight, Dynatrace , Oracle database console, AWR snapshots reports,...)
- Need tools to check if OS resource reach a normal/abnormal level according to load level

9.2.2 Recommended to have database administration skills

9.3 Step1: Sort by customer loss of service event and assign a time interval

9.3.1.1 Objective: correlate error logs, resource usage with final user loss off services

9.3.1.2 Example: sort problems by gravity and assign on time interval



9.4 Step2: Reduce the technical scope analysis

9.4.1 Objective: For each event type and time interval find what parts of PLM architecture as problems

9.4.1.1 Application server part

9.4.1.2 Database server part

9.4.1.3 Database instance part

9.4.2 If available get a global vision using diagnostic tools like Foglight, Dynatrace , Oracle database console

9.4.2.1 Objective: understand DB tier, application server, network impact on performances

9.4.2.2 Foglight example where database is not the root cause: resource part used by a PLM service



9.4.2.3 Example: database tier is the root performance cause

To be described.

9.4.3 Analyze PLM application server logs: check if exist persistence errors

If persistence.log is not empty on a server this means that there are database problems that impact PLM application

If errors, correlate error timestamps with final user loss of services

9.4.4 Analyze database part:

9.4.4.1 Check the database alert file

Check if you detect critical ORA-xxxx errors during the loss of service

See APPENDIX: typical Oracle ORA-xxxx critical errors

9.4.4.2 Extract a global vision from AWR snapshots of Instance/db sever waits and limits

See APPENDIX: typical Oracle AWR snapshot analysis

9.4.4.3 Extract a global vision from STD_DB_TUNING of database server operating system

APPENDIX: Typical STD_DB_TUNING analysis

9.4.4.4 Extract a global vision from STD_DB_TUNING of database instance

APPENDIX: Typical STD_DB_TUNING analysis

9.4.5 Focalize on part having problems and assign interval time

9.4.5.1 When database part is the root cause of loss of services

Example on database typical events (sorted by gravity):



9.4.5.2 When database part is not the root cause => not in the scope of this document

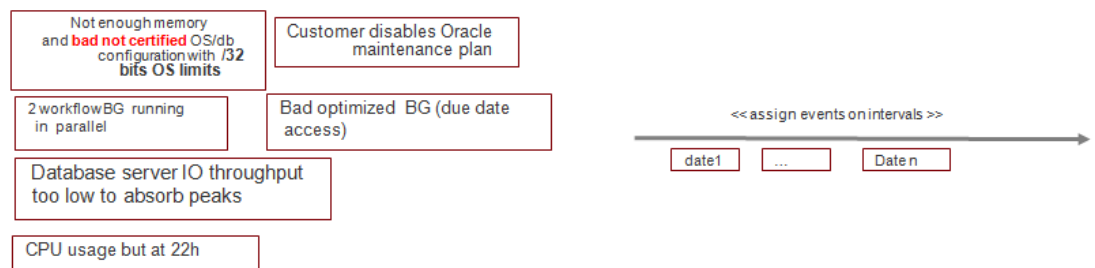
9.4.6 Check changes with possible impact on instability or connections

9.4.6.1 Check main functional or application usage changes or aggravating changes

Example:

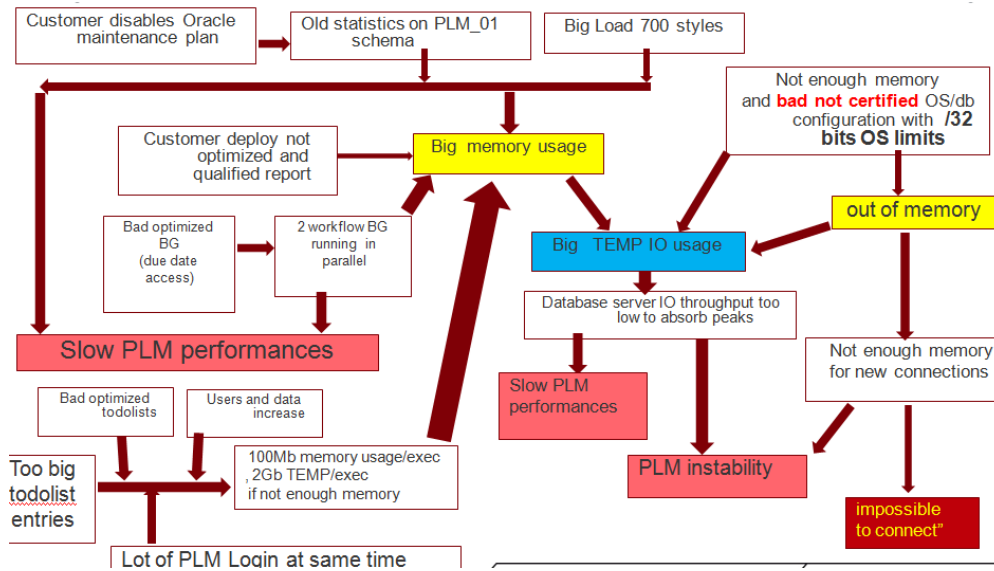


9.4.6.2 Check technical main or aggravating changes (db server example)



9.5 Step3: Correlate all events/time intervals: Root and aggravating causes!

9.5.1.1 Typical example on 32 bits

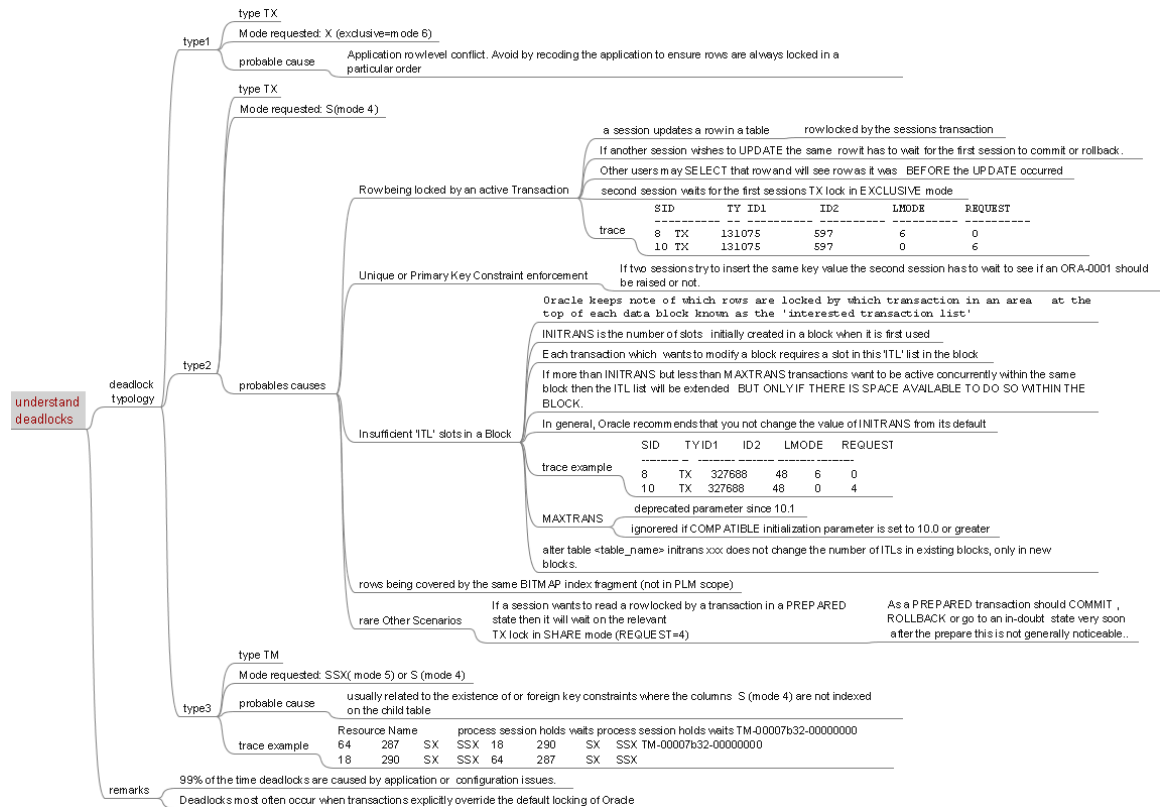


10. APPENDIX: TYPICAL ORACLE ORA-XXXX CRITICAL ERRORS

10.1 ORA-00060 errors: Deadlocks (contentions) due to missing index on foreign key

10.1.1.1 Needed database skills: advanced

10.1.1.2 Understand deadlock type



10.1.1.3 Step1: Detect deadlock from alert file and get the trace file

Example:

Wed Jan 30 12:09:35 2013

ORA-00060: Deadlock detected. More info in file

C:\APP\ORACLE\diag\rdbms\ldplm000\ldplm000\trace\ldplm000_ora_8776.trc.

Wed Jan 30 12:09:35 2013

10.1.1.4 Step2: Extract trace file main information

The main information needed to diagnose is:

- Username
- Deadlock Type (TM...)

- Hold type (SSX,...)
- Sessions statements (e.g delete from ...)
- Resource name in hexa (ex: 0032def50032def5)

```

*** 2012-02-21 07:22:23.651
DEADLOCK DETECTED ( ORA-00060 )

[Transaction Deadlock]

The following deadlock is not an ORACLE error. It is a
deadlock due to user error in the design of an application
or from issuing incorrect ad-hoc SQL. The following
information may aid in determining the deadlock:

Deadlock graph:
-----Blocker(s)-----      -----Waiter(s)-----
Resource Name                process session holds waits  process session holds waits
TM-0032def5-00000000         97    246    SX    SSX          99    681    SX    SSX
TM-0032def5-00000000         99    681    SX    SSX          97    246    SX    SSX

session 246: DID 0001-0061-000000FD      session 681: DID 0001-0063-00000422
session 681: DID 0001-0063-00000422      session 246: DID 0001-0061-000000FD

Rows waited on:
  Session 246: no row
  Session 681: no row

----- Information for the OTHER waiting sessions -----
Session 681:
  sid: 681 ser: 52905 audsid: 506260290 user: 91/PLM_SRDSPICV3
  flags: (0x45) USR/- flags_id1: (0x1) BSY/-/-/-/-
  flags2: (0x40008) -/-
  pid: 99 O/S info: user: SYSTEM, term: SRDSICBD4, ospid: 16188
  image: ORACLE.EXE (SHAD)
  client details:
    O/S info: user: eu_autobuildplm_svc, term: unknown, ospid: 1234
    machine: srdspicv3 program: JDBC Thin Client
    application name: JDBC Thin Client, hash value=2546894660
  current SQL:
  /* delete com.lectra.fip.server.persistence.contentmanagement.monitoring.ContentMonitoringImpl */ delete from
  CONTENTMONITORING where ID=:1

----- End of information for the OTHER waiting sessions -----

Information for THIS session:

----- Current SQL Statement for this session (sql_id=lgvgncha0czvt) -----
/* delete com.lectra.fip.server.persistence.contentmanagement.monitoring.ContentMonitoringImpl */ delete from
CONTENTMONITORING where ID=:1

```

10.1.1.5 Step2: get and start database bundle STD_LOCK script to manage LOCKS

Under <DATABASE_BUNDLE_PATH>\PLM\admin\manage_schema\script\advanced

STD_LOCK.cmd

```

*****
*                               DIAGNOSE LOCKS                               *
*                               *****                               *
* Objective: - Help to diagnose Locks                                       *
*            - Generate scripts                                             *
* Requirement:                                                                *
*****
1. Show current database locks on a specific schema
2. Deadlocks: Show missing indexes status on FK and generate scripts
   {can run scripts}

```

```

    { missing FK on indexes can generate deadlocks on hot access }
3. Deadlocks: Find locked object_name from object_id in hexa
    {coming from deadlock trace}
4. Show ITL waits and generate scripts {sys user}
    {too low ini_trans can generate deadlocks on hot access }
5. run sqlplus {sys user}
6. Windows Dos prompt
7. Exit
Choose (1/5) [1]?

```

10.1.1.6 Step3: Choose STD_LOCK.cmd option 3 to get locked objects

Choose your target schema (e.g **PLM_SRDSPICV3**)
 Enter the object_id1 in hexa coming from deadlock trace? (e.g **0032def5**)
 Enter the object_id2 in hexa coming from deadlock trace? (e.g **0032def5** same than previous here)

Output example:

```

OBJECT_ID OBJECT_NAME                                OBJECT_TYPE
-----
3333877 CONTENTMONITORINGINFO                       TABLE      <= blocked object

```

10.1.1.7 Step4: Investigate if there is a foreign key between locked objects using STD_FIND.cmd

Objects are CONTENTMONITORINGINFO and **CONTENTMONITORING**

Under <DATABASE_BUNDLE_PATH>\PLM\admin\manage_schema\script\advanced\

STD_FIND.cmd

```

*****
* CENTRALIZE ALL SEARCH
*
* Objective : - save time on support and DBA tasks
*             - Generate scripts
* Updated   : 2013/09/24
* Version   : 07
* Requirement:
*****
1. DATA SEARCH: Find which table.column contain a specific string
   objective: faster way to find table which contain almost a column
             matching a string
   For each table: - exit when the first row is founded
                  - show only the first column found
2. DATA SEARCH: Find which table.column contain a string
   - This can be LONG and resource consumer on big tables
   - because count matching values on each column of each table
   - On big database recommended to use the previous table list to restrict IO resource
   - Build a select to retrieve each matching row
3. STRUCTURE_DEPENDENCY: find table father and children dependancy using Foreign key
4. STRUCTURE_DEPENDENCY: show father FK and all levels of childs
5. STRUCTURE_DEPENDENCY: show dependencies using USER_SOURCE
   { Which PL/SQL structure use a specific statement part?}
6. run sqlplus {schema user}
7. Windows Dos prompt
8. Exit
Choose (1/5) [1]?_

```

Choose option 3

Output example:

```
Table (FK pointant vers) ? [TABLE1]
*****
CONTENTMONITORING

--- CONTENTMONITORING children tables:

CHILDREN                CONSTRAINT_NAME        C
-----
CONTENTMONITORINGINFO  CONTENTM_INFOS_FK      R

--- CONTENTMONITORING Fathers tables:
```

10.1.1.8 Step5: Investigate if there is a not a missing index on the foreign key column

e.g Here missing index on CONTENTMONITORINGINFO.MONITORING_ID column

This can be finding by running **STD_LOCK.cmd**, option 2

Caution: just generate create index script and don't modify the DDL without Lectra support team agreement.

This script generates scripts and once done ask if you want to apply suggested index or exit: choose always to exit (CTRL-C)

Check logs to see suggested indexes

11. APPENDIX: TYPICAL STD_DB_TUNING ANALYSIS

11.1 Remainder: how to get results

11.1.1 Starting

Under <DATABASE_BUNDLE_PATH>\PLM\admin\manage_schema\script\advanced\tuning\

Run **STD_TUNING.bat**

11.1.2 Typical input

```
*****
* Objective   : - Used by Support team to get a                               *
*              * OS global vision using systeminfo                         *
*              * Database tuning global vision                             *
*              - Useful when database console not available                 *
* Requirement:                                                                 *
* Target      : Certified from Oracle 11106 to 11204                         *
*****

compute OS global vision (systeminfo)
*****
Result on tmp_STD_DBtuning_systeminfo_WRDSVALRD04_20140521_14h59Min.lst
```

```

compute database global vision
*****
Application alias [APP]
PLM
service [LDPLM000]
sys password [LDPLM000]
USER is "SYS"

Name of the log file ? [tmp_STD_DBTuningWRDSVALRD04_20140521_14h59Min]

```

Wait for the next prompt (recommended to exit because long):

```

APPLICATION SQL STATEMENT: Top 10 by Version Count
*****
==> Do you want to find bigger segments? (Can be very long)
(CTRCL-C to exit or ENTER to continue)

```

11.1.3 Output: Two files (OS and Oracle instance information) with timestamp

Two files

- tmp_STD_DBTuning_systeminfo_<hostname>_<timestamp>
- tmp_STD_DBTuning_<hostname>_<timestamp>

Example:

- tmp_STD_DBTuning_systeminfo_WRDSVALRD04_20140521_14h59Min.lst
- tmp_STD_DBTuningWRDSVALRD04_20140521_14h59Min.lst

11.2 Get operating system information on windows (tmp_STD_DBTuning_systeminfo)

11.2.1.1 Check OS version, free physical and virtual memory

```

Host Name: WRDSVALRD04
OS Name: Microsoft Windows Server 2008 R2 Standard
OS Version: 6.1.7601 Service Pack 1 Build 7601
OS Manufacturer: Microsoft Corporation
OS Configuration: Standalone Server
OS Build Type: Multiprocessor Free
Registered Owner: Windows User
Registered Organization:
Product ID: 00477-001-0000421-84994
Original Install Date: 4/13/2011, 12:05:01 AM
System Boot Time: 4/22/2014, 11:53:36 AM
System Manufacturer: Hewlett-Packard
System Model: HP Compaq 8000 Elite CMT PC
System Type: x64-based PC
Processor(s): 1 Processor(s) Installed.
[01]: Intel64 Family 6 Model 23 Stepping 10 GenuineIntel ~1983 Mhz
BIOS Version: Hewlett-Packard 786G7 v01.02, 10/22/2009
Windows Directory: C:\Windows
System Directory: C:\Windows\system32
Boot Device: \Device\HarddiskVolumel
System Locale: en-us;English (United States)
Input Locale: fr;French (France)
Time Zone: (UTC+01:00) Brussels, Copenhagen, Madrid, Paris
Total Physical Memory: 8,063 MB
Available Physical Memory: 2,887 MB
Virtual Memory: Max Size: 16,125 MB
Virtual Memory: Available: 7,078 MB
Virtual Memory: In Use: 9,047 MB

```

```

Page File Location(s): C:\pagefile.sys
Domain: RD_VALIDATION
Logon Server: \\WRDSVALRD04
Hotfix(s): 2 Hotfix(s) Installed.
           [01]: KB2661254
           [02]: KB976902
Network Card(s): 1 NIC(s) Installed.
                 [01]: Intel(R) 82567LM-3 Gigabit Network Connection
                   Connection Name: Local Area Connection
                   DHCP Enabled: Yes
                   DHCP Server: 172.16.48.224
                   IP address(es)
                   [01]: 172.16.159.23

```

11.3 Get instance information on windows (tmp_STD_DBtuning_<hostname>)

11.3.1 Most used global information useful for performance analysis

11.3.1.1 Time since ORACLE instance has been started

```

INSTANCE: When ORACLE database has been started
*****

```

```

DB Startup Time      ELAPSE_SINCE_STARTUP
-----
12/05/2014 16:42:16 8 Days 22 Hour 19 Min 13 Sec

```

11.3.1.2 How to compute needed memory according to usage

Example:

```

SID                                VAL
-----
memory_target                      4312 Mo
memory_max_target                   4312 Mo

COMPONENT                          CURRENT_SIZE_MB MIN_SIZE_MB MAX_SIZE_MB MEMORY_GRANULE_MB
-----
shared pool                         1372           536       1372           4
SGA Target                          2312           2312      2312           4
PGA Target                          2000           2000      2000           4

-- SGA target advice

SGA_SIZE SGA_SIZE_FACTOR ESTD_DB_TIME ESTD_DB_TIME_FACTOR ESTD_PHYSICAL_READS
-----
1734      .75      2027605      1.2156      532809224
2312      1        1667987      1           160065258 current value
2890     1.25     1608106      .9641      123490347
3468     1.5     1599433      .9589      144939091
4046     1.75     1598098      .9581      123490347 better
4624     2        1597598      .9578      113070098

```

```

Memory_target=4046+ 2000=6046 => 6044 multiple of 1024

```

11.3.1.3 Global waits

```

Buffer Busy Waits by class:
----- prompt Contention in the buffer is often indicative of the Data Base Writer
(DBWR) processes being too slow. It may be wise to consider increasing
the number of DBWR's available to the system.

** data block      : usually occur when DBWR processes are too low.
** segment header : usually occur when full table scans are happening
**                : with data loads - reschedule the data load.
** undo header    : occur if there are not enough RBS segments to support

```




```

**
** undo block      : rarely occur, caused by multiple users updating records in
**                : the same data block at a very fast rate. Increase pctfree
**                : of the table being modified.
** File header block : occur when many full table scans execute simultaneously with data loading processes
**                : When occurs on TEMP segments adding more tempfiles to the tablespace and/or rebuild
the
**                : tablespace with larger Extent Sizes can help
** Buffer Busy Waits = When an Oracle session needs to access a block in the buffer cache, but cannot
because the
**                : buffer copy of the data block is locked. Common in an I/O-bound. If you are
getting buffer busy waits
**                : on the file header block for a tempfile (datafile in a temporary tablespace) then
try increasing
**                : the "next extent" size in the temporary tablespace

```

CLASS	Total Waits	Total Time Waited
data block	3,189,166	473,905
undo block	54,875	111
undo header	30,518	85
segment header	4,972	2,249
1st level bmb	2,878	1,208
2nd level bmb	2,769	1,322
file header block	165	12
system undo header	57	1
system undo block	1	0
3rd level bmb	0	0
bitmap block	0	0
free list	0	0
unused	0	0
sort block	0	0
save undo block	0	0
save undo header	0	0
extent map	0	0
bitmap index block	0	0
sum	3,285,401	

18 rows selected.

```

Waits events
-----
** db file sequential read : A single-block read (i.e., index fetch by ROWID)
** db file scattered read  : multiple data blocks read .This statistic is NORMALLY indicating disk contention on
full table scans
** log file sync          : When a user session commits, the sessions redo information needs to be flushed from
memory
**
**                : to the redo logfile to make it permanent. At the time of commit, the user session
will post the LGWR
**                : to write the log buffer (containing the current unwritten redo, including this
session's redo
**                : information) to the redo log file. When the LGWR has finished writing, it will post
the user session
**                : to notify it that this has completed. The user session waits on 'log file sync'
while waiting for LGWR
**                : to post it back to confirm all redo changes have made it safely on to disk.
** File header block      : occur when many full table scans execute simultaneously with data loading processes
**                : When occurs on TEMP segments adding more tempfiles to the tablespace and/or rebuild
the
**                : tablespace with larger Extent Sizes can help
** Buffer Busy Waits      = When an Oracle session needs to access a block in the buffer cache, but cannot
because the
**                : buffer copy of the data block is locked. Common in an I/O-bound. If you are
getting buffer busy waits
**                : on the file header block for a tempfile (datafile in a temporary tablespace) then
try increasing
**                : the "next extent" size in the temporary tablespace

```

no rows selected

STAT_NAME	SECONDS
DB time	1668050.17
DB CPU	1451627.55
sql execute elapsed time	1435940.06
background elapsed time	187935.691
parse time elapsed	40222.2965
hard parse elapsed time	29207.9397
background cpu time	9991.75319

```

PL/SQL execution elapsed time                7746.51224
connection management call elapsed time      5712.9161
hard parse (sharing criteria) elapsed time    2723.81245
hard parse (bind mismatch) elapsed time     613.038553
failed parse elapsed time                    370.718768
PL/SQL compilation elapsed time              129.87081
sequence load elapsed time                   49.098084
repeated bind elapsed time                   24.939152
RMAN cpu time (backup/restore)               0
inbound PL/SQL rpc elapsed time              0
Java execution elapsed time                  0
failed parse (out of shared memory) elapsed  0
time

```

11.3.1.4 Top 10 IO consumers objects by Tablespace

can also be used to detect if audit_trail=DB instead of NONE

OWNER	OBJECT_NAME	TABLESPACE_NAME	VALUE
PLM_01	FIELDVALUE	TD_PLM_SMALL	228459754
PLM_01	FIELDVALUE	TD_PLM_SMALL	210915667
PLM_01	RICHTEXT	TD_PLM_SMALL	8015200
PLM_01	UK_FIELDVALUE_NAME_OWNER_ID	TI_PLM_SMALL	6512493
PLM_01	RFILE	TD_PLM_SMALL	5910839
WLP_01	CMN_SEC_ASSGND_OBJ_PERM_N1	INDX_LARGE	4503594
PLM_01	C0031091_UK	TI_PLM_SMALL	3157006
PLM_01	ROLETARGET	TD_PLM_SMALL	3036159
PLM_01	ENTERPRISESESSION	TD_PLM_SMALL	2479551
PLM_01	IMPORTCONTENT	TD_PLM_SMALL	2336621

11.3.2 Most used detailed information useful for performance analysis

11.3.2.1 Check if statistics are computed at regular interval

- LAST ANALYZED TABLES FOR APPLICATION USERS AND SCHEMA STATUS

OWNER	TABLE_COUNT	MIN_LAST_ANALYZED	MAX_LAST_ANALYZED	ACCOUNT_STATUS	CREATION_DATE
PLM_01	408	2014/04/16 01:12	2014/04/23 22:02	OPEN	2014/02/26 00:38
PLM_01_JMS	9	2014/03/03 22:00	2014/04/23 22:02	OPEN	2014/03/02 23:03
PLM_SAS	127	2014/02/26 22:00	2014/04/23 22:01	OPEN	2014/02/26 01:23
WLP_01	905	2014/04/20 03:03	2014/04/23 22:02	OPEN	2014/02/26 01:18

- LAST ANALYZED INDEXES FOR APPLICATION USERS

OWNER	INDEX_COUNT	MIN_LAST_ANALYZED	MAX_LAST_ANALYZED
PLM_01	990	2014/04/16 01:12	2014/04/23 22:02
PLM_01_JMS	12	2014/03/03 22:00	2014/04/23 22:00
PLM_SAS	66	2014/02/26 22:00	2014/04/23 22:01
WLP_01	2141	2014/04/20 03:03	2014/04/23 22:01

11.3.2.2 Schema: number of objects (missing indexes can impact performances)

Utilisateur autres que SYSTEM, SYS,OUTLN : nb d'objets

OWNER	OBJECT_TYPE	nb objet dans DBA_OBJECTS
PLM_01	INDEX	989
	LOB	1
	PACKAGE	2
	PACKAGE BODY	2
	PROCEDURE	4
	SEQUENCE	4
	TABLE	408
	TRIGGER	2
	VIEW	12
PLM_01_JMS	INDEX	10

	LOB	2
	TABLE	9
PLM_SAS	DATABASE LINK	4
	FUNCTION	8
	INDEX	65
	LOB	17
	SEQUENCE	2
	TABLE	127
PUBLIC	SYNONYM	24194
WLP_01	FUNCTION	105
	INDEX	2029
	LOB	112
	PACKAGE	157
	PACKAGE BODY	36
	PROCEDURE	473
	SEQUENCE	442
	TABLE	905
	TRIGGER	85
	TYPE	7
	VIEW	556

11.3.2.3 Count rows in each schemas: check bigger tables

Current data sizing: bigger tables list extract

OWNER	NOMTABLE	TABLESPACE	NB_LIG	CHAIN_CNT	LAST_ANALY	PFREE	FLIST
	ABSTRACTIMAGE	TD_PLM_SMALL	2365633	0	2014-04-16		10
	AXEVALUE	TD_PLM_SMALL	273686	0	2014-04-23		10
	AXEVALUE_AXE_VALUES_SKUS	TD_PLM_SMALL	387416	0	2014-04-22		10
	BREAKDOWN	TD_PLM_SMALL	660136	0	2014-04-23		10
	BREAKDOWN_BRKD_AS_CONTAINE	TD_PLM_SMALL	624427	0	2014-04-22		10
	CHANGEDescription	TD_PLM_SMALL	111487	0	2014-04-22		10
	CLASSIFIABLE_CLASSIFIABLE_	TD_PLM_SMALL	216286	0	2014-04-22		10
	CONTENTMAPPER	TD_PLM_SMALL	2671663	0	2014-04-16		10
	ENTERPRISESESSION	TD_PLM_SMALL	372782	0	2014-04-23		10
	FIELDDEF	TD_PLM_SMALL	20838	0	2014-04-16		10
	FIELDVALUE	TD_PLM_SMALL	2494337	0	2014-04-22		10
	FULLIMAGE	TD_PLM_SMALL	296568	0	2014-04-16		10
	IMAGESET	TD_PLM_SMALL	295975	0	2014-04-16		10
	IMG	TD_PLM_SMALL	2366023	0	2014-04-16		10
	IMPORTCONTENT	TD_PLM_SMALL	2362508	0	2014-04-16		10
	PROD	TD_PLM_SMALL	55465	0	2014-04-23		10
	PRODUCT_CUSTOMROLE	TD_PLM_SMALL	281553	0	2014-04-22		10
	PRP	TD_PLM_SMALL	87043	0	2014-04-16		10
	RFILE	TD_PLM_SMALL	320091	0	2014-04-23		10
	RICHTEXT	TD_PLM_SMALL	233373	0	2014-04-22		10
	ROLETARGET	TD_PLM_SMALL	1312622	0	2014-04-23		10
	ROLETARGET_OWNER_TARGETS	TD_PLM_SMALL	1296113	0	2014-04-23		10
...							
	THUMBNAILIMAGE	TD_PLM_SMALL	2067465	0	2014-04-16		10
	TRACKING	TD_PLM_SMALL	354409	0	2014-04-23		10

11.3.2.4 Most consumers statements extract: top 10 by type (elapse, CPU, exec, IO, get, parsing...)

12. APPENDIX: TYPICAL ORACLE AWR SNAPSHOT ANALYSIS

12.1 Requirements

AWR Snapshots reports done using database bundle scripts during final user loss of service

12.1.1 Check snapshot interval match with final user loss of service

This can be a short period or a full day with performance problems

	Snap Id	Snap Time	Sessions	Cursors/Session
Begin Snap:	24343	17-Mar-14 09:00:13	82	5.8
End Snap:	24448	21-Mar-14 18:00:17	109	11.8
Elapsed:		6,300.06 (mins)		
DB Time:		13,256.08 (mins)		

12.2 Step1: get a global vision of database server Instance/OS resource usage and limit

12.2.1 Objective: Check what kind of “wait class”: DB CPU/ DB IO or other?

CPU in the TOP DB usage is a normal behavior because index usage use CPU

You just need to check that the database server CPU limit is not reached using tool like Oracle database console or other

12.2.2 IO: how to detect abnormal IO usage and waits

12.2.2.1 Check Top foreground wait class

Wait Class	Waits	%Time -outs	Total Wait Time (s)	Avg wait (ms)	%DB time
User I/O	2,887,629	0	10,697	4	76.13
DB CPU			3,190		22.70
Application	8,992	0	63	7	0.45
Network	9,652,756	0	19	0	0.14
Concurrency	832	43	5	6	0.04

IO problem (IO sub-system limit reached) can be caused by:

- Most of the time: Bad statement (not optimized)
- Bad application usage
- Real IO sub-system problem

12.2.2.2 Check IO speed (Slow if IO avg read >=10 ms)

Example on 32 bits of huge IO and IO subsystem reach the limit because a lot of TEMP usage (missing Oracle memory)

Tablespace IO Stats:

• ordered by IOs (Reads + Writes) desc

Tablespace	Reads	Av Reads/s	Av Rd(ms)	Av Blks/Rd	Writes	Av Writes/s	Buffer Waits	Av Buf Wt(ms)
TEMP03	210,516	58	24.06	16.99	196,511	55	2	5.00
TEMP01	2,541	1	239.50	18.56	66,822	19	19	840.53
TEMP02	6,003	2	253.05	20.20	52,264	15	0	0.00
TD_PLM_SMALL	18,636	5	5.52	14.29	148	0	397	11.16
USERS_SMALL	6,251	2	5.25	3.34	3,274	1	663	2.37
TI_PLM_SMALL	6,848	2	4.44	1.16	227	0	146	7.26
USERS_LARGE	5,978	2	4.56	1.15	279	0	2	5.00
INDX_SMALL	3,788	1	6.29	1.06	1,787	0	2,668	0.11
INDX_LARGE	2,181	1	5.23	1.03	689	0	8	6.25
SYSAUX	1,781	0	6.49	1.20	714	0	0	0.00
UNDOTBS1	368	0	2.20	1.00	1,983	1	224	0.09
SYSTEM	2,218	1	4.74	3.41	90	0	53	0.38
TD_PLM_JMS	69	0	5.80	1.00	54	0	0	0.00
TD_PLM_LARGE	1	0	10.00	1.00	1	0	0	0.00
TI_PLM_LARGE	1	0	60.00	1.00	1	0	0	0.00
USERS	1	0	10.00	1.00	1	0	0	0.00

12.2.2.3 Example of Intensive TEMP usage instead of memory

Top 5 Timed Foreground Events

Event	Waits	Time(s)	Avg wait (ms)	% DB time	Wait Class
direct path read temp	219,055	7,456	34	42.64	User I/O
direct path write temp	315,523	6,654	21	38.06	User I/O
enq: RO - fast object reuse	558	949	1701	5.43	Application
log file sync	79,559	894	11	5.12	Commit
DB CPU		842		4.81	

12.3 Step2: find consumers

12.3.1.1 Go to the SQL statistics link

SQL Statistics

- [SQL ordered by Elapsed Time](#)
- [SQL ordered by CPU Time](#)
- [SQL ordered by User I/O Wait Time](#)
- [SQL ordered by Gets](#)
- [SQL ordered by Reads](#)
- [SQL ordered by Physical Reads \(UnOptimized\)](#)
- [SQL ordered by Executions](#)
- [SQL ordered by Parse Calls](#)
- [SQL ordered by Sharable Memory](#)
- [SQL ordered by Version Count](#)
- [Complete List of SQL Text](#)

12.3.1.2 Example: Find most elapse consumer

Check “SQL ordered by Elapsed Time” link

Look at the column “elapsed Time by exec”

Look at number of executions

Elapsed Time (s)	Executions	Elapsed Time per Exec (s)	%Total	%CPU	%IO	SQL Id	SQL Module	SQL Text
31,051.92	20	1,552.60	56.80	34.60	0.00	9akxxk3mffvryq	JDBC Thin Client	l'load - HQL - [findByCriteri...
7,471.25	0		13.67	2.03	34.56	cx5nirkhwsm5a	JDBC Connect Client	SELECT Q.NUM_ROWS, Q.ID_ACTION...
4,861.29	188	25.86	8.89	87.75	0.00	7qrzks61qtvaq	JDBC Connect Client	SELECT * FROM NMS_MESSAGES WHE...
3,737.01	0		6.84	75.03	0.16	bq9barsv9g94b	JDBC Connect Client	SELECT PDT_MILESTONES_ID_DL, P...
3,737.01	0		6.84	75.03	0.16	cad0tphs4hqr	JDBC Connect Client	BEGIN WLP_MS01_LAUNCH_SP (:V1...
687.68	120	5.73	1.28	93.52	0.01	3jv8xb5pmfvc9	JDBC Connect Client	SELECT Q.NUM_ROWS, Q.ID_ACTION...
187.10	2,975	0.06	0.34	1.27	89.93	q7kbzypz2qz4v	JDBC Connect Client	INSERT INTO NMS_MESSAGES (ID,...
141.75	1,320	0.11	0.26	99.32	0.02	9sy0pbhbtxia8	JDBC Connect Client	SELECT Q.NUM_ROWS, Q.ID_ACTION...
128.01	1	128.01	0.23	0.27	11.51	d15cdr0zt3vtp	Oracle Enterprise Manager.Metric Engine	SELECT TO_CHAR(current_timesta...
99.15	79,681	0.00	0.18	97.47	0.00	q6s2qrqdc086	JDBC Connect Client	update cmn_sch_jobs set status...

Click on the SQL text column to get the full statement code

To be continued.

13. APPENDIX: ORACLE DATABASE CONSOLE HOW TO

13.1 Since 11GR1: How to make PERFORMANCE tab available

13.1.1 About Performance tab Oracle licensing or SQL PROFILE usage considerations

On PRODUCTION and PRE-PRODUCTION these capabilities need to be purchased via the Diagnostics Pack and Tuning Packs license which is available only on Enterprise Edition

Remarks:

- Strangely, these functionalities works also on Oracle 11GR2 SEONE and SE but since Oracle 12c this is not possible and need an EE edition to be enabled

13.1.2 11GR2 example using SQL*Plus

13.1.2.1 Temporary enable control_management_pack_access if not yet enabled

```
alter system set control_management_pack_access="DIAGNOSTIC+TUNING" scope=both;
```

13.1.3 11GR2 example using database console

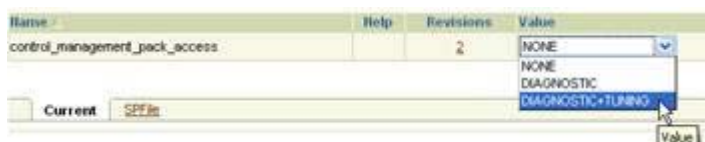
- Connect to the database console link using **SYS** user
- Go the database setup link



- Select Management Pack Access item



- Modify Pack Access and choose DIAGNOSTIC+TUNING



- If you want this definitive at last database restart check the box:
 - Apply changes in current running instance(s) mode to SPFile. For static parameters, you must restart the database.
- OK and then, APPLY
- Return to database performance TAB



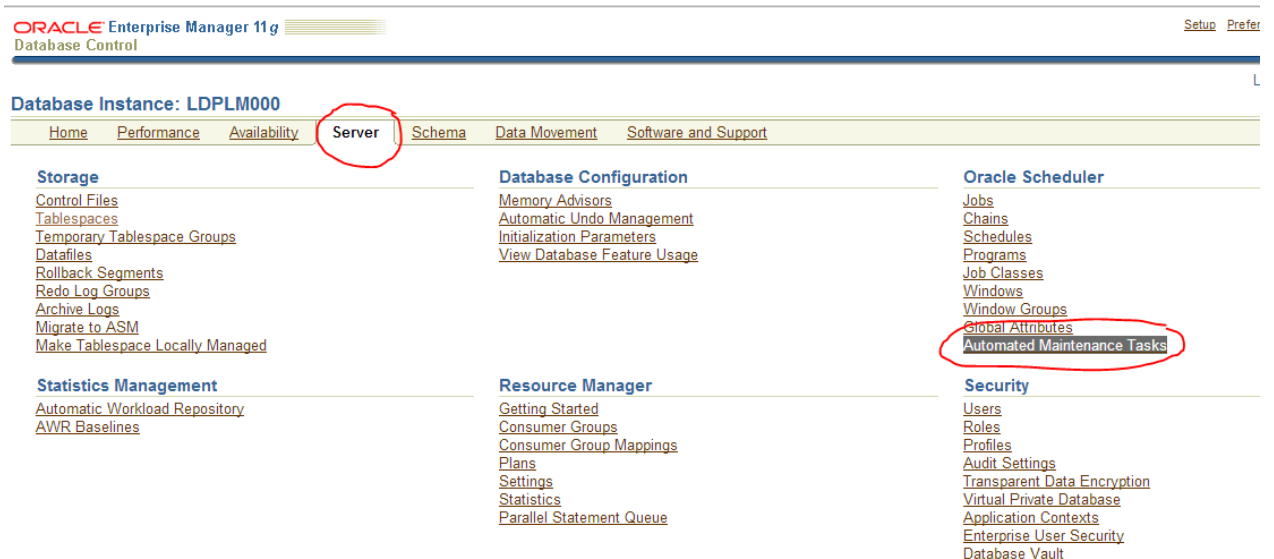
13.2 How to enable a SQL profile computed by Automatic SQL Tuning

13.2.1 Caution: Each potential SQL profile should be tested before in Pre-production and validated by the R&D

13.2.2 Steps

From database console, display Automatic SQL Tuning Result Summary view report

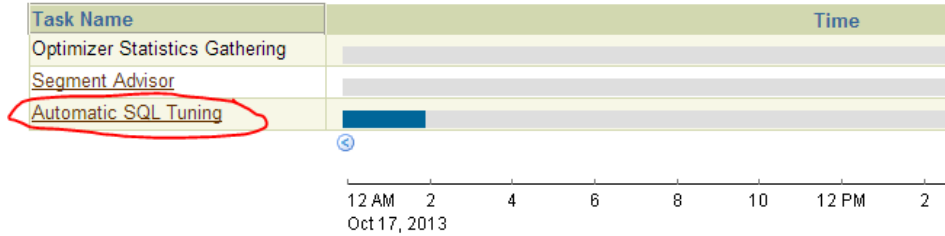
Go to Server tab, "Automated Maintenance Tasks":



Select Automatic SQL Tuning

Automated Maintenance Tasks

Status **Enabled** [Configure](#)
 TIP If the status is Disabled, there are no future windows.



Push the "View report" button:

Automatic SQL Tuning Result Summary

The Automatic SQL Tuning runs during system maintenance windows as an automated maintenance task, searching for ways to improve the execution plans of high-load SQL stat

Task Status

Automatic SQL Tuning (SYS_AUTO_SQL_TUNING_TASK) is currently **Enabled** [Configure](#)
 Automatic Implementation of SQL Profiles is currently **Disabled** [Configure](#)
 Key SQL Profiles **14** [Implement All](#)

Summary Time Period

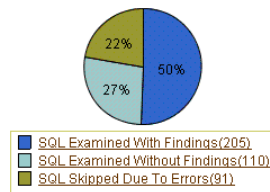
Choose a time period to focus the graphs and statistics below on a specific range of tuning results. Drill down to view focused results or see the results for all SQLs by clicking:
 Time Period [Go](#) [View Report](#)

Begin Date **Sep 30, 2013 10:00:05 PM (UTC+02:00)** End Date **Oct 17, 2013 1:08:41 PM (UTC+02:00)**

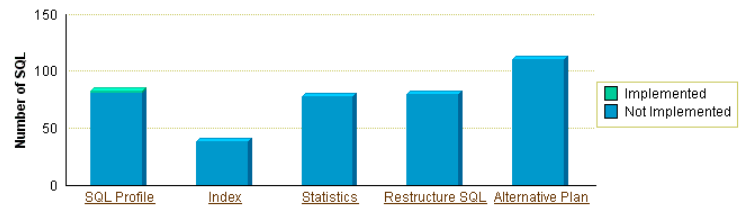
Overall Task Statistics

Executions **17** Candidate SQL **3256** Distinct SQL Examined **406**

SQL Examined Status



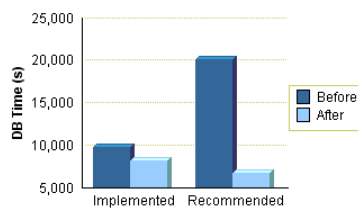
Breakdown by Finding Type



Profile Effect Statistics

Workload Potential DB Time Benefit (seconds per week)

Implemented (sec) **1506** Potential Benefit(sec) **13395**



13.2.2.1 Step: Select a statement and push the “View implementation” button

ORACLE Enterprise Manager 11g Database Control

Database Instance: LCP1M000 > Advisor Central > SQL Tuning Summary SYS_AUTO_SQL_TUNING_TASK > SQL Tuning Details SYS_AUTO_SQL_TUNING_TASK > Automatic SQL Tuning Result Details: All Analyzed SQLs

Begin Date: Sep 30, 2013 10:00:05 PM (UTC+02:00) End Date: Oct 17, 2013 1:10:32 PM (UTC+02:00) Logged in As SYSTEM

Recommendations
Only profiles that explicitly improve SQL performance were implemented.

[View Recommendations](#) [Implement All SQL Profiles](#)

Select SQL Text	Parsing Schema	SQL ID	Weekly DD Time Benefitted	Per-Execution % Benefit	Statistics	SQL Profile	Index	Restructure SQL	Alternative Plan	Miscellaneous	Timed Out	Error	Date
UPDATE WLP_MS01_ACTION_HISTORY SET LAST...	WLP_01	4fu2aovca22	15407.10	98		(98%)							10/11/2013 10:00:03 PM
SELECT CUST_TSK_MF01_GP_DURATION FROM SR...	WLP_01	0uy6vndc5d8a	5826.13	87	(<10%)	(87%)							9/30/2013 10:00:05 PM
UPDATE WLP_CS01_DL_ASSO_PRJ...	WLP_01	a28u1a8sam	3655.50	89		(89%)							9/30/2013 10:00:05 PM
SELECT COUNT(*) FROM ODF_CA_TASK_T_PRJA...	WLP_01	3r3ub2m7z1i2	3152.90	74	(74%)								10/10/2013 10:00:02 PM
SELECT NVL(ROUND(SUM(TOTAL_ESTIMATE)*10...	WLP_01	8g9m34pmb7	3144.26	97		(97%)							9/30/2013 10:00:05 PM
SELECT T_MF01_GP_DURATION FROM ODF_CA_TA...	WLP_01	3c7zq65t8tdy	3143.31	74	(74%)								10/10/2013 10:00:02 PM
SELECT COUNT(STATUS_ID) HL_STATUS FR...	WLP_01	au77qyqut8t	2890.68	97		(97%)							10/11/2013 10:00:05 PM
SELECT COUNT(*) FROM WLP_MS01_W_CAPACITY...	WLP_01	0v9m1pmb889	2328.01	99	(99%)								9/30/2013 10:00:03 PM
/* load one-to-many com lectra fo same	PLM_01	g1p2o5a89uk	2049.97	99	(99%)								10/30/2013 10:00:06 PM
/* SELECT NVL(A29 'GROUP_NAME' 'A30' 'A	WLP_01	2m2mwhd6y3b6	1691.55	50	(50%)								9/30/2013 10:00:05 PM
MERGE INTO IMP_WMPRODUCTLIFECYCLESTATEH...	WLP_01	aua2778kzqd	1226.27	36	(36%)								10/30/2013 10:00:06 PM
/* fedWMProductLifeCycleAndVMLinks	PLM_01	5d92bc1b871e0	1076.97	82	(82%)								10/10/2013 10:00:01 PM
INSERT INTO WLP_MS01_PDT_STEP_MS01_PDT...	WLP_01	caff71whpqp6	651.08	99	(99%)								10/10/2013 10:00:02 PM
/* fed3CurrentState '1 select state#	PLM_01	388ab7c1eaf6	543.63	95	(95%)								10/10/2013 10:00:01 PM
INSERT INTO IMP_WMPRODUCTLIFECYCLESTATEH...	WLP_01	prfmsc8wq2	482.70	59	(59%)								10/10/2013 10:00:04 PM
SELECT PACKIONS PROJ_ID FROM BPM_D...	WLP_01	5d9p8b82e0	348.51	99	(99%)								10/10/2013 10:00:04 PM
/* select * from (select row_number) over	WLP_01	qur65m31ndu	321.44	98	(98%)								10/10/2013 10:00:04 PM
SELECT object_id a, id...	WLP_01	11d9mep8ky	320.99	50	(50%)								10/10/2013 10:00:04 PM
SELECT PACKIONS PROJ_ID FROM BPM_D...	WLP_01	cb8b3c8a0af	302.99	99	(99%)								10/10/2013 10:00:04 PM
select (id, pd id para...	WLP_01	7m8u1cm8y7	246.90	99	(99%)								9/30/2013 10:00:05 PM
MERGE INTO IMP_WMPRODUCTLIFECYCLESTATE I...	WLP_01	Ad7yans83z	200.85	55	(55%)								9/30/2013 10:00:05 PM
DELETE FROM IMP_WMPRODUCTLIFECYCLESTATE	WLP_01	c18mcs8db0s	195.16	99	(99%)								10/30/2013 10:00:01 PM
select id, job_definition_id, j nam...	WLP_01	03ac2b8kbf	168.94	10	(10%)								9/30/2013 10:00:05 PM
INSERT INTO IMP_WMPRODUCTLIFECYCLESTATE ...	WLP_01	7d8mcs87qcc	168.75	97	(97%)								10/10/2013 10:00:01 PM
SELECT PACKIONS PROJ_ID FROM BPM_D...	WLP_01	54zqoc8d8m1	160.35	99	(99%)								10/16/2013 10:00:03 PM

If new SQL profile: click on the “compare Explain plan” link

Database Instance: LCP1M000 > Advisor Central > SQL Tuning Summary SYS_AUTO_SQL_TUNING_TASK > SQL Tuning Details SYS_AUTO_SQL_TUNING_TASK > Recommendations for SQL ID: ca1f7khwbpq45

Only one recommendation should be implemented.

SQL Text
INSERT INTO WLP_MS01_PDT_STEP_MS01_PDT_ID_DEF_STEP_ID_MILESTONE SELECT PNT_MS01_PDT_PLAN_ID_DEF_STEP_MILESTONE PROJ_ID_MILESTONE FROM BPM_DEF_STEPS_DS_WLP_MS01_PRODUCT_PLAN_PDT_PLAN_WLP_M...

Select Recommendation
[Original Explain Plan \(Annotated\)](#)

Select Type	Findings	Recommendations	Rationale	Benefit (%)	Other (%)	Benefit/Other (%)	Ratio	Explain Plan	Compare Explain Plan
SQL Profile	A potentially better execution plan was found for this statement.	Consider accepting the recommended SQL profile. No SQL profile currently exists for this recommendation.	The SQL profile was not automatically created because auto-creation was disabled. Set task parameter ACCEPT_SQL_PROFILES to TRUE to enable auto-creation.	99.82		99.82	100	100	100

If a new SQL profile: Look at IO or CPU cost gain

Original Explain Plan (Annotated)

Original Plan Value: 187299838

Recommended Plan Value: 187299838

Operation	Cost	IO	CPU
INSERT STATEMENT	18	25,368,892	18
LOAD TABLE CONVENTIONAL	17	25,368,892	18
TABLE ACCESS FULL	16	24,823,828	16
INDEX FAST FULL SCAN	4	88,887	4
INDEX UNIQUE	3	88,887	3
INDEX FAST FULL SCAN	2	88,887	2
INDEX	14	30,813,348,390	14
INDEX	12	1,312,266,782	12
INDEX	10	183,487,816	10
VIEW	9	148,847,831	9

Step: If gain is optimal then return to the previous screen and push the “implement” button

Database Instance: LCP1M000 > Advisor Central > SQL Tuning Summary SYS_AUTO_SQL_TUNING_TASK > SQL Tuning Details SYS_AUTO_SQL_TUNING_TASK > Recommendations for SQL ID: ca1f7khwbpq45

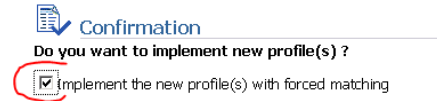
Only one recommendation should be implemented.

SQL Text
INSERT INTO WLP_MS01_PDT_STEP_MS01_PDT_ID_DEF_STEP_ID_MILESTONE SELECT PNT_MS01_PDT_PLAN_ID_DEF_STEP_MILESTONE PROJ_ID_MILESTONE FROM BPM_DEF_STEPS_DS_WLP_MS01_PRODUCT_PLAN_PDT_PLAN_WLP_M...

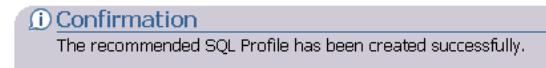
Select Recommendation
[Original Explain Plan \(Annotated\)](#)

Select Type	Findings	Recommendations	Rationale	Benefit (%)	Other (%)	Benefit/Other (%)	Ratio	Explain Plan	Compare Explain Plan
SQL Profile	A potentially better execution plan was found for this statement.	Consider accepting the recommended SQL profile. No SQL profile currently exists for this recommendation.	The SQL profile was not automatically created because auto-creation was disabled. Set task parameter ACCEPT_SQL_PROFILES to TRUE to enable auto-creation.	99.82		99.82	100	100	100

Caution: check the check box “implement the new profile with forced matching” and click yes



Wait for confirmation message:



13.2.2.2 Step: If a new database console recommend new indexes wait for R&D agreement

New indexes need to be created manually to follow Lectra naming rules best practice and set the good tablespace

13.3 How to enable a new SQL profile by running SQL tuning advisor

13.3.1 Caution:

13.3.1.1 SQL Tuning Advisor use a lot of resource: not recommended during application usage

13.3.1.2 Each potential SQL profile should be tested before in Pre-production and validated by the R&D

13.3.2 Steps:

13.3.2.1 Run your statement or choose a statement already running

13.3.2.2 Schedule a SQL Tuning Advisor from database console

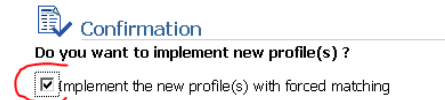
- Go to performance tab
- Click on the Top Activity link
- Select your statement using a lot of resources
- Push the button “Schedule SQL Tuning Advisor”
- Wait a few minutes for completion
- If a new profile exist and the gain is more than 50% you can test it

Example: High consumer statement called by Workflow background process



You can check the potential IO or CPU gain by pushing the “compare explain plan” button

Caution: check the check box “implement the new profile with forced matching” and click yes



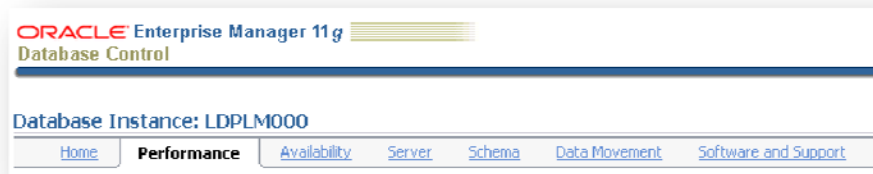
Wait for confirmation message:



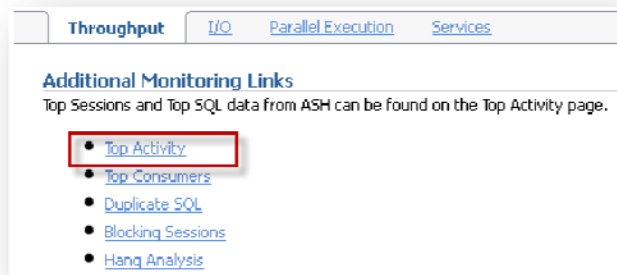
13.4 How to find and kill a big consumer impacting the PRODUCTION

13.4.1.1 Connect to the database console

13.4.1.2 Select Performance Tab



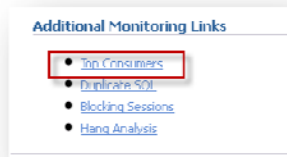
13.4.1.3 Select top activity link



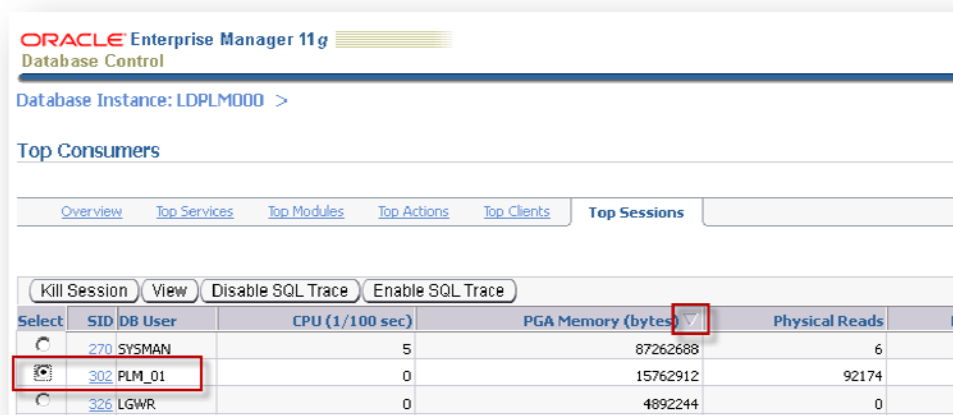
13.4.1.4 Find the most consumer session ID



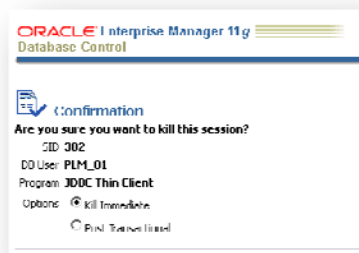
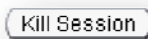
13.4.1.5 Go to Top consumer list



13.4.1.6 Order by PGA Memory and search for the Session ID



13.4.1.7 Select the session you want to kill and press Kill Session



13.4.1.8 Return to the top activity to check if your session has been killed

13.5 How to find and trace a big consumer (DEV environment)

13.5.1.1 Clean your Oracle diag log folder

Objective:

Be able to find your trace when tracing a specific transaction causing problem

Caution:

It is not recommended to do it in PROD environment because this can use a lot of resource

Steps:

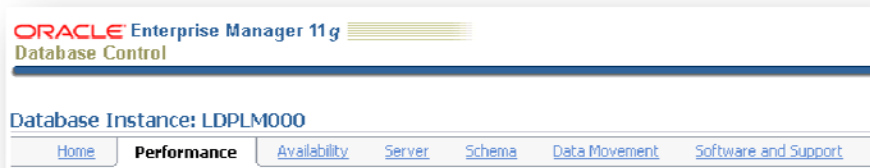
Under <drive>\app\oracle\diag\rdbms\<INSTANCE_NAME>\<INSTANCE_NAME>\trace

If needed backup Clean (or/and backup) all *.trc logs

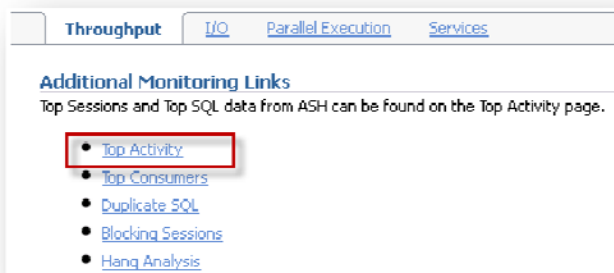
Caution: don't remove instance alert file (e.g. alert_ldplm000.log)

13.5.1.2 Connect to the database console

13.5.1.3 Select Performance Tab



13.5.1.4 Select top activity link



13.5.1.5 Find the most consumer session ID

Activity (%)	Session ID	User Name	Program
31.15	398	WLP_DBA1	SQL Developer
13.93	581	SYS	ORACLE EXE (PSP0)
9.84	2	SYS	ORACLE EXE (DIA0)
9.02	394	DBSNMP	emaagent.exe

13.5.1.6 Go to Top consumer list



13.5.1.7 Order by PGA Memory and search for the Session ID

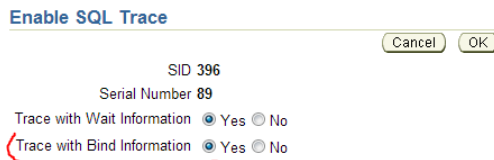
Select	SID	DB User	CPU (1/100 sec)	PGA Memory (bytes)	Physical Reads	Logical Reads	Hard Parses	Total Parses	Disk Sorts	Status	Program	Module	OS PID	Machine	OS User	SQL Trace
●	396	WLP_DBA1	1501	9097816	1	299333	0	0	0	ACTIVE	SQL Developer	SQL Developer	472	WTLOISY	t.loisy	DISABLED
○	389	DBW0	0	6181464	0	0	0	0	0	ACTIVE	ORACLE EXE (DBW0)		12692	SRDSVALBD2	SYSTEM	DISABLED
○	583	LGWR	0	5108520	0	0	0	0	0	ACTIVE	ORACLE EXE (LGWR)		12888	SRDSVALBD2	SYSTEM	DISABLED
○	199	Q001	0	4362168	0	0	0	0	0	ACTIVE	ORACLE EXE (Q001)	Streams	17672	SRDSVALBD2	SYSTEM	DISABLED

13.5.1.8 Check it is the good session with good module and machine

13.5.1.9 Enable SQL trace

Once the session selected, push the button Enable SQL Trace

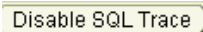
13.5.1.10 Select "Trace with bind information"



13.5.1.11 Check the SQL trace is enabled

Select	SID	DB User	CPU (1/100 sec)	PGA Memory (bytes)	Physical Reads	Logical Reads	Hard Parses	Total Parses	Disk Sorts	Status	Program	Module	OS PID	Machine	OS User	SQL Trace
●	396	WLP_DBA1	1499	9753176	0	297951	0	0	0	ACTIVE	SQL Developer	SQL Developer	472	WTLOISY	t.loisy	ENABLED
○	389	DBW0	0	6181464	0	0	0	0	0	ACTIVE	ORACLE EXE (DBW0)		12692	SRDSVALBD2	SYSTEM	DISABLED
○	583	LGWR	0	5108520	0	0	0	0	0	ACTIVE	ORACLE EXE (LGWR)		12888	SRDSVALBD2	SYSTEM	DISABLED

13.5.1.12 Caution: Once your session ended, disable the trace



Check tracing has been disabled for this session

Select	SID	DB User	CPU (1/100 sec)	PGA Memory (bytes)	Physical Reads	Logical Reads	Hard Parses	Total Parses	Disk Sorts	Status	Program	Module	OS PID	Machine	OS User	SQL Trace
○	396	WLP_DBA1	1495	9753176	7	284460	0	0	0	ACTIVE	SQL Developer	SQL Developer	472	WTLOISY	t.loisy	DISABLED
○	389	DBW0	0	6181464	0	0	0	0	0	ACTIVE	ORACLE EXE (DBW0)		12692	SRDSVALBD2	SYSTEM	DISABLED

13.5.1.1 Once your session ended check the trace under the diag folder

<drive>\app\oracle\diag\rdbms\<INSTANCE_NAME>\<INSTANCE_NAME>\trace

14. APPENDIX: DEFINITION OF COMPONENTS AND CONCEPTS

14.1.1 Oracle database server, Oracle instance

An Oracle database server consists of an Oracle database and an Oracle instance.

Every time a database is started, a system global area (SGA) is allocated and Oracle background processes are started. The combination of the background processes and memory buffers is called an Oracle instance.

Example: LDPLM000 is a database server.

An Oracle database is a collection of data treated as a unit. The purpose of a database is to store and retrieve related information.

14.1.2 Oracle schema, Oracle database user

A schema is a collection of database objects. A schema is owned by a database user and has the same name as that user.

An Oracle database can contain several schemas. Application objects like tables are created in schemas.

Example: PLM_01 (owned by LDPLM000 database server) is the schema that contains PLM tables

14.1.3 Oracle database console (Oracle Enterprise Manager)

The Web-based Database Control serves as the primary tool for managing your Oracle database

From the Oracle Enterprise Manager, you can

- Perform administrative tasks such as creating schema objects (tablespaces, tables, and indexes), managing user security, backing up and recovering your database, importing and exporting data, schedule jobs...
- View performance and status information about your database.

14.1.4 Lectra Database bundle

Set of scripts provided by Lectra to

- Install and configure
 - Oracle software ,
 - sqlnet ,
 - PLM instance and schemas
- manage PLM schemas (create, drop, migrate, export ...)
- Manage database (checks, configure, diagnostic)

Database bundle is deployed during the database part of the PLM installation process.

15. APPENDIX: PLM TYPICAL PERFORMANCE PROBLEMS

15.1 On Oracle 11GR1 only (PLM <v3)

15.1.1 PLM search SKU can use 100% CPU and is slow:

15.1.1.1 Cause: Oracle bug 6438892 (11GR1 only)

Bug 6438892 Suboptimal plan with ROWNUM predicate and NESTED LOOPS

15.1.2 Fix

Run the provided script

- <DATABASE_BUNDLE_PATH>\PLM\admin\patch\ORA_patch_11GR1_bug_6438892\
ORA_patch_11GR1_bug_6438892.cmd