Why Is My Oracle E-Biz Database Slow? A Million Dollar Question.

Ajith Narayanan
Software Development Advisor, Dell IT
NZOUG-13 , Wellington, New Zealand, March 19th

Disclaimer

The views/contents in this slides are those of the author and do not necessarily reflect that of Oracle Corporation and/or its affiliates/subsidiaries.

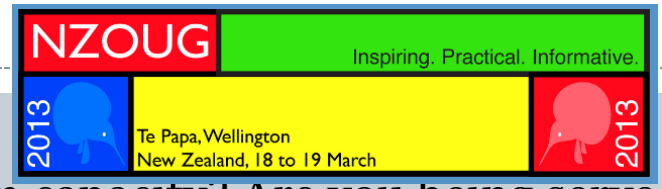
The material in this document is for informational purposes only and is published with no guarantee or warranty, express or implied..

Who Am I?

Ajith Narayanan
Software Development Advisor
Dell IT

- 8+ years of Oracle DBA experience.
- Blogger :- http://oracledbascriptsfromajith.blogspot.com
- Website Chair:- http://www.oracleracsig.org – Oracle RAC SIG

Agenda



Why system capacity? Are you being served properly?

Database Tier Analysis For Oracle E-Biz Suite

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

- ✓ What Are The Different Statistical Methods Of Analyzing The System Capacity Of E-Biz Database Tier
- ✓ Simple Math – CPU & Memory Analysis.
- ✓ Linear Regression Model for CPU.
- ✓ Queuing Theory – CPU or I/O bound system.

Middle-Tier Analysis For Oracle E-Biz Suite

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

- ✓ Did You Check Your Access. log?
- ✓ JDBC Settings?
- ✓ Concurrent Managers, Are They Sized Correctly?
- ✓ Concurrent Programs Analyzing
- ✓ JVM Settings, Are They Sized Properly? Analyze Before Concluding

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~Q&A~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Why System Capacity? Are You Being Served Properly?

How long can you wait to have your favorite dish at your favorite restaurant ? Waiter says you will have to wait for another 1 hour to be served.

Imagine a angry waiter on your table, refusing to serve you. (No Service)

Obviously, you are not happy !!

Any computer system serves you in similar fashion if not properly sized.

Capacity Planning plays a vital role in such situations.

Proper capacity analysis done by restaurant owner could have helped the waiter serve you your favorite dish faster .

Database Tier Analysis For Oracle E-Biz Suite

~~~~~~
~~~~~~

What are the different statistical methods of analyzing the system capacity of E Biz database tier?

Fundamental Forecasting Models

- **Simple Math -** This model can take single component inputs either application or technical metrics. This method is usually involved with short-duration projects, The precision is usually low, but sufficient when used appropriately.
- **Essential Forecasting Mathematics –** This method can produce relatively precise forecasts. This is again a single component input method using technical metrics. Can be used with short-duration projects.
- **Linear Regression Analysis –** This method is typically used to determine how much of some business activity can occur before the system runs out of gas.
- **Queuing Theory –** This is basically an upgrade to essential forecasting mathematics, can be used for high precision forecasting.


NZOUG
Inspiring. Practical. Informative.
2013
Te Papa, Wellington
New Zealand, 18 to 19 March
2013

SIMPLE MATH

Simple Math - CPU

Scenario:- Forecast the CPU requirements for a 4-node 16 CPU Database tier, The customer wants to downsize the infrastructure.

On Host1 , during peak utilization timings (1/5/09 3:00 AM- 4:00 AM), CPU utilization was 39.27%.
CPU consumption = 0.3927 X 14,400 s (per hour available capacity) = 5654.88s

Similarly, on Host2, Host3, Host4 observed Peak CPU consumption were 43.87% (Avg) 6173.28 s, 41.22 %(Avg) 5935.68 s and 53.11% (Avg) 7503.84s respectively.

Total CPU requirement = 25267.68 s/Hour

Calculate Utilization with 16 CPU's

~~~~~~~~~~~~~~~~~~~~~~~~

Considering 10% overhead for OS, estimated CPU utilization on 8-core (Dell 1950 QC)
Physical Server = (25267.68 + 25267.68 *10%) / (16*60*60) = 0.4825425 = **48.25%**

Calculate Utilization with 12 CPU's

~~~~~~~~~~~~~~~~~~~~~~~~

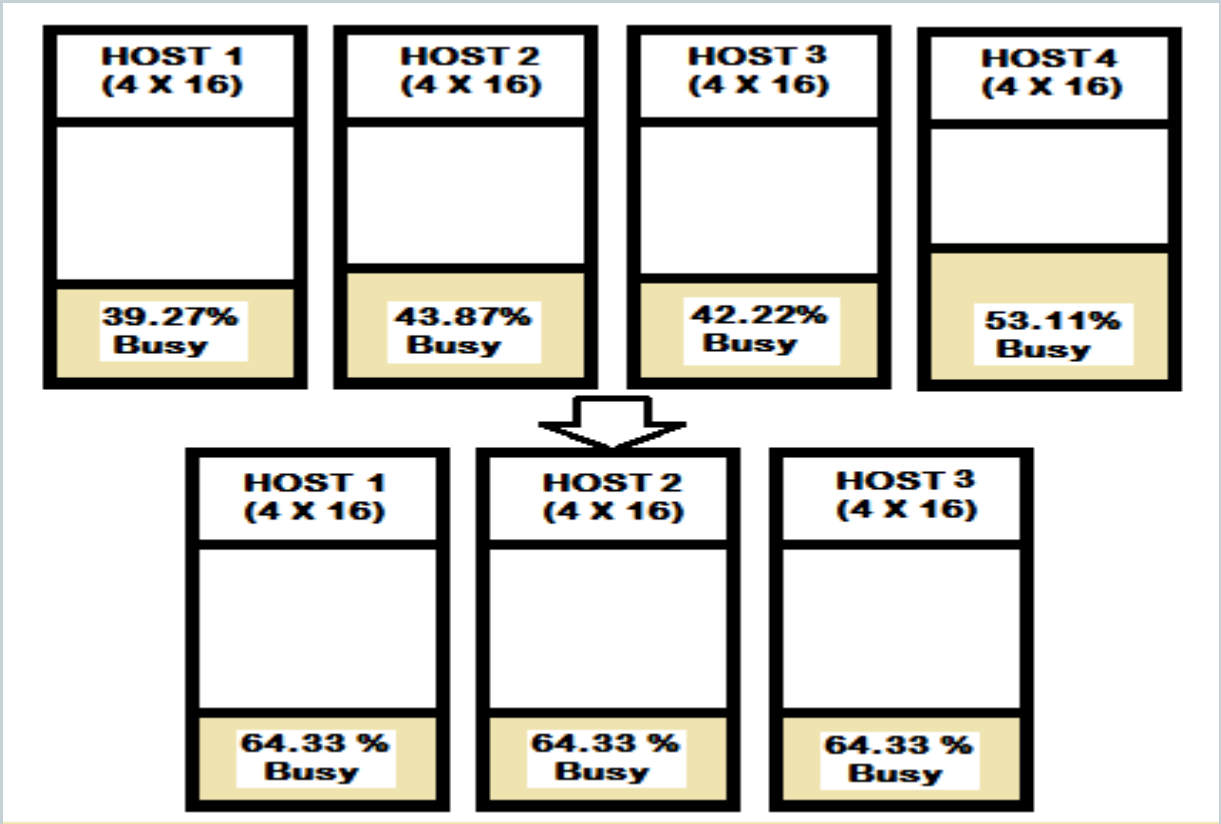
Considering 10% overhead for OS , estimated CPU utilization on 4-core (Dell 1950 QC)
Physical Server = (25267.68 + 25267.68 *10%)/ (12*60*60) = 0.64339 = **64.33%**

Calculate Utilization with 8 CPU's

~~~~~~~~~~~~~~~~~~~~~~~~

Considering 10% overhead for OS , estimated CPU utilization on 4-core (Dell 1950 QC)
Physical Server = (25267.68 + 25267.68 *10%)/ (8*60*60) = 0.87735 = **87.73%**
~~~~~~~~~~~~~~~~~~~~~~~~

Simple Math - CPU



LINEAR REGRESSION

Linear Regression Analysis

WHAT IS LINEAR REGRESSION
==========

- ✓ Linear regression analysis is a method for investigating relationships among variables. eg. Logical Rds Vs CPU utilization
- ✓ Relation is y=mx +c (equation of straight line), Here c represents the y-intercept of the line and m represents the slope.
- ✓ Here the variable "user calls" is used to predict the value of CPU Utilization .So, "user calls" becomes the explanatory variable. On the other hand, the variable whose value is to be predicted is known as response or dependent variable
- ✓ Generally response variable is denoted as Y & predicted variable denoted as X

CPU utilization = user calls * m + c

Corr coeff $r = \Sigma (y_i - y)(x_i - x) / SQRT (\Sigma (x_i - y)^2 (x_i - x)^2)$

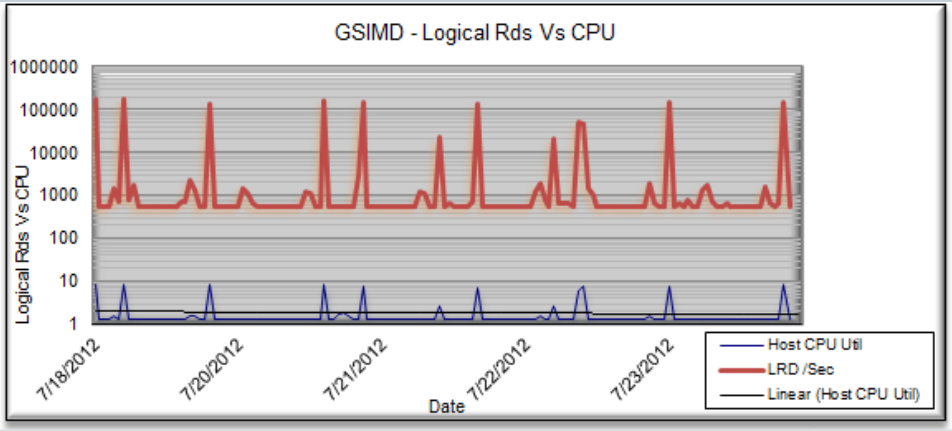
Again there is no need to solve this complex mathematical equation; Excel's predefined function CORREL () is available for us (Good news!).

Correlation Coefficient (r)	Practical Meaning
0.0 to 0.2	Very week
0.2 to 0.4	Week
0.4 to 0.7	Moderate
0.7 to 0.9	Strong
0.9 to 1.0	Very strong

Linear Regression Analysis

Scenario:- Database capacity forecasting

Co-relation table	Host CPU Util
LRD /Sec	0.98
PRD/Sec	0.56
User Calls/Sec	0.48
Commits/Sec	0.44



Capacity Forecasting for DB Node GSIMD

Metric Name	Business Hours
Avg Logical Rds per sec	10110
Avg CPU utilization	1.84

Analysis: Impact of Workload Increase on CPU utilization

Inputs	Number of CPUs	8			
	Arrival Rate	10110	Avg Logical Rds per sec		
	CPU Utilization	1.84%			
	Arrival Rate	185971.809	Max Logical Rds per sec is	1739.485252	% Of Avg workload 10110 Avg Logical Rds /Sec
Output	Service Time	0.00001455			

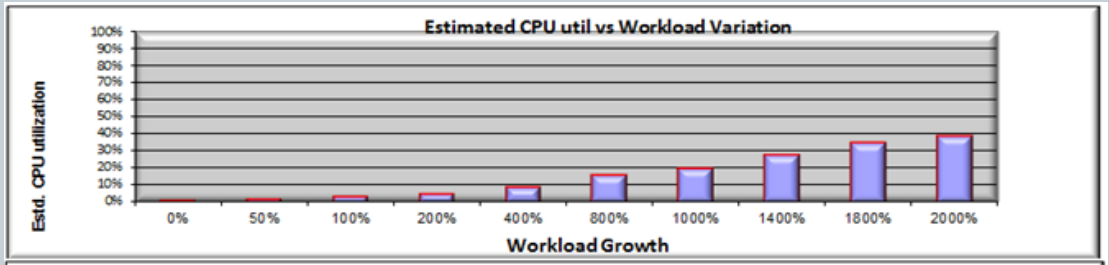
Observations:

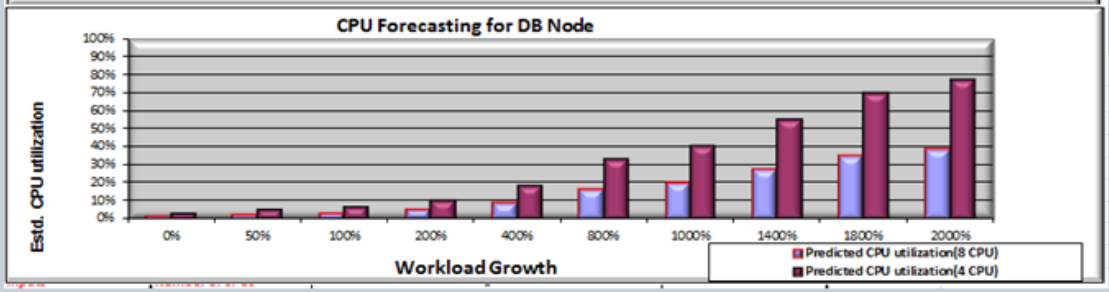
- **CPU utilization is highly correlated with Logical Reads. Correlation coefficient 0.98 in (ideal 1.0) . Both the CPU usage & Logical Reads trend line are similar. (Spikes are matching 98%)**

Linear Regression Analysis

Scenario:- Database capacity forecasting

Workload variation	Arrival rate	Avg Busy	RT	RT Change (%)	Utilization with 4 CPU's
0%	10110	1.84%	0.00001455	0.00%	3.68%
50%	15165	2.76%	0.00001455	0.00%	5.52%
100%	20220	3.68%	0.00001455	0.00%	7.36%
200%	30330	5.52%	0.00001455	0.00%	11.04%
400%	50550	9.20%	0.00001455	0.00%	18.39%
800%	90990	16.55%	0.00001455	0.00%	33.11%
1000%	111210	20.23%	0.00001455	0.00%	40.46%
1400%	151650	27.59%	0.00001455	0.00%	55.18%
1800%	192090	34.94%	0.00001456	0.02%	69.89%
2000%	212310	38.62%	0.00001456	0.03%	77.25%

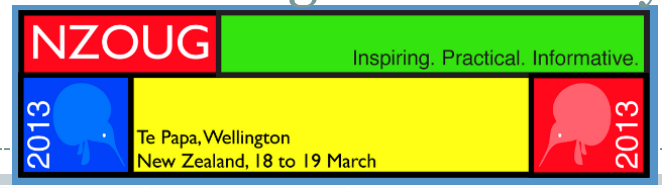




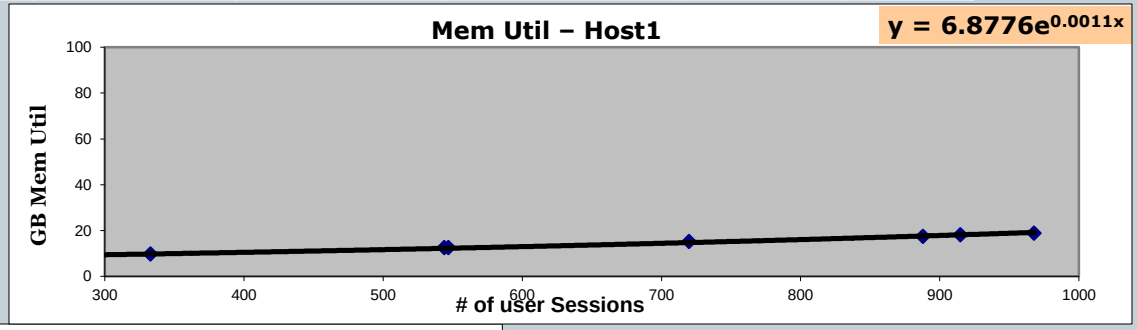
Observations:

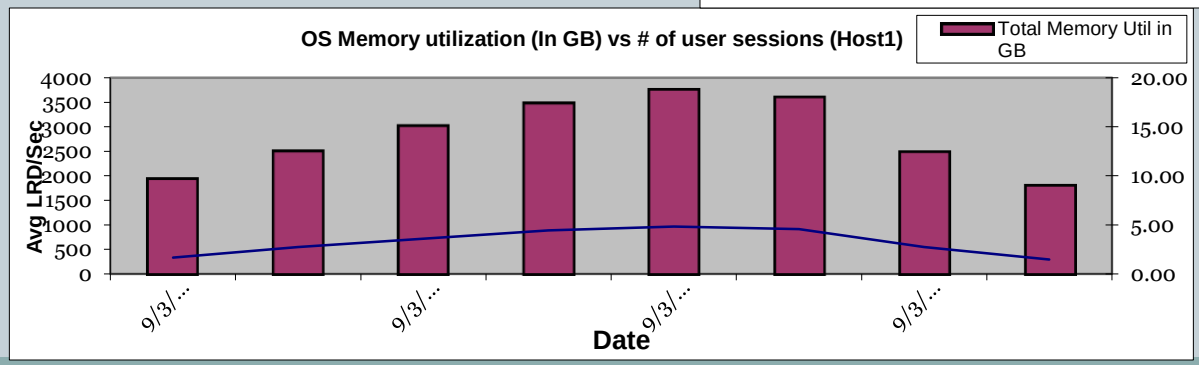
- **Current Capacity can handle additional 2000% workload by maintaining CPU utilization at 38.62% (Considering additional OS overhead etc.)**
- **Moreover, there is a downsizing possibility seen for reducing the box capacity from 8 CPU to 4 CPU (As indicated by rightmost column)**

Linear Regression Analysis



		Average	651.13		5.90	14.18	92.24%
		Max	968.00		10.08	18.83	121.14%
Instance Name	Snap ID	Date	# of User Sessions	Swap Utilization	PGA Utilization in GB	Total OS Memory utilization (In GB)	Memory Util. %
DB1	4288	9/3/2008 9:00	333	0	1.97	9.79	62.97%
DB1	4289	9/3/2008 9:30	544	0	4.45	12.58	80.92%
DB1	4290	9/3/2008 10:00	720	0	6.78	15.16	97.57%
DB1	4291	9/3/2008 10:14	888	0	8.84	17.47	112.41%
DB1	4292	9/3/2008 10:30	968	0	10.08	18.83	121.14%
DB1	4293	9/3/2008 11:00	915	0	9.42	18.09	116.40%
DB1	4294	9/3/2008 11:30	547	0	4.36	12.49	80.37%
DB1	4295	9/3/2008 12:00	294	0	1.32	9.08	58.42%





Note:- There is bug in SAR memory utilization data in Linux

Linear Regression Analysis

Memory(GB)=6.8776*EXP(0.0011*λ)

	Host1		Host1			
			Average Workload		Max Workload	
	Host1		Mem. Util in GB	% Mem. Uitl	Mem. Util in GB	% Mem. Uitl
Workload Metrics	Median	Max				
# of User Session						
	651	968.00	14.07	90.57%	19.95	128.35%

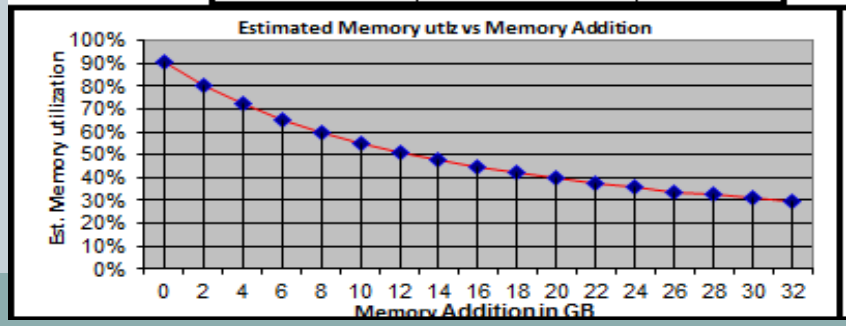
Note: Here we have to look at how much additional memory the memory utilization is coming at about 80%

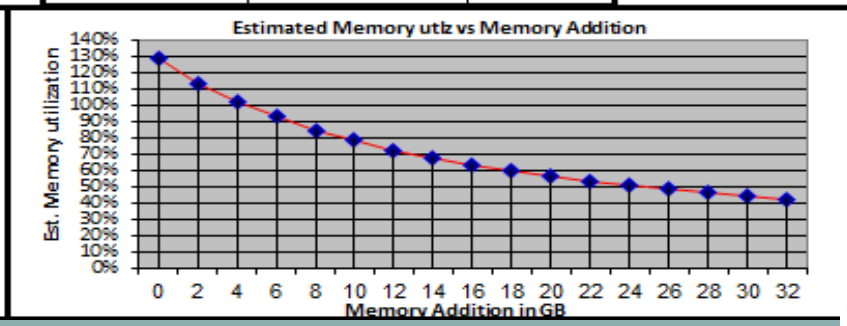
Memory required to handle Median workload

Additional Memory	Total Mem. Required	% usage
0	15.54	90.57%
2	17.54	80.24%
4	19.54	72.03%
6	21.54	65.34%
8	23.54	59.79%
10	25.54	55.11%
12	27.54	51.10%
14	29.54	47.64%
16	31.54	44.62%
18	33.54	41.96%
20	35.54	39.60%
22	37.54	37.49%
24	39.54	35.60%
26	41.54	33.88%
28	43.54	32.33%
30	45.54	30.91%
32	47.54	29.61%

Memory required to handle Max workload

Additional Memory	Total Mem. Required	% usage
0	15.54	128.35%
2	17.54	113.72%
4	19.54	102.08%
6	21.54	92.60%
8	23.54	84.73%
10	25.54	78.10%
12	27.54	72.43%
14	29.54	67.52%
16	31.54	63.24%
18	33.54	59.47%
20	35.54	56.12%
22	37.54	53.13%
24	39.54	50.45%
26	41.54	48.02%
28	43.54	45.81%
30	45.54	43.80%
32	47.54	41.96%





QUEUING THEORY

Queuing Theory

Queuing Theory is basically an upgrade to essential forecasting mathematics. This method can do a baseline of the current capacity along with its forecasting(Scalability/Downgrading possibility)

Erlang C Forecasting

==========

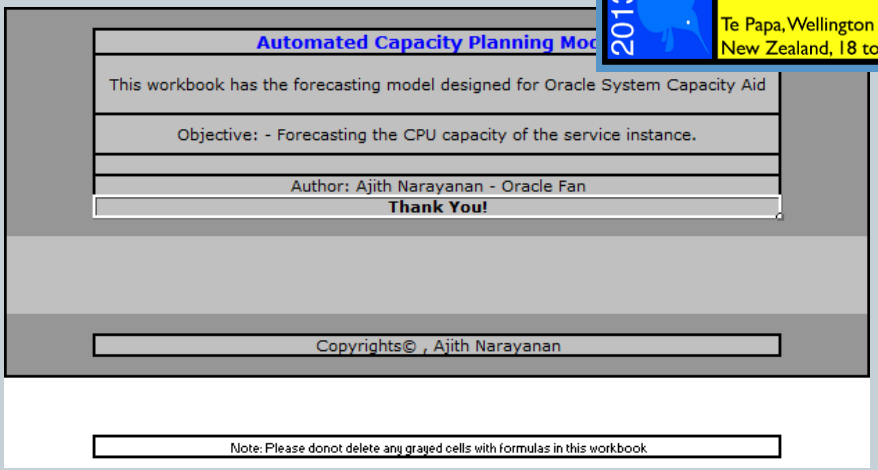
Agner Krarup Erlang(1878-1929) is the man behind this math. He studied the performance of telephone networks.

When Erlang C function is used, we do not apply the essential forecasting response time formulas, Instead, we have a single new queue time formula that can be applied to both CPU & I/O subsystem.

For CPU subsystems, there is only one queue, So the entire system arrival rate is (λ_{sys}), But for IO subsystems, the arrival rate at each queue (λ_q) is the system arrival rate(λ_{sys}) divided by the number of I/O devices.

$$U = \frac{S_t \lambda_q}{m} \qquad Q = \lambda_q Q_t \qquad E_c = \text{Erlang}(m, S_t, \lambda_q) \qquad Q_t = \frac{E_c S_t}{m(1-U)}$$

Queuing Theory



(1)RAWDATA:
Dump your raw data in this worksheet. You'll have atleast 120 rows after the data gathering for 1 hr steady state duration; The first 60 rows of data is used to develop the forecast model and the next 60 rows for model validation.

Workload Baseline	
Peak workload (90%)	71214.3
Peak workload (95%)	71496.45

Gathered Workload Data

Date	Time	System Arrival Rate (sys_lambda)	Queue Arrival Rate (lambda)	Service Time (Ts)	Queue Time (Tw)	Utilization % (Util)	Response Time (Tq)	Servers	ErlangC	Random#	Conversions: Actual Workload UserCalls/30Secs
6/11/2012	6:30:15 AM	2.176966667	2.176966667	6.291322789	1.23041	0.856	7.521728	16	0.450597414	0.6750285	65309
6/11/2012	6:30:45 AM	2.178566667	2.178566667	6.272748137	1.19589	0.8541	7.468634	16	0.445048296	0.2761636	65357
6/11/2012	6:31:15 AM	0.000266667	0.000266667	51402	10148	0.8567	61549.97	16	0.452652745	0.8039326	8

(3)AnalysisModel:
This page computes the weighted average of service time and queue time and gives the residual error by comparing it with actual service time and queue time.

Metrices	Weighted Avg	Average	Std Dev
Service Time	0.06839203	25.61587	78.61618
Queue Time	1.304223405	485.547	1345.278

Gathered Workload Data

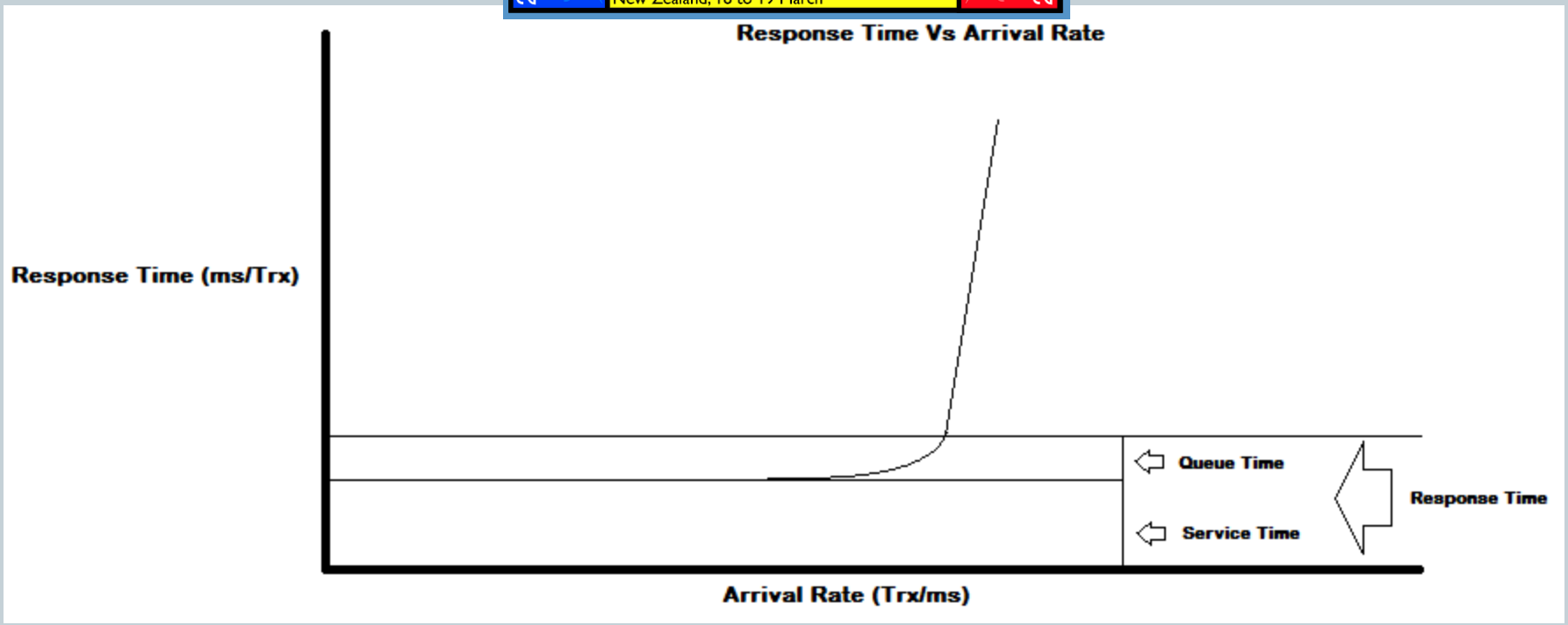
Date	Time	System Arrival Rate (sys_lambda)	Server Arrival Rate (lambda)	Service Time (Ts)	Queue Time (Tw)	Utilization % (Util)	Response Time (Tq)	Servers
6/11/2012	7:19:16 AM	0.001033	0.001033	133.8735484	2943.051321	0.86	16330.41	16
6/11/2012	6:38:45 AM	2.204533	2.204533	0.062779727	1.387759108	0.87	7.665732	16
6/11/2012	6:43:16 AM	0.000967	0.000967	141.897931	2824.156269	0.86	17013.95	16
6/11/2012	7:10:46 AM	2.330567	2.330567	0.059460217	1.334480847	0.87	7.280503	16

Residual Error Finding

Residual (Derived Ts)	Residual^2(Ts)	Residual (Derived Tw)	Residual^2(Tw)
133.8051564	17903.81987	2941.747098	8653875.989
-0.005612304	3.1498E-05	0.083535703	0.006978214
141.829539	20115.61813	2822.852046	7968493.674
-0.008931813	7.97773E-05	0.030257442	0.000915543

- Example CPU Subsystem Modelling
- Example IO Subsystem Modelling

Queuing Theory



Few More For Tips DB Tier Best Practices

- BDE Check –

bde_chk_cbo.sql generates a spool file with all Apps relevant initialization parameters according to official list on notes 216205.1 and 396009.1

- Hugepage Sizing For Database Memory Locking
- Consider Tuning Top Resource consuming SQL's (AWR), ADDM & ASH reports
- Deep Dive Into Top Wait Events (Makes Sure IDLE events, are reduced).
- Want to know what is in trace file? (Use tvdxtat_40beta9 by Christian Antognini)
- Want to have a quick review of AWR report ? (Use Tyler Muth's (http://tylermuth.wordpress.com) to make AWR reports more readable)

Middle-Tier Analysis For Oracle E-Biz Suite

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Did You Check Your Access. log?

Log To Check : $LOG_HOME/ora/10.1.3/Apache/access_log*

• Status 500 (internal server error) may typically be seen for a JServ request and often means the JVM has some kind of problem or has died.

For example: This entry may indicate that the JServ JVM is not responding to any requests:
192.168.1.10 - - [21/Jun/2006:13:25:30 +0100] "POST /oa_servlet/actions/processApplicantSearch HTTP/1.1" 500 0

• Status 403 (forbidden) could typically be seen for oprocmgr

For example: This entry in access_log may indicate a problem with system configuration (oprocmgr.conf): requests and often means there is a misconfiguration that needs to be resolved.
192.168.1.10 - - [21/Jun/2006:13:25:30 +0100] "GET /oprocmgr-service?cmd=Register&index=0&modName=JServ&grpName=OACoreGroup&port=16000 HTTP/1.1" 403 226

Run the below script to search for the above errors from access_log

Start of script
Check for HTTP statuses in 400 or 500 range for Jserv or PLSQL requests only
awk ' $9>=400 && $9<=599 { print $0 }' access_log*
grep -e "servlet" -e "\/pls\/"
grep -v .gif ## ## Check for requests taking more than 30 seconds to be returned ## awk ' $11>30 {print $0} ' access_log*
This one is not an exception report, you need to manually check , Look for when the JVMs are restarting
grep "GET /oprocmgr-service?cmd=Register" access_log*
End of script

JDBC Settings?

Change /db/appl/<contextname>/inst/apps/<context_hostname>/appl/fnd/12.0.0/secure/<context>.dbc file:

From: FND_MAX_JDBC_CONNECTIONS=500
From: FND_JDBC_BUFFER_DECAY_INTERVAL=300
From: FND_JDBC_USABLE_CHECK=false
From: FND_JDBC_BUFFER_MIN=1
From: FND_JDBC_BUFFER_MAX=5

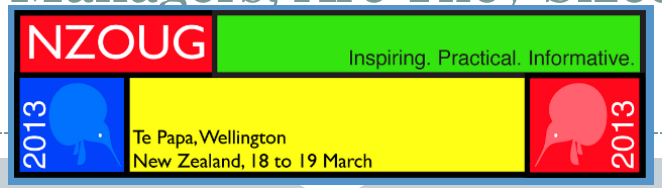
To: FND_MAX_JDBC_CONNECTIONS=100
To: FND_JDBC_BUFFER_DECAY_INTERVAL=60
To: FND_JDBC_USABLE_CHECK=true
To: FND_JDBC_BUFFER_MIN=5
To: FND_JDBC_BUFFER_MAX=50

Change /db/appl/ <contextname>/ inst/apps/<context_hostname>/appl/admin/ <context_hostname>. xml file:

From: s_fnd_max_jdbc_connections=500
From: s_fnd_jdbc_buffer_decay_interval=300
From: s_fnd_jdbc_usable_check=false
From: s_fnd_jdbc_buffermin=1
From: s_fnd_jdbc_buffermax=5

To: s_fnd_max_jdbc_connections=100
To: s_fnd_jdbc_buffer_decay_interval=60
To: s_fnd_jdbc_usable_check=true
To: s_fnd_jdbc_buffermin=5
To: s_fnd_jdbc_buffermax=50

Concurrent Managers, Are They Sized Correctly?



Thumbrule for Concurrent queue's configuration

Configuration examples of 3 "typical" queues:

Fast Queue
Sleep = 15 (seconds)
Cache Size = 10
Target = 5

Standard Queue
Sleep = 60 (seconds)
Cache Size = 20
Target = 10

Slow Queue
Sleep = 60 (seconds)
Cache Size = 10
Target = 5

Target
Make sure the number of targets (processes) don't exceed more than 20 per queue, and also remember the "rule of thumb" 3-5 processes per CPU. This rule is in most cases exceeded.

Sleep
For a queue (e.g. CUST_FAST) running fast jobs set the Sleep to 15 (seconds).
For a queue (e.g. CUST_SLOW) running slow jobs set the Sleep to 60 (seconds).
For a standard queue (e.g. CUST_STANDARD or STANDARD) set the Sleep to 60 (seconds).
For any other queues set the Sleep to 60 (seconds).

Cache Size
If Cache Size (CS) is not set, then set the cache size equal to the target value. Set CS to 2 times target value for a fast, slow and standard queue.

Concurrent Managers, Are They Sized Correctly?

Concurrent Manager Name	User Concurrent Man				CS	Proc	Total Proc/Node	Total Proc/CPU(16 CPUs)
XYZRACNODE6	XYZ Reporting Node 6 Mar				32	16		
XYZRACNODE1	XYZ Standard Node 1 Man				60	10		
FNDCRM	Conflict Resolution Manage					1	29	1.8125
FNDIM_INDXYZPRDAP0	Internal Monitor: INDXYZP					1		
FNDSM_INDXYZPRDAP0	Service Manager: INDXYZPRDAP 01		INDXYZPRDAP01	0		1		
Concurrent Manager Name	**User Concurrent Manager Name**	**Res Con Gr**	**CM Server Name**	**Sleep**	**CS**	**Proc**	**Total Proc/Node**	**Total Proc/CPU(16 CPUs)**
XYZOIMAS	XYZ Order Import Master Manag er		INDXYZPRDAP02	180	80	32		
XYZRACNODE2	XYZ Order Import Child Manage r		INDXYZPRDAP02	240	72	18		
XYZRACNODE2_APJ	XYZ Order Import Child Manage r APJ		INDXYZPRDAP02	180	72	18	70	
FNDIM_INDXYZPRDAP0	Internal Monitor: INDXYZPRDA P02		INDXYZPRDAP02	60		1		
FNDSM_INDXYZPRDAP0	Service Manager: INDXYZPRDAP 02		INDXYZPRDAP02	0		1		4.375
Concurrent Manager Name	**User Concurrent Manager Name**	**Res Con Gr**	**CM Server Name**	**Sleep**	**CS**	**Proc**	**Total Proc/Node**	**Total Proc/CPU(16 CPUs)**
XYZRACNODE3	XYZ Standard Node 3 Manager		INDXYZPRDAP03	240	64	18		
XYZHVQRM	XYZ High Volume Requests Mana ger		INDXYZPRDAP03	240	60	8		
FNDIM_INDXYZPRDAP0	Internal Monitor: INDXYZPRDA P03		INDXYZPRDAP03	60		1	29	
FNDSM_INDXYZPRDAP0	Service Manager: INDXYZPRDAP 03		INDXYZPRDAP03	0		1		
INVMGR	Inventory Manager		INDXYZPRDAP03	120	30	1		1.8125
Concurrent Manager Name	**User Concurrent Manager Name**	**Res Con Gr**	**CM Server Name**	**Sleep**	**CS**	**Proc**	**Total Proc/Node**	**Total Proc/CPU(16 CPUs)**
XYZRACNODE4	XYZ Standard Node 4 Manager		INDXYZPRDAP04	240	64	24		
XYZACP	XYZ Accounting Program Manage r		INDXYZPRDAP04	120	60	16		
XYZRMM	XYZ Receivables Master Manage r		INDXYZPRDAP04	240	32	16		
XYZXCAM	XYZ XXG Create Accounting Man ager		INDXYZPRDAP04	180	32	16		
XYZACCUSER	Accounting Jobs - User Submitt ed		INDXYZPRDAP04	300	16	8		
XYZMECRITICAL	XYZ Month End Critical Jobs		INDXYZPRDAP04	600	32	8	107	
XYZQUICKSCHDLE	XYZ Quick Schedule Jobs Manag er		INDXYZPRDAP04	10	32	8		
XYZPTJI	XYZ Purge Temporary Journal I mport		INDXYZPRDAP04	240	8	5		
XYZOPP4	XYZ OPP Node 4		INDXYZPRDAP04	60		4		
FNDIM_INDXYZPRDAP0	Internal Monitor: INDXYZPRDA P04		INDXYZPRDAP04	60		1		
FNDSM_INDXYZPRDAP0	Service Manager: INDXYZPRDAP 04		INDXYZPRDAP04	0		1		6.6875
Concurrent Manager Name	**User Concurrent Manager Name**	**Res Con Gr**	**CM Server Name**	**Sleep**	**CS**	**Proc**	**Total Proc/Node**	**Total Proc/CPU(16 CPUs)**
XYZGLCM	XYZ Generate Late Charge Mana ger		INDXYZPRDAP05	240	32	32		
XYZRACNODE5	XYZ Standard Node 5 Manager		INDXYZPRDAP05	240	64	24		
FNDIM_INDXYZPRDAP0	Internal Monitor: INDXYZPRDA P05		INDXYZPRDAP05	60		1		
FNDSM_INDXYZPRDAP0	Service Manager: INDXYZPRDAP 05		INDXYZPRDAP05	0		1	61	
WFALSNRSVC	Workflow Agent Listener Servic e	PARALLEL_0	INDXYZPRDAP05	60		1		
WFMLRSVC	Workflow Mailer Service	HIGH_PRIOR ITY_INT_GR OUP	INDXYZPRDAP05	120		1		
XXG_LIST_SERVICE	XXG List Service	HIGH_PRIOR ITY_INT_GR OUP	INDXYZPRDAP05	120		1		3.8125
Concurrent Manager Name	**User Concurrent Manager Name**	**Res Con Gr**	**CM Server Name**	**Sleep**	**CS**	**Proc**	**Total Proc/Node**	**Total Proc/CPU(16 CPUs)**
FNDICM	Internal Manager		INDXYZPRDAP06	30		1		
FNDIM_INDXYZPRDAP0	Internal Monitor: INDXYZPRDA P06		INDXYZPRDAP06	60		1	3	
FNDSM_INDXYZPRDAP0	Service Manager: INDXYZPRDAP 06		INDXYZPRDAP06	0		1		0.1875
Concurrent Manager Name	**User Concurrent Manager Name**	**Res Con Gr**	**CM Server Name**	**Sleep**	**CS**	**Proc**	**Total Proc/Node**	**Total Proc/CPU(16 CPUs)**
XYZRACNODE7	XYZ Standard Node 7 Manager		INDXYZPRDAP07	240	64	24		
STANDARD	Standard Manager		INDXYZPRDAP07	30	64	16		
XYZOPP	XYZ OPP Node 7		INDXYZPRDAP07	60		4	47	
FNDIM_INDXYZPRDAP0	Internal Monitor: INDXYZPRDA P07		INDXYZPRDAP07	60		1		
FNDSM_INDXYZPRDAP0	Service Manager: INDXYZPRDAP 07		INDXYZPRDAP07	0		1		
WFARDSRV	WF AR Distributions Agent List ener		INDXYZPRDAP07	60		1		2.9375
Concurrent Manager Name	**User Concurrent Manager Name**	**Res Con Gr**	**CM Server Name**	**Sleep**	**CS**	**Proc**	**Total Proc/Node**	**Total Proc/CPU(16 CPUs)**
XYZRACNODE9	XYZ Standard Node 9 Manager		INDXYZPRDAP08	240	80	32		
XYZRACNODE8	XYZ Standard Node 8 Manager		INDXYZPRDAP08	240	50	25		
XYZMEBIP	XYZ Month End BIP Manager		INDXYZPRDAP08	240	64	24		
XYZREPORT	XYZ Reporting BIP Manager		INDXYZPRDAP08	300	60	16		
XYZMCPCM	XYZ Customer Profile Copy Man ager		INDXYZPRDAP08	300		8		
FNDCPOPP	Output Post Processor		INDXYZPRDAP08	120		8	120	
XYZSQLREPS	XYZ Reporting SQL Manager		INDXYZPRDAP08	300	8	4		
CFODELPRINT	XYZ CFO Delivery Print Manage r	HIGH_PRIOR ITY_INT_GR OUP	INDXYZPRDAP08	300	2	1		
FNDIM_INDXYZPRDAP0	Internal Monitor: INDXYZPRDA P08		INDXYZPRDAP08	60		1		
FNDSM_INDXYZPRDAP0	Service Manager: INDXYZPRDAP 08		INDXYZPRDAP08	0		1		7.5

Concurrent Programs Analyzing

Group the concurrent programs based on the execution history

- **High Volume Jobs >1000**

List all jobs where number of submissions was 1000 or more. The list is sorted according to number of executions. Use this list to point out if a job and/or jobs that should be running in a different queue (e.g fast, slow or standard queue, cross check the slow and fast jobs table to determine this).

- **Fast Running Jobs**

All jobs where the runtime was 2 minutes or less. The list is sorted according to max runtime.
Use this list to point out if a job and/or jobs that should be running in a different queue.

- **Long Running Jobs During Peak Hour**

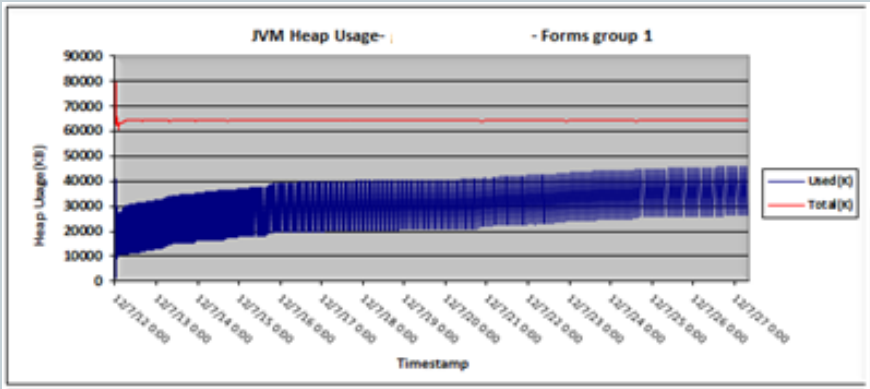
List jobs with the start date and time of the jobs where the runtime was 30 minutes or more and was submitted during the peak hours. The list is sorted according to max runtime.
Use this list to point out if a job and/or jobs could be submitted outside the peak hours.

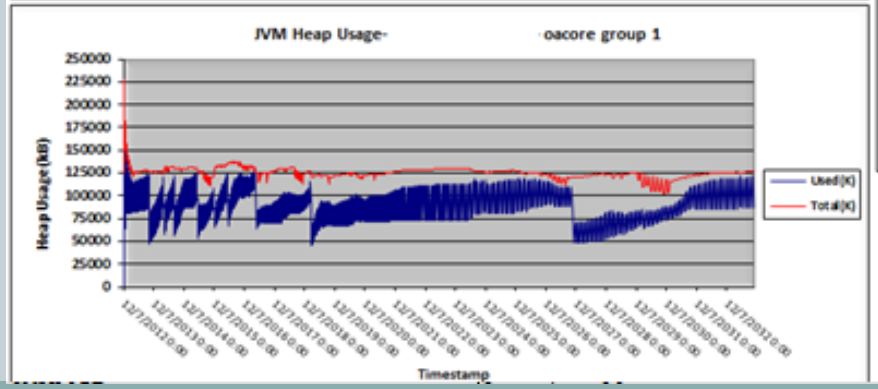
- **Check if "Purge Concurrent Request and/or Manager Data" Concurrent Program is Scheduled**

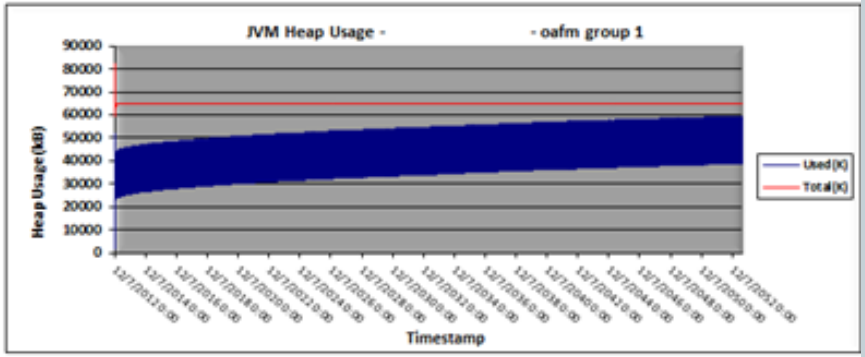
- **Concurrent Programs That Have Trace Enabled**

JVM Settings, Are They Sized Properly? Analyze Before Concluding

Node indxyzprd01	Full GC (avg) Secs	Full GC (max) Secs	Full GC (min) Secs	Count	Pause Duration (Secs)
FORMS	0.004344638	0.018386	0.002458	398	1.729166
OACORE	0.012064052	0.08705	0.001699	542	6.538716
OAFM	0.004308295	0.027928	0.002614	1056	4.549559

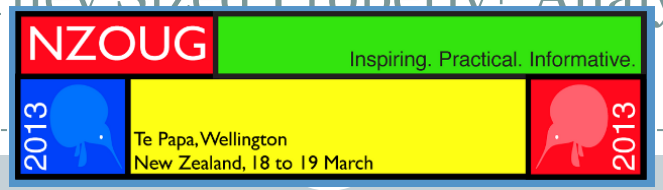






GVViewer or jvmstat

JVM Settings, Are They Sized Properly? Analyze Before Concluding



$LOG_HOME/ora/10.1.3/opmn/OC4J~oacore~default_group_*

Example:-

94562.018: [GC 670227K->595360K(892672K), 0.0221060 secs]
94617.600: [GC 672480K->617324K(799104K), 0.0307160 secs]
94648.483: [GC 694444K->623826K(872384K), 0.0405620 secs]
94706.754: [Full GC 756173K->264184K(790720K), 0.8990440 secs]
94718.575: [GC 458782K->424403K(737536K), 0.0471040 secs]
94740.380: [GC 501646K->436633K(793600K), 0.0656750 secs]
94817.197: [GC 512473K->441116K(795136K), 0.0749340 secs]

Description: -

Here the first column 94562.018, 94617.600 show the time in seconds when GC happened. Inside the square bracket it indicates whether it's a minor GC or FULL GC. That is followed by some number 670227K->595360K. The number on left side of -> indicate original size of live objects before GC and number after -> indicate size of live objects after GC. Number in the bracket (892672K) indicates total size of live objects allocated. Number after comma indicates time it took to complete garbage collection. For example in the first rows it took 0.0221060 secs for completing GC.

Review the frequency of collections; especially major collections (i.e. Full GC)

Recommendations to be given

a) Enable verbose GC to tune heap sizes based on the GC traffic
b) If full GCs are too frequent, consider increasing -Xms and -Xmx GC tuning
c) Bigger heaps => GC will take longer
d) Longer GCs => users may experience pauses
e) For the OACoreGroup JVMs start with the lower of the following two values:

Number of cores on the middle tier
Peak Concurrent users / 100

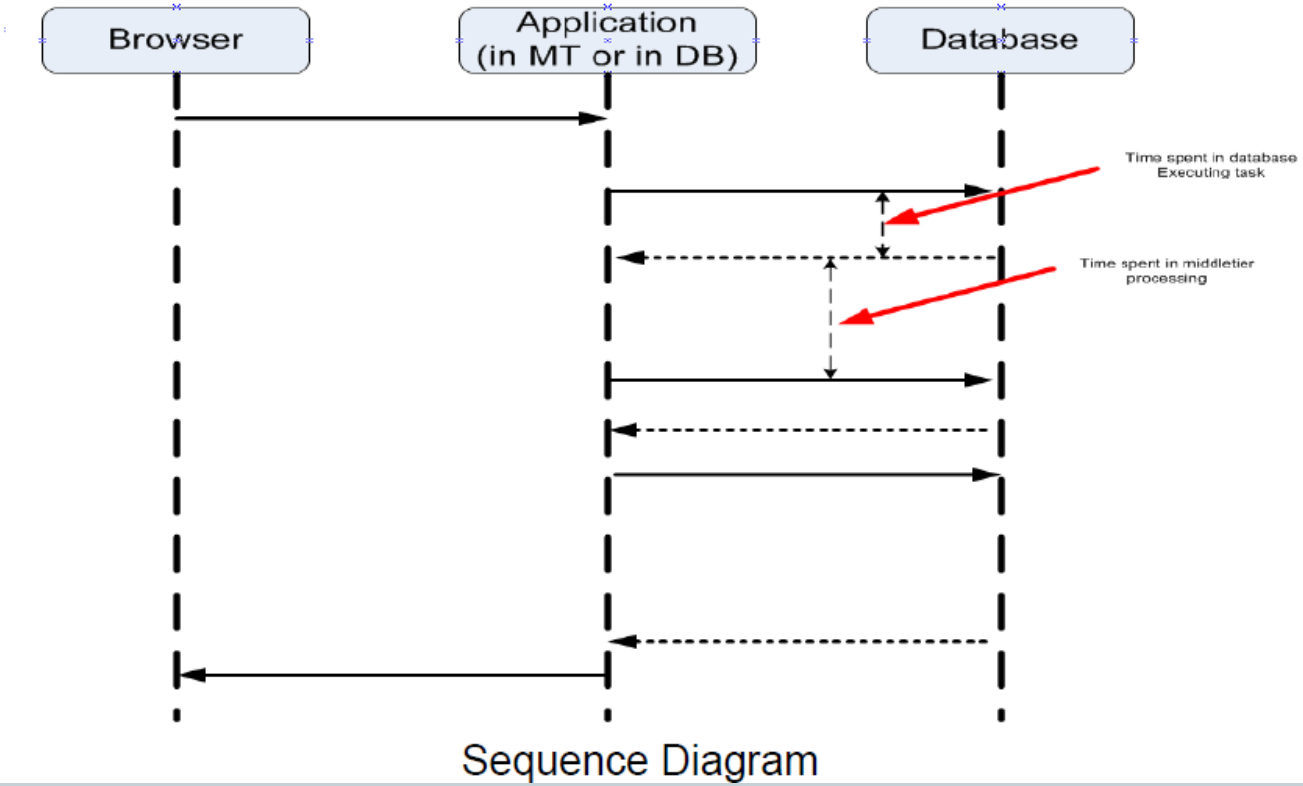
Note:- If you have 2 x Dual Core CPUs in your server and have 500 peak users, then 4 JVMs is the recommended starting point, since the number of cores is the lower number. However, if you only had 300 peak users, then you would configure 3 JVMs as a starting point as this is now the lower figure. Size your maximum heap size as either 512 MB or 1 GB. If you start with 512 MB and find that more memory is required, increase to a maximum of 1 GB. If more memory than 1 GB is required, then add another JVM instead (free physical memory allowing) to increase the total memory available overall.

Few More For Tips DB Tier Best Practices

- Start with Best Practices : (note: 1121043.1)
- SQL Tuning
  - Trace files
  - SQLT output (note: 215187.1)
  - Trace Analyzer (note: 224270.1)
  - AWR Report (note: 748642.1)
  - AWR SQL Report (awrsqrpt.sql)
  - 11g SQL Monitoring
  - SQL Tuning Advisor
- PL/SQL Tuning
  - Product logs
  - PL/SQL Profiler (note: 808005.1)
- Reports Tracing
  - note: 111311.1
- Database Tuning
  - AWR Report (note: 748642.1)
  - ADDM report (note: 250655.1)
  - Automated Session History (ASH) Report
  - LTOM output (note: 352363.1)
- Forms Tuning
  - Forms Tracing (note: 373548.1)
  - FRD Log (note: 445166.1)
  - Generic note: 438652.1
- Middletier Tuning
  - JVM Logs
  - JVM Sizing (note: 362851.1)
  - JDBC Tuning (note: 278868.1)
- OS
  - OSWatcher (note: 301137.1)

What Did We Try To Do Till Now?

Define the problem



Sequence Diagram

Create An Impact With Capacity Analysis

So let's see an example of 4-cpu Intel boxes and put them together in a cluster with Oracle11G and RAC on top:

Price for the hardware: About US$15,000 or so.

Price for the OS (Linux): About US$ 0.5- or thereabout (it depends!)

Price for Oracle w/ RAC: US$480,000,-

So that's half a million to Oracle. Put another way: It's 1 dollar to the box movers for every 32 dollars paid for Oracle RAC.

Psychologically it's hard for the customers to understand that they have to buy something that expensive to run on such cheap hardware. The gap is too big, and Oracle will need to address it soon.

There's nothing like RAC on the market, but that doesn't mean you have to buy RAC. I usually joke that it's like buying a car for US$10.000,- that has all the facilities you need from a good and stable car.

Airbags and ABS brakes are US$500.000,- extra, by the way. Well, airbags and ABS are wonderful to have and they increase your security. But it's a lot of money compared to the basic car price.


NZOUG
Inspiring. Practical. Informative.
2013
Te Papa, Wellington
New Zealand, 18 to 19 March
2013

Q&A

References

References

1. Forecasting Oracle Performance by Craig (Apress, 2007)
2. http://blogs.oracle.com/stevenChan/Shallahamer

Thank You