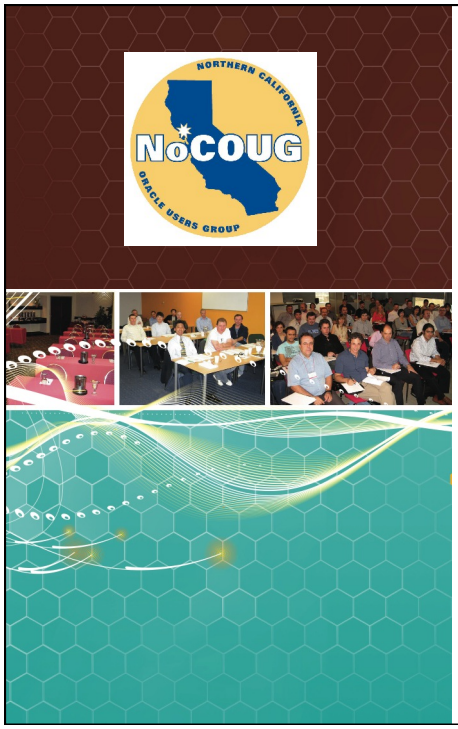


Creative Wait Interface Maneuvers:

Fast Performance Problem Resolution

Craig A. Shallahamer
OraPub, Inc.
craig@orapub.com

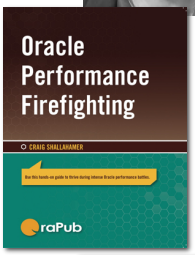
This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

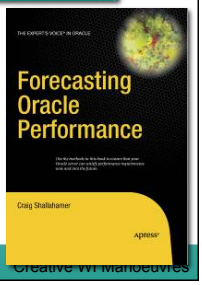
Agenda

- What is “wait time”?
- Can we really trust it?
- How Oracle gathers wait time
  - time is unknown
  - time is known
- Wait time and the OS administrator
- Creative ways to use wait time.
- Using a time based analysis to bring forth solutions.

Who Am I?

- Studied economics, mathematics and computer science at Cal Polytechnic State University San Luis Obispo, California, USA.
- Started working with Oracle technology in 1989 as a Forms 2.3 developer on Oracle version 5.
- Soon after started performance firefighting daily.
- Co-founded both Oracle's Core Technology and System Performance Groups.
- Left Oracle to start OraPub, Inc. in 1998.
- Authored 24+ technical papers and worked in 31 countries.
- Author two books: Oracle Performance Firefighting and Forecasting Oracle Performance.
- Teaches performance analysis around the world.
- Oracle ACE Director.
- Blogs performance research: A Wider View





raPub

Creative WI Manoeuvres

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

raPub

focusing exclusively on Oracle systems performance analysis

stori

Automated. Interactive.
Oracle SE Performance Analysis

www.storifree.com
www.stori.orapub.com

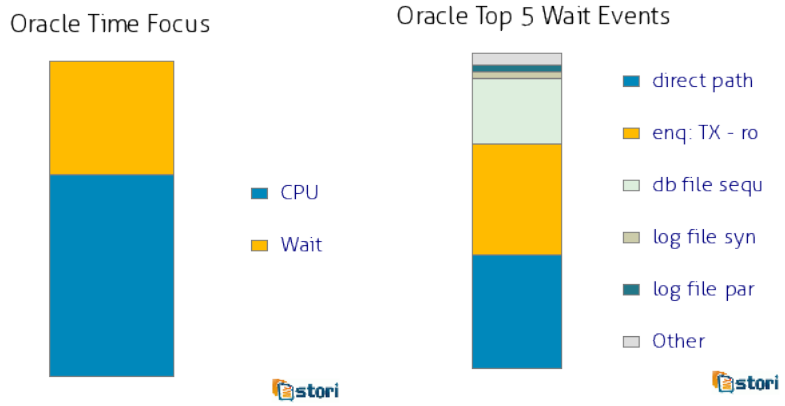
Resources

- Research Blog
- Free Tools
- Free Presentations
- Free Papers
- Books
- Consulting
- Training

raPub

Creative WI Manoeuvres

What is “behind” the wait time?



Sure, we can see the wait times but how does Oracle come up with the wait time and is it reliable?

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

It’s a lot about instrumentation.

Instrumentation is the branch of science that deals with measurement and control in order to increase efficiency.

Oracle’s kernel code instrumentation allows us to better understand where Oracle processes spend their time.

We could instrument our workday by keeping records of where we spent our time. The result would be a time card. We can ask Oracle for its “time card.”

Source code instrumentation.

- Instrumented source code has time-gathering functions placed into the source code enabling an understanding of where time is spent.
- It can enable fast and accurate diagnosis.
- Yes, there is overhead but without instrumentation you have a limited understanding of where the performance problems reside.
- Sampling is another option; less disruptive but cannot provide completely accurate information and sometimes cannot provide information (e.g., occurrences of some event)
- There are different levels/types of instrumentation.

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

This provides clues to improve performance.

```
SQL> l
  1  select   time_type,
  2           count(*) r,
  3           sum(time_spent) tt,
  4           avg(time_spent) at
  5  from     instrumentation
  6* group by time_type
SQL> /

Resource  Runs Tot Time Avg Time
--------- ---- -------- --------
cpu         21   21.98     1.05
ior          7    4.06     0.58
iow          6   29.57     4.93
latch_get   12    2.67     0.22
```

There is actually a "family" of core wait event views.

- **v$event_name.** Lists all Oracle wait events including their Oracle assigned classification.
- **v$system_event**. High-level view of all wait events.
- **v$session_event**. A high-level *session* specific view of wait events. Time is updated when a new event is waited upon.
- **v$session_wait** and **v$session (10g+)**. A low-level session specific view of wait events. Time may be updated only every three seconds.
- **v$event_histogram (10g+).** Storing wait times occurrences by event, a better understanding of wait times is possible. Waits times are skewed from their average.

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

At the instance level.

```
SQL> @swpctx
Remember: This report must be run twice so both the initial and
final values are available. If no output, press ENTER twice.

Database: prod3                                        28-FEB-12 04:48pm
Report:   swpctx.sql            OSM by OraPub, Inc.              Page 1
               System Event CHANGE (blue line) Activity By PERCENT

                               Time Waited  % Time  Avg Time      Wait
Wait Event                           (sec)  Waited Waited(ms) Count(k)
------------------------------ ----------- -------- ---------- --------
db file scattered read              57.090   52.69        2.8        20
read by other session               41.150   37.98       14.1         3
latch: cache buffers chains          1.340    1.24        5.4         0
control file parallel write          0.770    0.71       70.0         0
log file sync                        0.660    0.61      110.0         0
log file parallel write              0.490    0.45       49.0         0
db file sequential read              0.090    0.08        8.2         0
latch: cache buffers lru chain       0.020    0.02        0.6         0
latch free                           0.000    0.00        0.0         0
db file parallel write               0.000    0.00        0.0         0
direct path write                    0.000    0.00        0.0         0
```

OSM: swpctx.sql

Now the bad news

Problem:
You're at a disadvantage when smart OS vendors or administrators challenge your diagnosis that is highly wait time based.

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

Now the bad news

Problem:
You're at a disadvantage when smart OS vendors or administrators challenge your diagnosis that is highly wait time based.

Solution:
Learn how Oracle determines wait time with such confidence you can use it in creative and non traditional ways.

The problems...

DBA to IO Admin: Oracle's IO reads are taking around 25 ms. We are working on tuning Oracle and tuning the application SQL, but we need the IO subsystem team to help reduce the IO times to less than 10 ms.

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

The problems...

DBA to IO Admin: Oracle's IO reads are taking around 25 ms. We are working on tuning Oracle and tuning the application SQL, but we need the IO subsystem team to help reduce the IO times to less than 10 ms.

IO Admin to DBA: When I look at the IO times, I don't see many 25 ms times or even times longer than 25 ms. And where did you get the 25 ms figure anyways?

The problems...

DBA to IO Admin: Oracle's IO reads are taking around 25 ms. We are working on tuning Oracle and tuning the application SQL, but we need the IO subsystem team to help reduce the IO times to less than 10 ms.

IO Admin to DBA: When I look at the IO times, I don't see many 25 ms times or even times longer than 25 ms. And where did you get the 25 ms figure anyways?

DBA to IO Admin: These IO time comes from Oracle's wait interface. And the average wait time for single-block reads is clearly shown to be 25 ms. I really need your help here.

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

The problems...

DBA to IO Admin: Oracle's IO reads are taking around 25 ms. We are working on tuning Oracle and tuning the application SQL, but we need the IO subsystem team to help reduce the IO times to less than 10 ms.

IO Admin to DBA: When I look at the IO times, I don't see many 25 ms times or even times longer than 25 ms. And where did you get the 25 ms figure anyways?

DBA to IO Admin: These IO time comes from Oracle's wait interface. And the average wait time for single-block reads is clearly shown to be 25 ms. I really need your help here.

IO Admin to DBA: Oh. The timing details came from Oracle? Well, I would expect them to be inflated with all the code between the OS and what you see.

(c)2013 OraPub, Inc.

Creative WI Manoeuvres

The problems...

DBA to IO Admin: Oracle's IO reads are taking around 25 ms. We are working on tuning Oracle and tuning the application SQL, but we need the IO subsystem team to help reduce the IO times to less than 10 ms.

IO Admin to DBA: When I look at the IO times, I don't see many 25 ms times or even times longer than 25 ms. And where did you get the 25 ms figure anyways?

DBA to IO Admin: These IO time comes from Oracle's wait interface. And the average wait time for single-block reads is clearly shown to be 25 ms. I really need your help here.

IO Admin to DBA: Oh. The timing details came from Oracle? Well, I would expect them to be inflated with all the code between the OS and what you see.

DBA to IO Admin: You're an idiot.

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

The problems...

DBA to IO Admin: Oracle's IO reads are taking around 25 ms. We are working on tuning Oracle and tuning the application SQL, but we need the IO subsystem team to help reduce the IO times to less than 10 ms.

IO Admin to DBA: When I look at the IO times, I don't see many 25 ms times or even times longer than 25 ms. And where did you get the 25 ms figure anyways?

DBA to IO Admin: These IO time comes from Oracle's wait interface. And the average wait time for single-block reads is clearly shown to be 25 ms. I really need your help here.

IO Admin to DBA: Oh. The timing details came from Oracle? Well, I would expect them to be inflated with all the code between the OS and what you see.

DBA to IO Admin: You're an idiot.

IO Admin to DBA: You're stupid.

(c)2013 OraPub, Inc.

Creative WI Manoeuvres

Hidden assumptions: T or F?

1. All operating system IO's are experiencing the 25 ms response time.
2. All Oracle IO's are experiencing the 25 ms response time.
3. All Oracle Single Block Read IO's are about 25 ms.
4. The typical wait time is the about same as the average wait time.
5. All Oracle IO's have associated wait time and wait count.

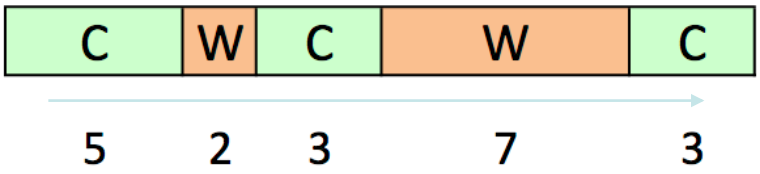
This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

The solution...

- Understand time based analysis.
- Learn how Oracle gets wait time...
- In a variety of situations.
- Be able to quickly demonstrate this.
- Understand the difference between the average and the typical wait time.
- Be able to quickly demonstrate average and typical value.
- Speak only what the statistic tells us.

Elapsed Time; What is it?

Elapsed Time = CPU Time + Wait Time




```
CPU   = 11
Wait =  9
---------
E     = 20
```

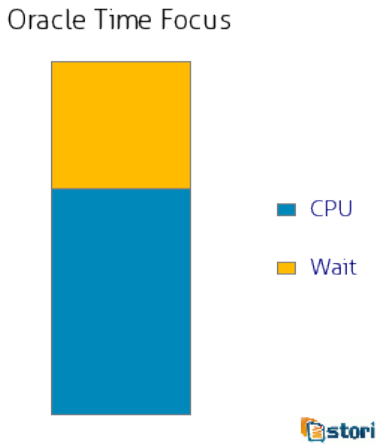
Single Session

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

The world of Time Based Analysis

- It's about time! OLTP-centric users want snappy response time and batch-centric users want short duration jobs. Therefore, our analysis and methods of communication will be more effective when time based.
- Time Based Analysis:
  - Combines wait time and CPU consumption
  - Classifies time to better understand, diagnose, and communicate the performance situation
  - Moves us into Unit of Work Time Based Analysis (UOWTBA)
  - Enables us to better understand the user experience

The Big Bar − What is in it?

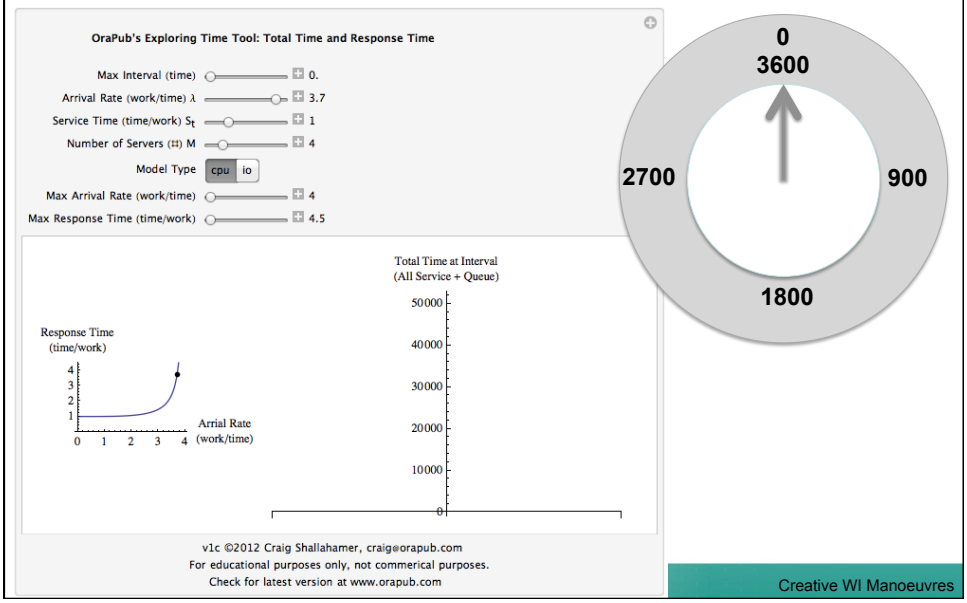


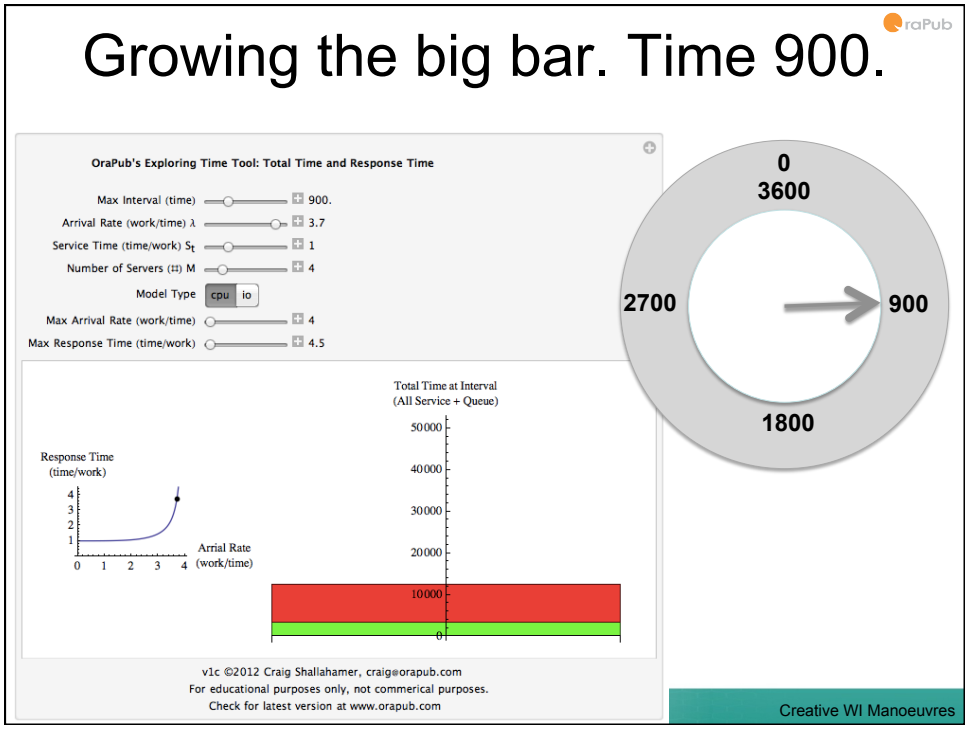
The “big bar” contains all the CPU consumption and wait time for all Oracle processes over a defined time interval.

Bar height slowly increases as Oracle performs work, one little piece at a time, consuming CPU and possibly waiting.

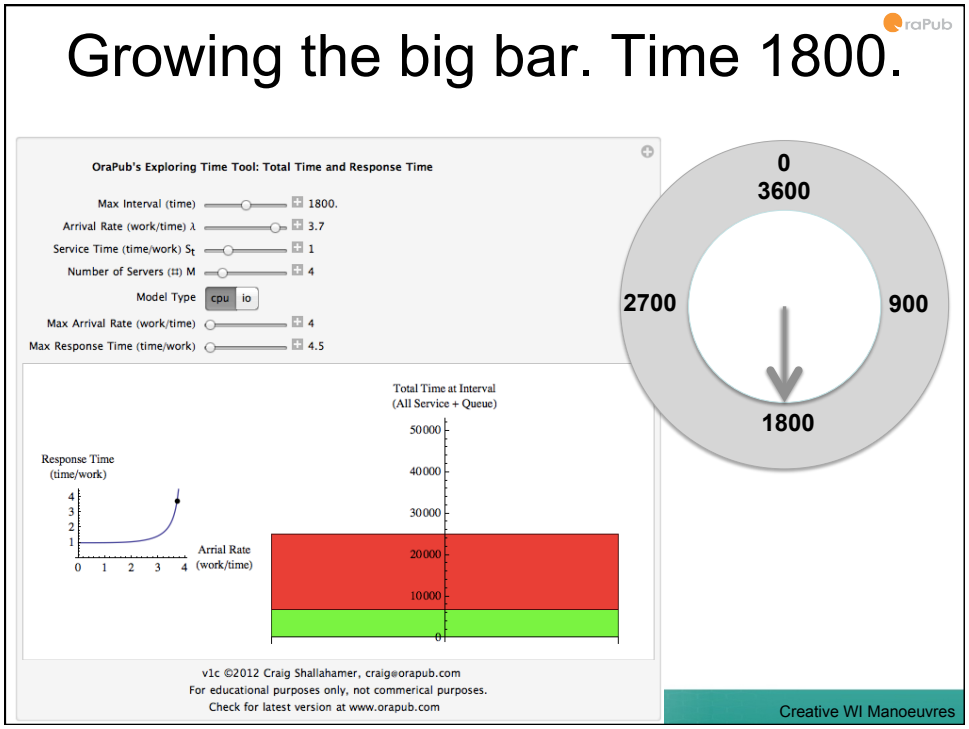
This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

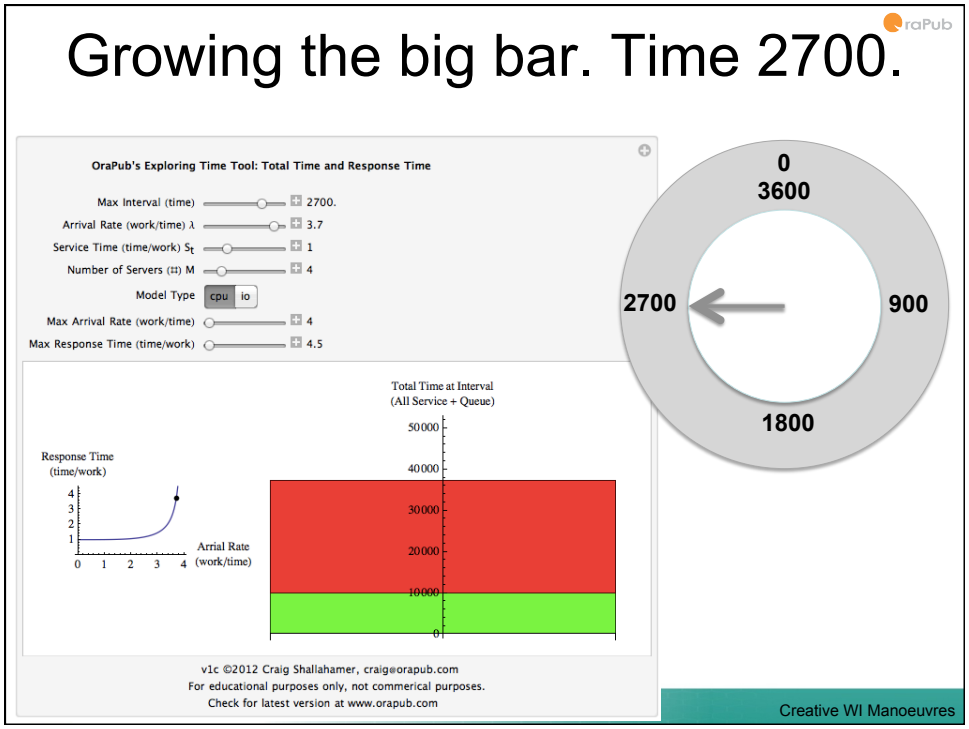
Growing the big bar. Time 0.



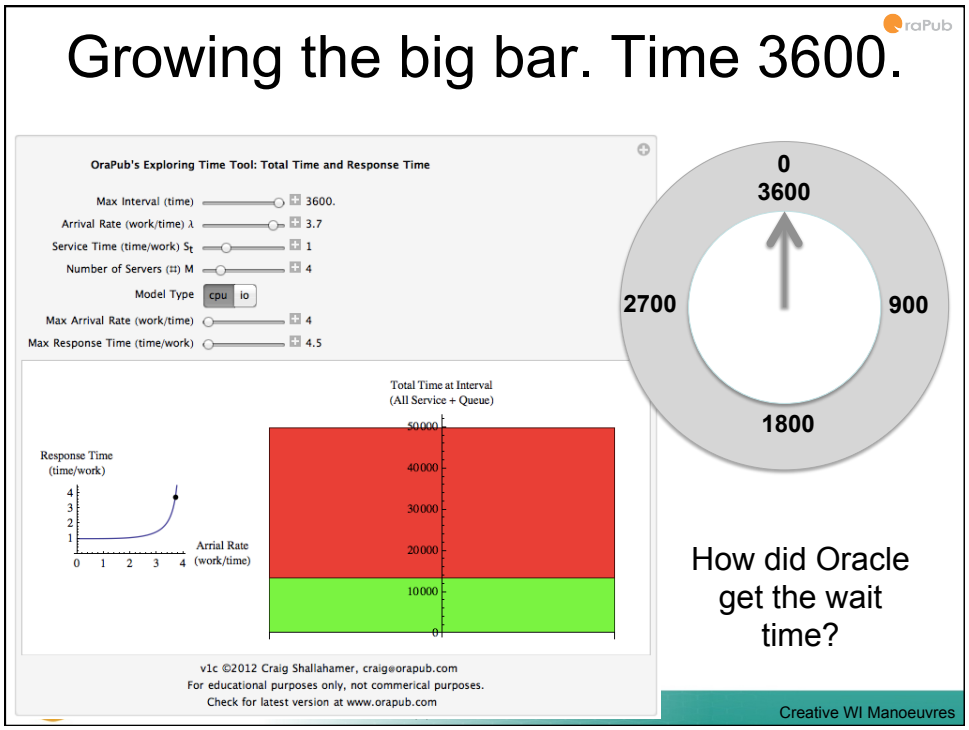


This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.





This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.



Even an SP or AWR report sets us up.

WORKLOAD REPOSITORY report for

DB Name	DB Id	Instance	Inst num	Release	RAC	Host
PROD15	3466472990	PROD15	1	10.2.0.3.0	NO	clue

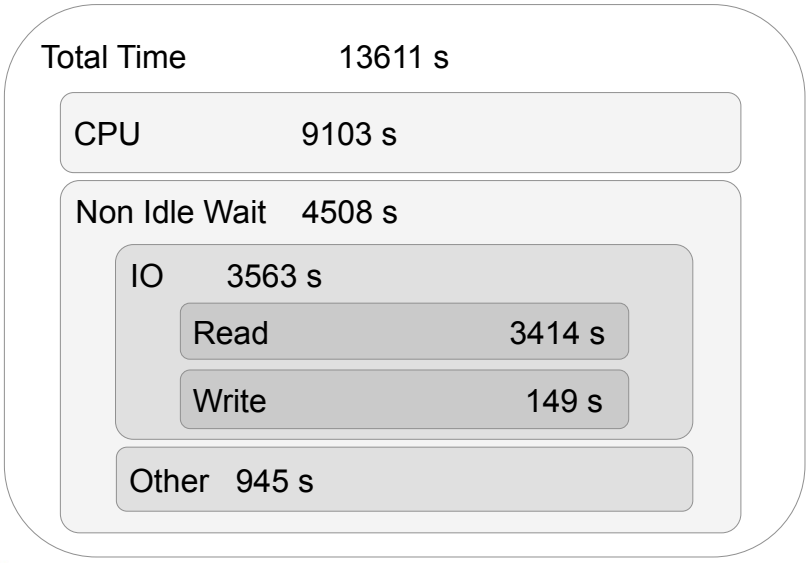
Top 5 Timed Events

Event	Waits	Time(s)	Avg Wait(ms)	% Total Call Time	Wait Class
CPU time		3,641		66.3	
db file sequential read	489,550	587	1	10.7	User I/O
db file scattered read	12,142	565	47	10.3	User I/O
direct path read temp	34,932	470	13	8.6	User I/O
log file parallel write	6,253	235	38	4.3	System I/O

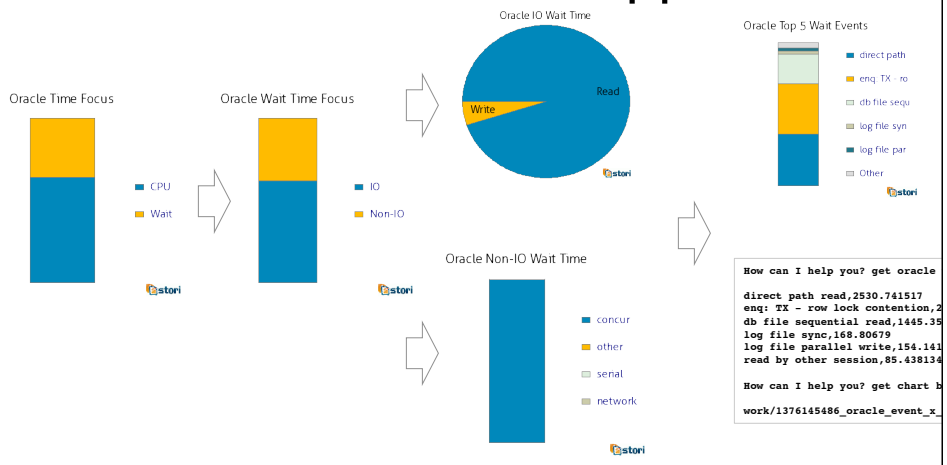
source: http://filebank.orapub.com/perf_stats/AWR_PROD15.html

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

Where to focus our work?



Watch the story develop and the solutions appear.



This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

Known and unknown wait time

- **Wait Time Unknown**. Oracle asks the OS, “What time is it?”, does what it needs to do (e.g., request a block of IO), pauses until the request is fulfilled, asks the OS, “What time is it?”, and records the time difference. This time difference becomes the wait time.
- **Wait Time Known.** Oracle set the wait time to a specific value and waits. For example, when a process sleeps during the latch acquisition process, Oracle may set the sleep time to 10 ms. This 10 ms sleep time becomes the wait time.

Wait time, unknown duration

```
[oracle@fourcore ~]$ ps -eaf|grep oracleprod18
oracle   19759 19750  1 11:10 ?        00:00:01 oracleprod18 (DESCRIPTION=. . .
. . .
[oracle@fourcore ~]$ strace -p 19759
. . .
gettimeofday({1273083131, 538145}, NULL) = 0
readv(11, [
{"\6\242\0\0\260B\17\1\25z\352\1\0\0\1\6G\32\0\0\1\0\0\0\352F\1\0\322x\352\1"..., 8192},
{"\6\242\0\0\261B\17\1\25z\352\1\0\0\1\6\315]\0\0\1\0\0\0\352F\1\0\324x\352\1"..., 8192},
{"\6\242\0\0\262B\17\1\25z\352\1\0\0\1\6\360\"\0\0\1\0\0\0\352F\1\0\325x\352\1"..., 8192},
{"\6\242\0\0\263B\17\1\25z\352\1\0\0\1\6\253\336\0\0\1\0\0\0\352F\1\0\327x\352\1"..., 8192},
{"\6\242\0\0\264B\17\1\25z\352\1\0\0\1\6\243\231\0\0\1\0\0\0\352F\1\0\330x\352\1"..., 8192},
{"\6\242\0\0\265B\17\1\25z\352\1\0\0\1\6\325\t\0\0\1\0\0\0\352F\1\0\331x\352\1"..., 8192},
{"\6\242\0\0\266B\17\1\25z\352\1\0\0\1\6`3\0\0\1\0\0\0\352F\1\0\333x\352\1"..., 8192},
{"\6\242\0\0\267B\17\1\25z\352\1\0\0\1\6\332'\0\0\1\0\0\0\352F\1\0\334x\352\1"..., 8192}], 8)
= 65536
Gettimeofday({1273083131, 538458}, NULL) = 0
```

O11.2 linux

538458-538145 = 313µs

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

Wait time, unknown duration

```
[oracle@fourcore ~]$ ps -eaf|grep oracleprod18
oracle   19759 19750  1 11:10 ?        00:00:01 oracleprod18 (DESCRIPTION=(LOCAL=YES)
(ADDRESS=(PROTOCOL=beq)))
. . .
[oracle@fourcore ~]$ strace -rp 19760
. . .
     0.000079 gettimeofday({1273083380, 795957}, NULL) = 0
     0.000081 _llseek(11, 8202387456, [8202387456], SEEK_SET) = 0
     0.000076 readv(11, [
{"\6\242\0\0004G\17\1\331\233\353\1\0\0\1\4\213\206\0\0\1\0\0\0\353F\. . ."..., 8192},
{"\6\242\0\0005G\17\1\360\233\353\1\0\0\3\4\364\35\0\0\1\0\0\0\353F\. . . "..., 8192},
{"\6\242\0\0006G\17\1\4\234\353\1\0\0\5\4%\"\0\0\1\0\0\0\353F\. . . "..., 8192},
{"\6\242\0\0007G\17\1{\235\353\1\0\0\2\4\3108\0\0\1\0\0\0\353F\. . . "..., 8192}], 4) = 32768
     0.000232 gettimeofday({1273083380, 796349}, NULL) = 0
. . .
```

o11.2 linux

```
796349 - 795957                = 392µs
.000081 + .000076 + .000232 = 389µs
```

Eliminating “Oracle”

```
[oracle@fourcore ~]$ strace -cp 19760
Process 19760 attached - interrupt to quit
Process 19760 detached
% time     seconds  usecs/call     calls    errors syscall
------ ----------- ----------- --------- --------- ----------------
 44.75    0.001108           0      9520      9091 semtimedop
 44.59    0.001104           0    101161           times
  5.01    0.000124           1        90           readv
  4.20    0.000104           0       224           pread64
  1.45    0.000036           0     21136           gettimeofday
  0.00    0.000000           0       323           getrusage
  0.00    0.000000           0        78           _llseek
  0.00    0.000000           0         1           sched_yield
  0.00    0.000000           0         1           semop
  0.00    0.000000           0        11           semctl
------ ----------- ----------- --------- --------- ----------------
100.00    0.002476                132545      9091 total
```

o11.2 linux

The sample was a couple of minutes.

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

Who is telling the truth?

DBA:
The average single block buffered read takes 25 ms.

Who is telling the truth?

DBA:
The average single block buffered read takes 25 ms.

IO Administrator:
I usually see IO times much faster than 25 ms.

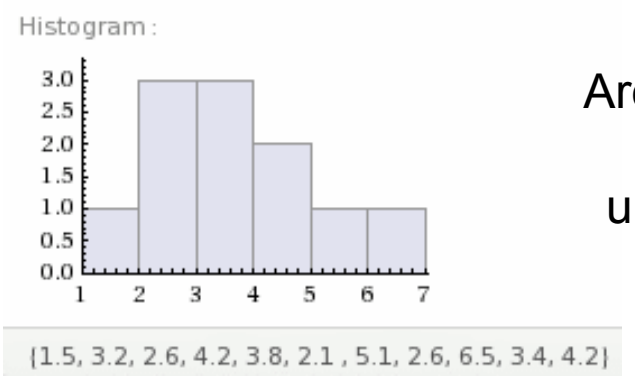
This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

Understanding the histogram

- The histogram is a fantastic way to understand a complex sample set or when there is lots of data.
- **The horizontal axis** are the sample values. The bars are uniformed sample value intervals, called bins.
- **The vertical axis** is the number of sample occurrences within the interval (bin).
- **Each sample** is represented, so if you add all the occurrences that will equal the number of sample values.

A simple histogram example.

```
{1.5, 3.2, 2.6, 4.2, 3.8, 2.1 , 5.1, 2.6, 6.5, 3.4, 4.2}
```



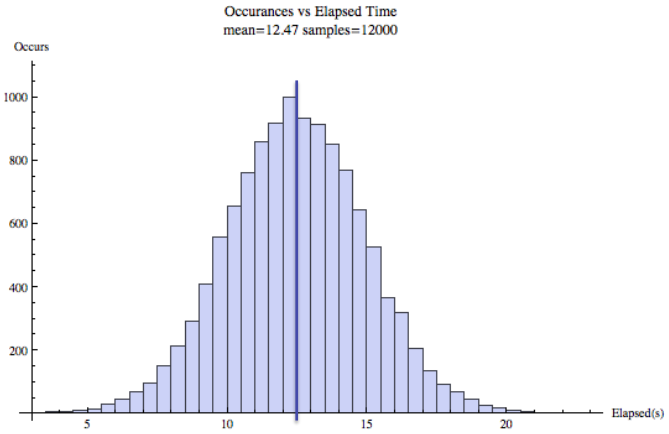
Are the bin sizes uniform?

```
mean=3.6    median=3.4
```

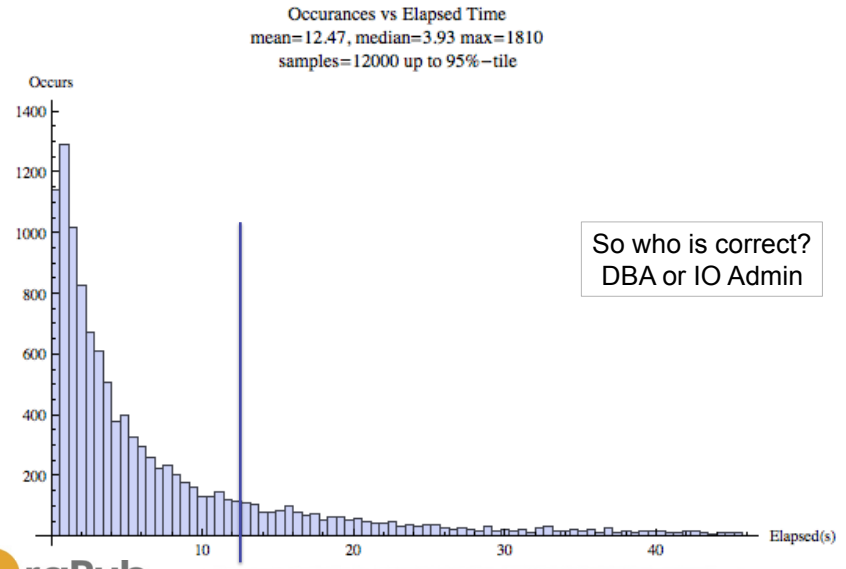

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

The perception of average WT.

$$WT_{avg} = 12.47 \text{ ms/event}$$



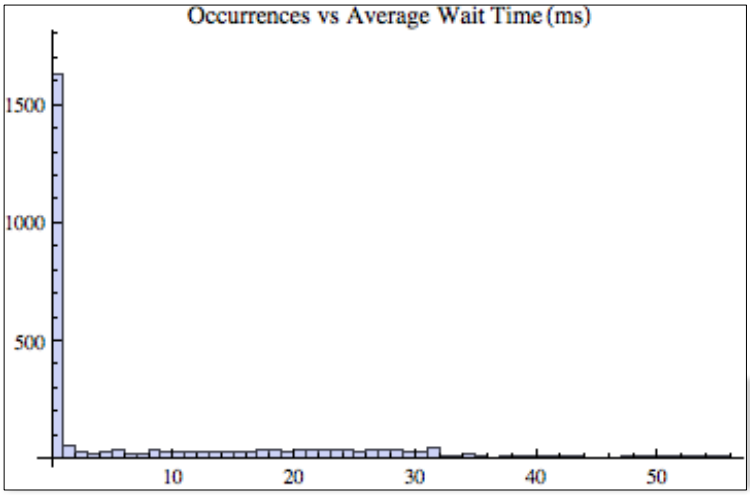
The reality of average WT.



So who is correct?
DBA or IO Admin

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

Reality−log file parallel write



Samples=3182 Mean=19.9503 Median=0.974367
Max=510.858 Quantile 90%=55.7297

Using v$event_histogram

Wait Event Histogram

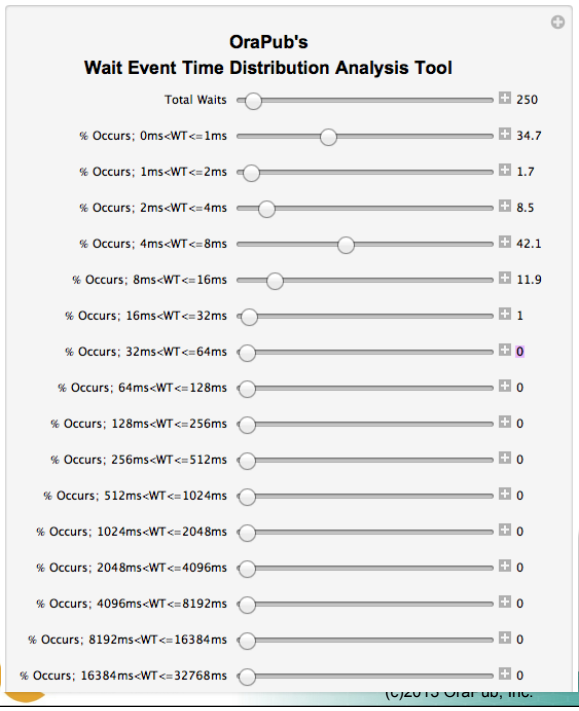
- Units for Total Waits column: K is 1000, M is 1000000, G is 1000000000
- % of Waits: value of .0 indicates value was <.05%; value of null is truly 0
- % of Waits: column heading of <=1s is truly <1024ms, >1s is truly >=1024ms
- Ordered by Event (idle events last)

Are the bin sizes uniform?

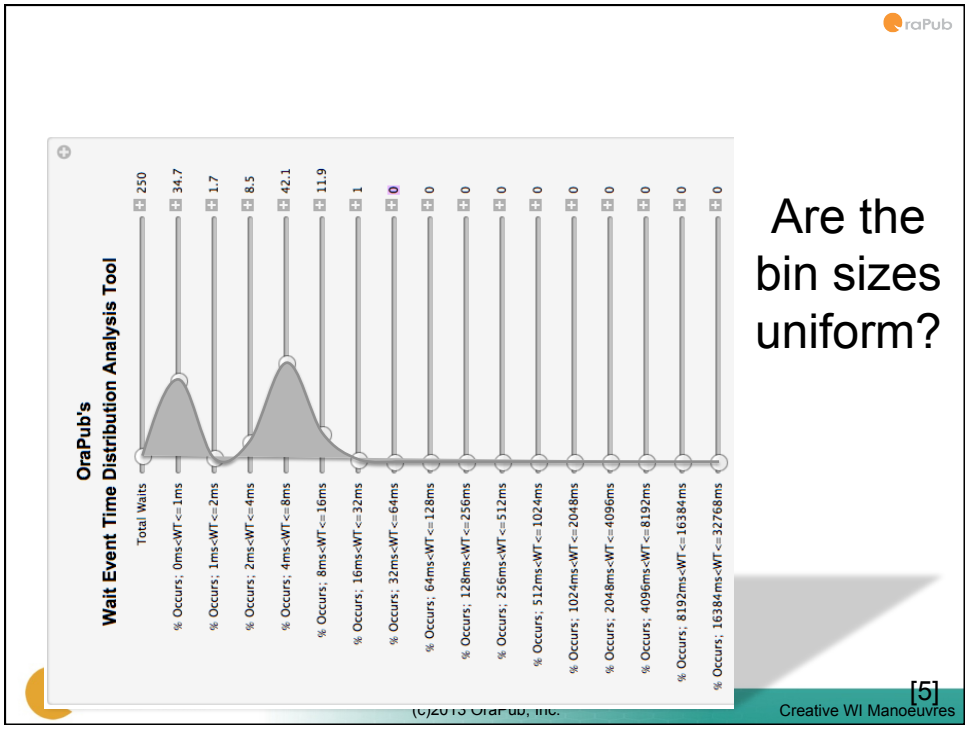
		% of Waits							
Event	Total Waits	<1ms	<2ms	<4ms	<8ms	<16ms	<32ms	<=1s	>1s
ADR block file read	192	84.9	.5	2.1	7.3	4.7	.5		
cursor: mutex S	18	100.0							
cursor: pin S	40	97.5	2.5						
cursor: pin S wait on X	1129					4.0	17.4	78.7	
db file parallel read	17						35.3	64.7	
db file parallel write	56.3K	21.4	35.3	11.2	11.0	18.8	2.2	.1	
db file scattered read	401	20.0	21.2	11.7	31.2	14.5	1.5		
db file sequential read	98.8M	34.7	1.7	8.5	42.1	11.9	1.0	.1	

source: awrrpt_RTX_20110426_0500_0800.html

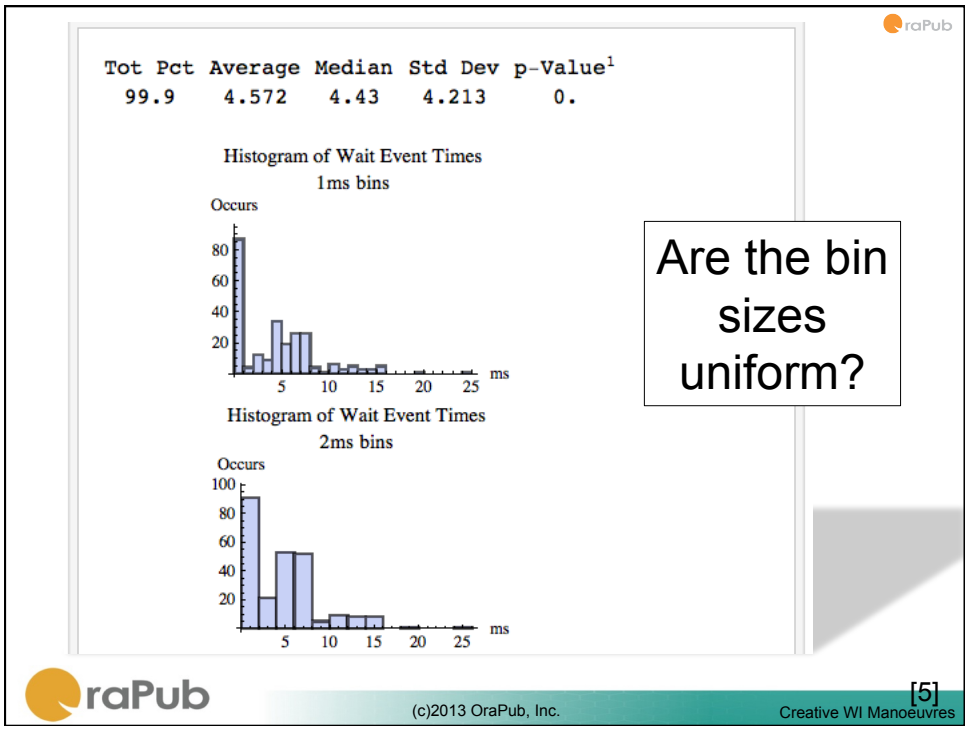
This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.



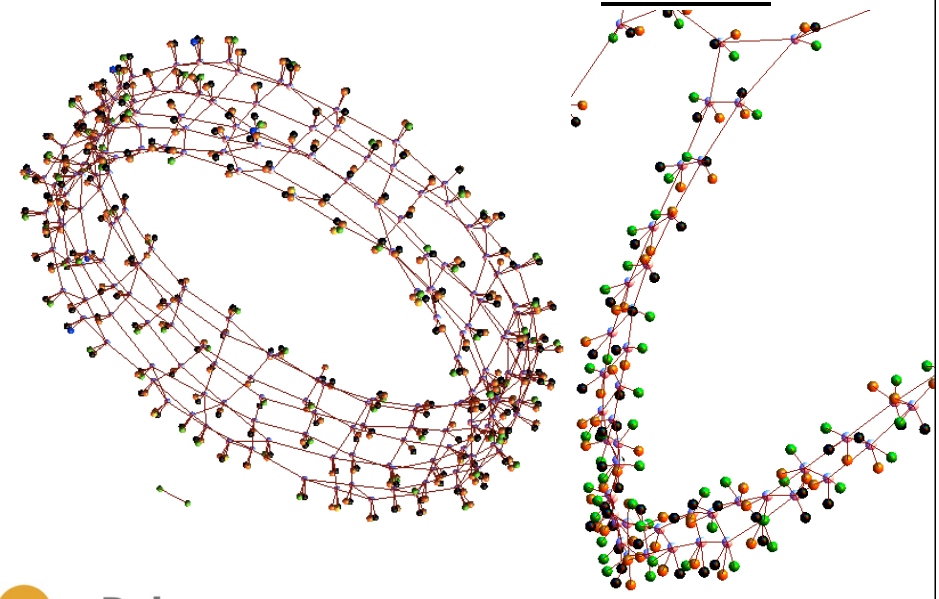
Picture the histogram in your mind.



This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

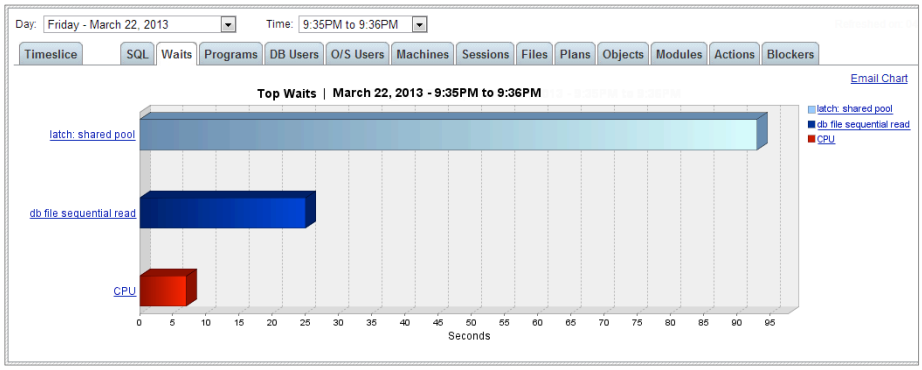


When wait time is known.



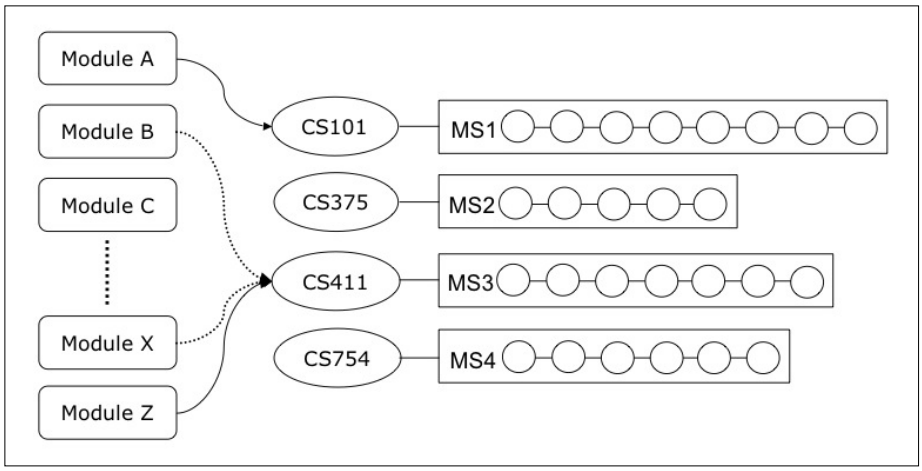
This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

The time we see from Oracle.



Sure, we can see the wait times but how does Oracle set the wait time and how do we know this "set" time?

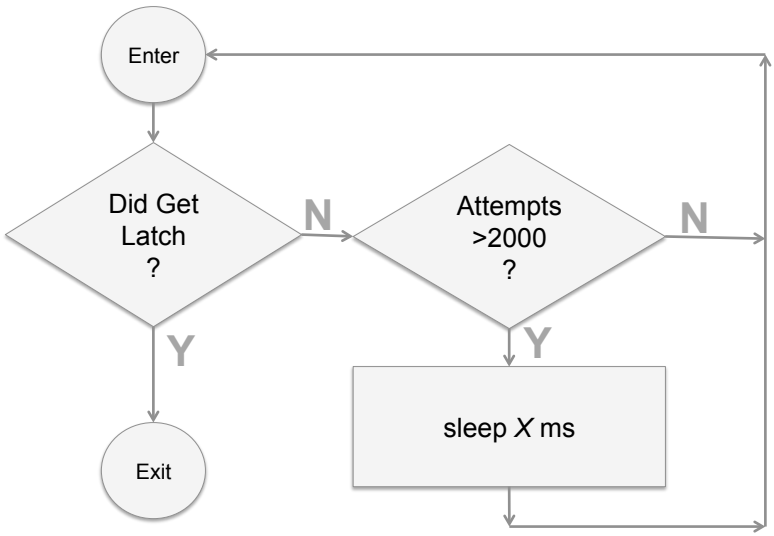
General serialization control.



Latches create the illusion of simultaneous access to Oracle memory structures.

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

Acquiring an Oracle latch.



Latch “sleep” time.

```
[oracle@fourcore]$ ps -eaf|grep oracleprod18
. . .
oracle   24477 24476 72 13:58 ?        01:22:03 oracleprod18 . . .
. . .
[oracle@fourcore]$ strace -rp 24477
Process 24477 attached - interrupt to quit
. . .
     0.000034 select(0, [], [], [], {0, 10001}) = 0 (Timeout)
     0.011137 gettimeofday({1264117983, 18827}, NULL) = 0
. . .
     0.000034 select(0, [], [], [], {0, 10001}) = 0 (Timeout)
     0.010299 gettimeofday({1264117983, 112927}, NULL) = 0
. . .
     0.000030 select(0, [], [], [], {0, 10001}) = 0 (Timeout)
     0.011160 gettimeofday({1264117983, 245021}, NULL) = 0
. . .
     0.000048 select(0, [], [], [], {0, 10001}) = 0 (Timeout)
     0.010456 gettimeofday({1264117983, 271123}, NULL) = 0
. . .
     0.000084 select(0, [], [], [], {0, 10001}) = 0 (Timeout)
     0.011098 gettimeofday({1264117983, 589336}, NULL) = 0
. . .
```


This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

Are *latch* wait times uniform?

```
Wait Event Histogram  DB/Inst: PDXPROD/PDXPROD  Snaps: 2625-2635
-> Total Waits - units: K is 1000, M is 1000000, G is 1000000000
-> % of Waits - column heading: <=1s is truly <1024ms, >1s is truly >=1024ms
-> % of Waits - value: .0 indicates value was <.05%, null is truly 0
-> Ordered by Event (idle events last)

                        Total ----------------- % of Waits ------------------
Event                   Waits  <1ms  <2ms  <4ms  <8ms <16ms <32ms  <=1s   >1s
------------------------ ----- ----- ----- ----- ----- ----- ----- ----- -----
. . .
db file sequential read     10K  90.3    .1    .2   3.5   3.6    .8   1.4
direct path read            12K  90.1   1.0    .9   2.0   2.1   1.1   2.8
direct path read temp      102   98.0                1.0   1.0
latch: cache buffers chain  53K    .6    .0    .1    .1    .2    .4  90.1   8.4
latch: library cache         5   60.0  20.0                    20.0
. . .
```

Agenda

- What is “wait time”?
- Can we really trust it?
- How Oracle gathers wait time
  - time is unknown
  - time is known
- Wait time and the OS administrator
- Creative ways to use wait time.
- Using a time based analysis to bring forth solutions.

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.

Want to dig deeper?

- **Presentations**: OraPub search, “time”
- **Craig’s Blog** – A Wider View
  - Search, “serialization”, “total time”, “visualization”
- **Training** from OraPub
  - 1 Day Performance Research Seminar
  - Oracle Performance Firefighting (I)
  - Adv Oracle Performance Analysis (II)

- **Tools** at www.orapub.com
  - Visual Total Time and Response Time Tool. OP and blog search, “total time”
  - Wait Event Time Distribution Analysis Tool. OP search, “wait histogram”
  - BC/LC Visualization. OP and blog search, “visualization”
  - OSM Toolkit. OP search, “osm”
- **Stori.** Interactive. Automated. Find and solve Oracle performance problems.
- **Books**
  - Oracle Performance Firefighting. “NOC_FF” $10 discount
  - Forecasting Oracle Performance. “NOC_FOP” $5 discount

This presentation was given by Craig Shallahamer at the NoCOUG conference on 15-AUG-2013.