ORACLE

Oracle RAC 12c Practical Performance Management and Tuning

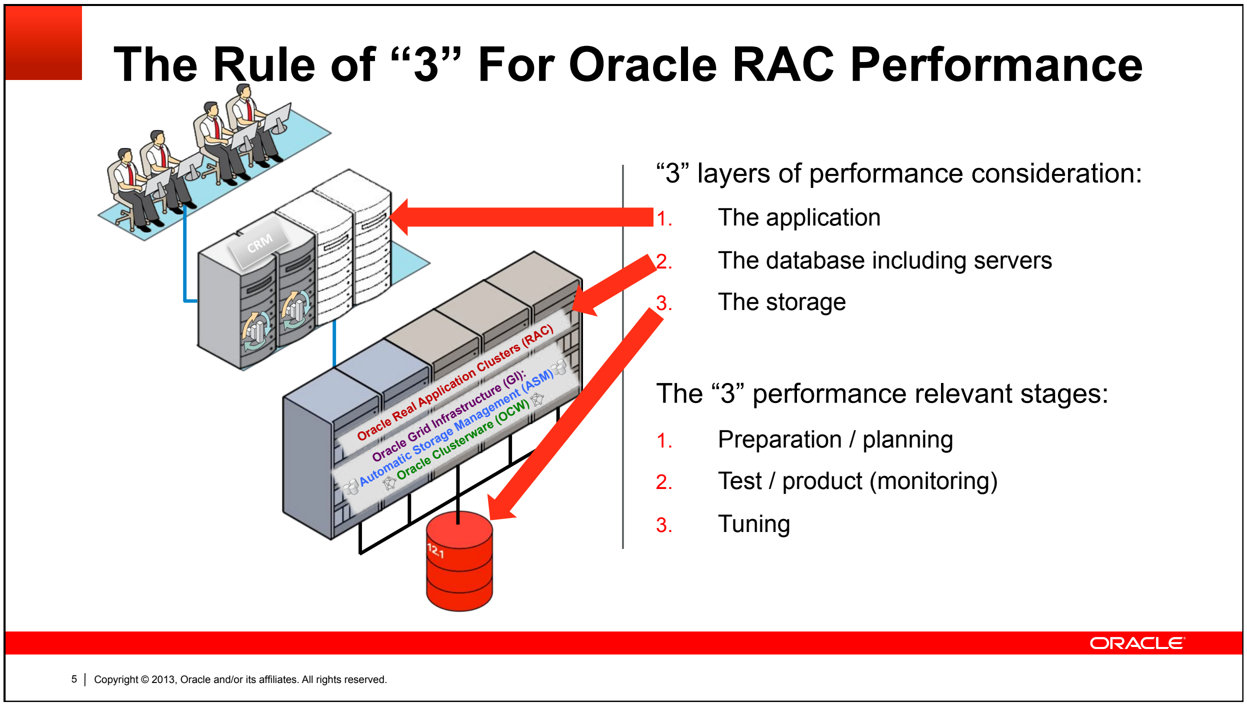
Markus Michalewicz,
Director of Product Management
Michael Zoll,
Oracle RAC Performance Architect

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Agenda

- **Introduction**
- Fundamentals
- Network Recommendations
- Application Considerations
- Testing
- Identifying Issues
- Monitoring & Analysis
- Summary and Q&A



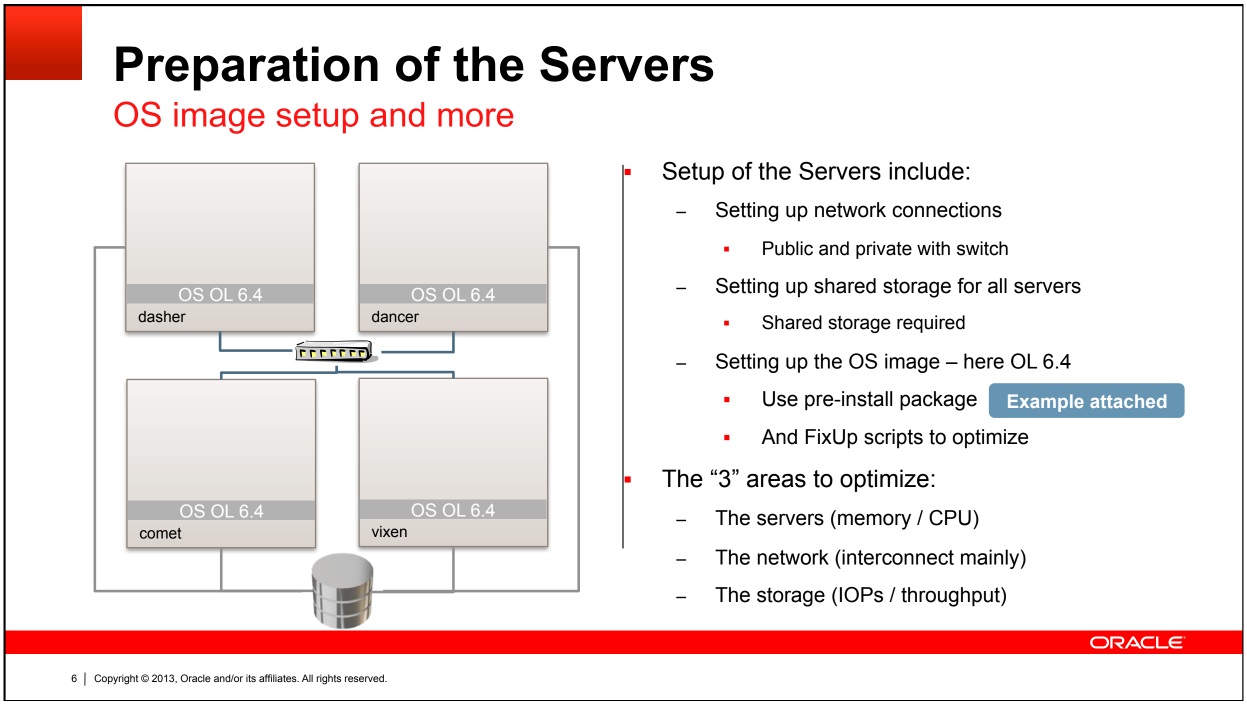
The Rule of “3” For Oracle RAC Performance

“3” layers of performance consideration:

1. The application
2. The database including servers
3. The storage

The “3” performance relevant stages:

1. Preparation / planning
2. Test / product (monitoring)
3. Tuning



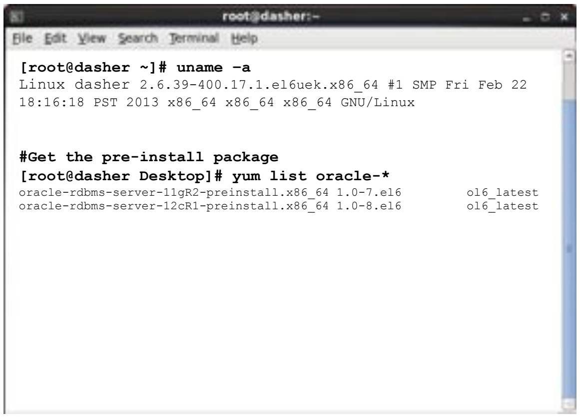
Preparation of the Servers

OS image setup and more

- Setup of the Servers include:
  - Setting up network connections
    - Public and private with switch
  - Setting up shared storage for all servers
    - Shared storage required
  - Setting up the OS image – here OL 6.4
    - Use pre-install package Example attached
    - And FixUp scripts to optimize
- The “3” areas to optimize:
  - The servers (memory / CPU)
  - The network (interconnect mainly)
  - The storage (IOPs / throughput)

Preparation of the Servers

Kernel and Oracle preinstall package



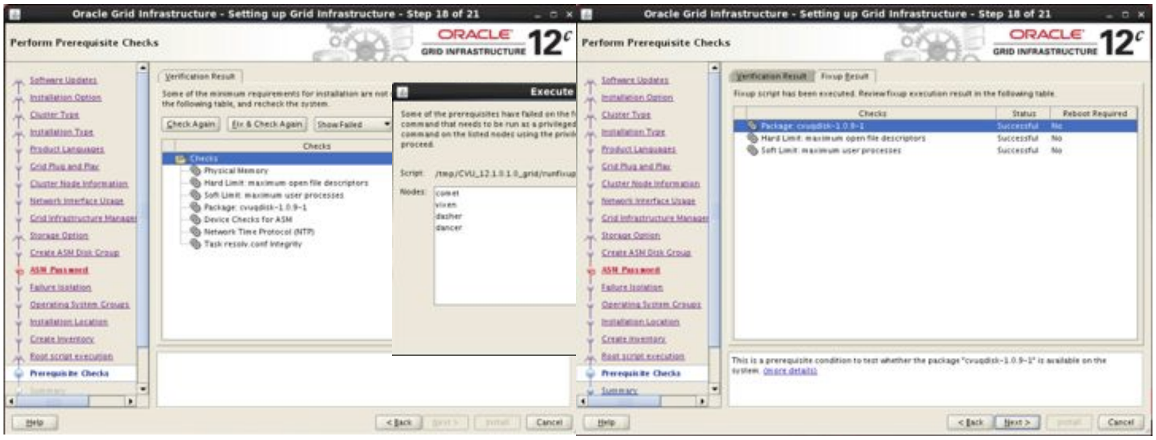
```
[root@dasher ~]# uname -a
Linux dasher 2.6.39-400.17.1.el6uek.x86_64 #1 SMP Fri Feb 22
18:16:18 PST 2013 x86_64 x86_64 x86_64 GNU/Linux


#Get the pre-install package
[root@dasher Desktop]# yum list oracle-*
oracle-rdbms-server-11gR2-preinstall.x86_64 1.0-7.el6        ol6_latest
oracle-rdbms-server-12cR1-preinstall.x86_64 1.0-8.el6        ol6_latest
```

- **Recommendations**:
  - Use OL 6.4+ UEK
  - Use the Oracle preinstall package
    - Available for
      - Oracle Database 11g Rel. 2
      - Oracle Database 12c Rel. 1
  - **Additional steps** to perform after using the pre-install package for an Oracle RAC cluster:
    - Add a “grid” user & “dba” group
    - Align UIDs / GUIDs across servers
    - Set SELinux to “permissive”, if possible
    - Check large / huge pages settings

Install Oracle Grid Infrastructure 12*c*

Use FixUp scripts to omptimize the configuraiton – part 1



Some of the minimum requirements for installation are not
the following table, and recheck the system.

Checks
- Physical Memory
- Hard Limit: maximum open file descriptors
- Soft Limit: maximum user processes
- Package: cvuqdisk-1.0.9-1
- Device Checks for ASM
- Network Time Protocol (NTP)
- Task resolv.conf integrity

Some of the prerequisites have failed on the f
command that needs to be run as a privileged
command on the listed nodes using the privi
proceed.

Script: /tmp/CVU_12.1.0.1.0_grid/runfixup

Nodes:
comet
vixen
dasher
dancer

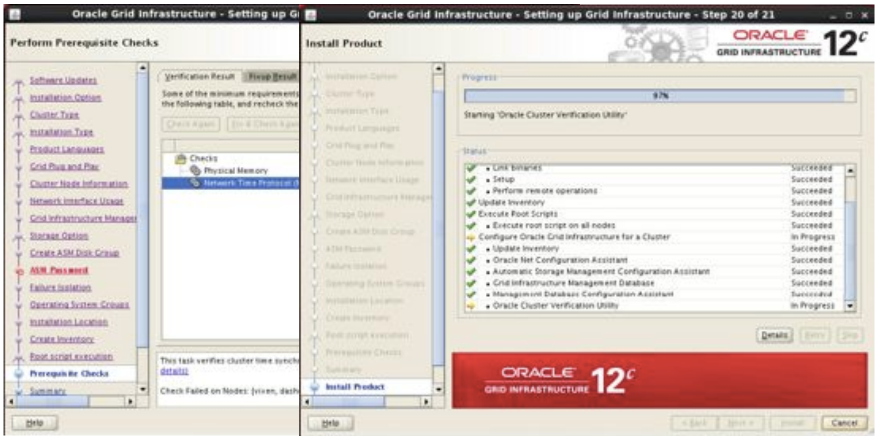
Fixup script has been executed. Reviewfixup execution result in the following table.

Checks	Status	Reboot Required
Package: cvuqdisk-1.0.9-1	Successful	No
Hard Limit: maximum open file descriptors	Successful	No
Soft Limit: maximum user processes	Successful	No

This is a prerequisite condition to test whether the package "cvuqdisk-1.0.9-1" is available on the system. (more details)

Install Oracle Grid Infrastructure 12*c*

Use FixUp scripts to omptimize the configuraiton – part 2



Some of the minimum requirements
the following table, and recheck the

This task verifies cluster time synch
details)

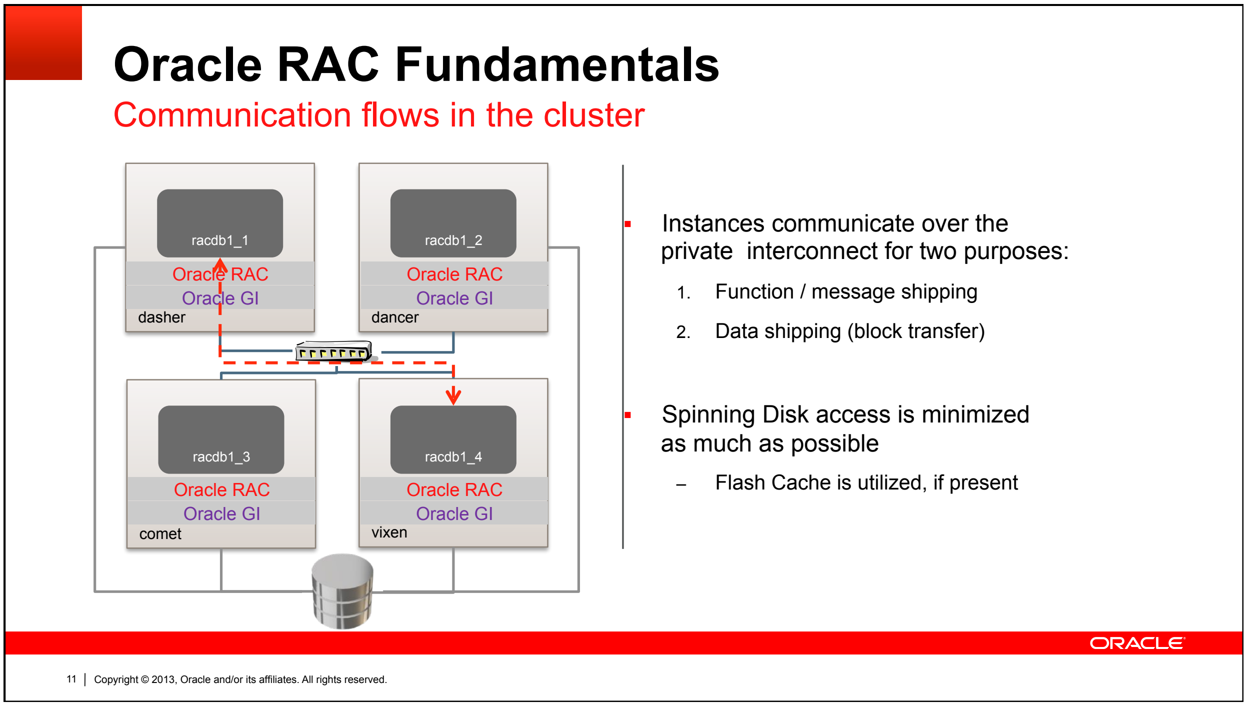
Check Failed on Nodes: [vixen, dash

Starting 'Oracle Cluster Verification Utility'

```
✓ • Link binaries                                          Succeeded
✓ • Setup                                                  Succeeded
✓ • Perform remote operations                              Succeeded
✓ Update Inventory                                         Succeeded
✓ Execute Root Scripts                                     Succeeded
✓ • Execute root script on all nodes                       Succeeded
➜ Configure Oracle Grid Infrastructure for a Cluster      In Progress
✓ • Update Inventory                                       Succeeded
✓ • Oracle Net Configuration Assistant                     Succeeded
✓ • Automatic Storage Management Configuration Assistant   Succeeded
✓ • Grid Infrastructure Management Database                Succeeded
✓ • Management Database Configuration Assistant            Succeeded
➜ • Oracle Cluster Verification Utility                    In Progress
```

Agenda

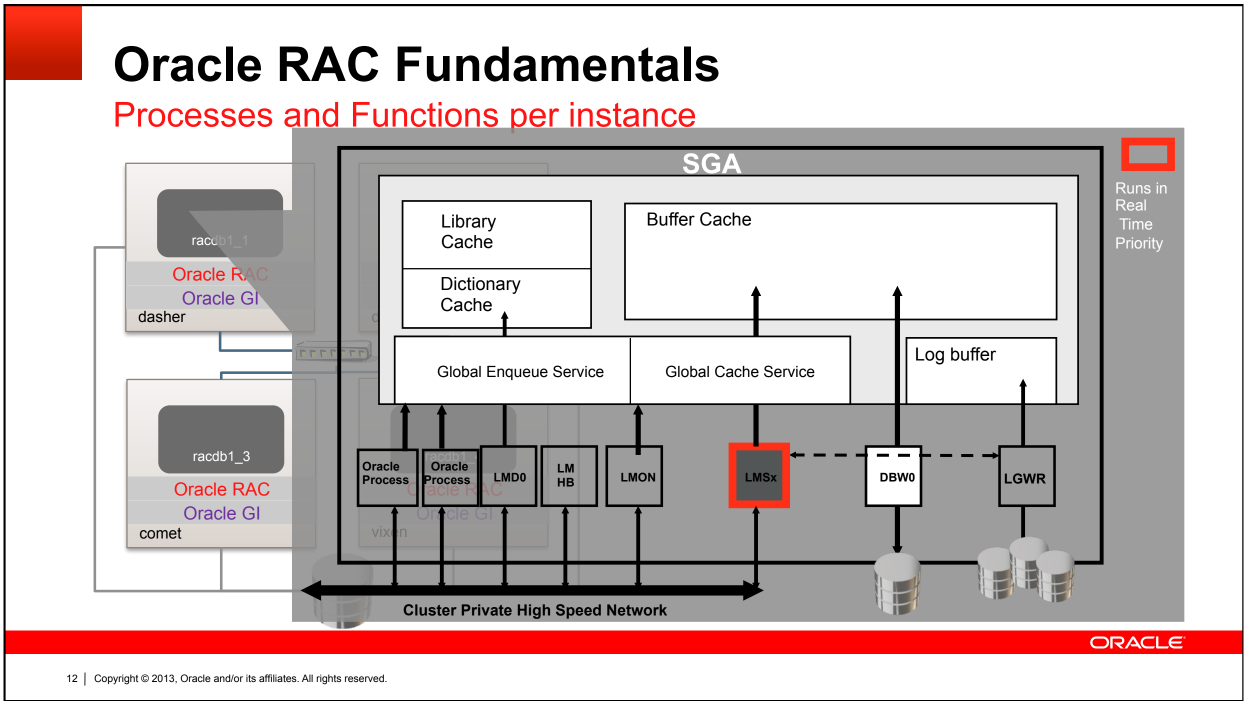
- Introduction
- **Fundamentals**
- Network Recommendations
- Application Considerations
- Testing
- Identifying Issues
- Monitoring & Analysis
- Summary and Q&A



Oracle RAC Fundamentals

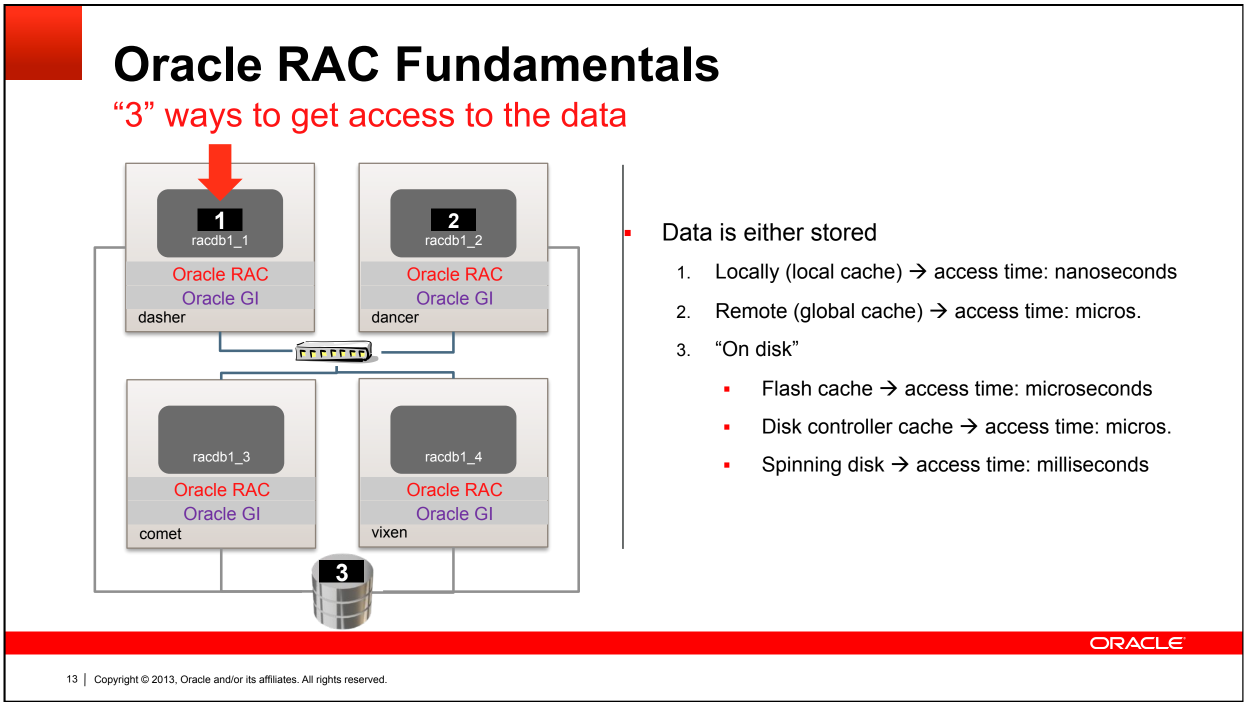
Communication flows in the cluster

- Instances communicate over the private interconnect for two purposes:
  1. Function / message shipping
  2. Data shipping (block transfer)
- Spinning Disk access is minimized as much as possible
  - Flash Cache is utilized, if present



Oracle RAC Fundamentals

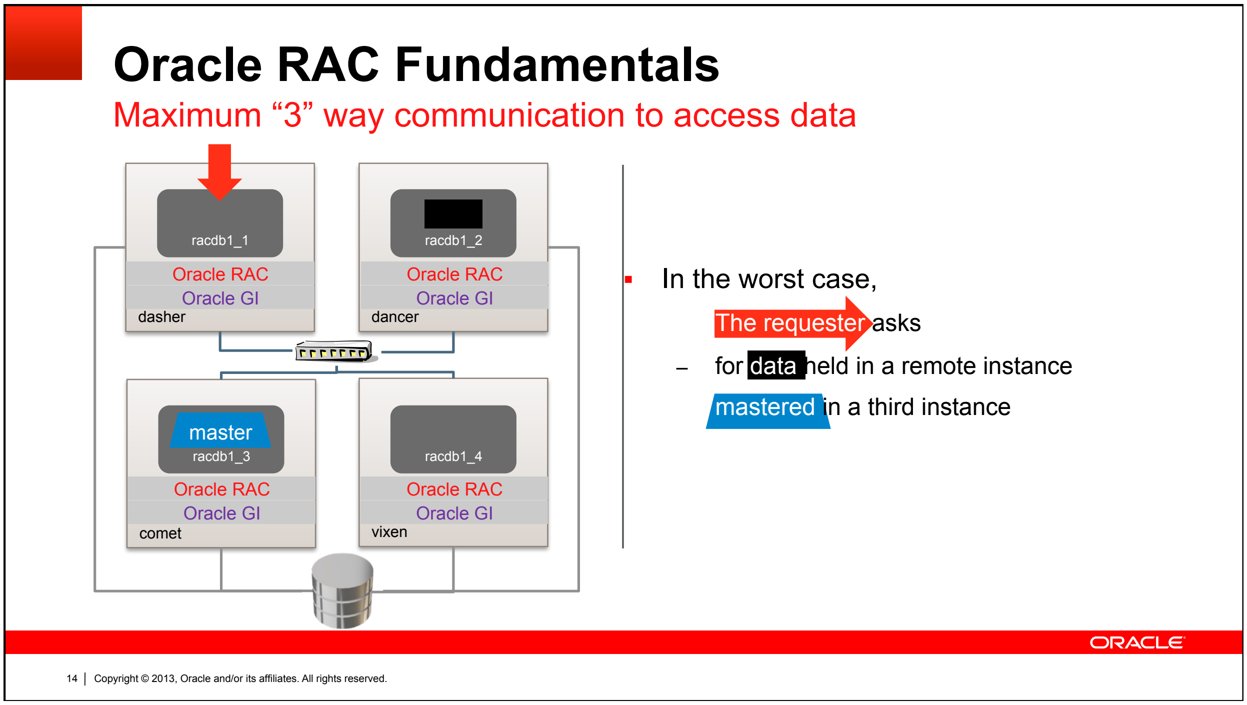
Processes and Functions per instance



Oracle RAC Fundamentals

“3” ways to get access to the data

- Data is either stored
  1. Locally (local cache) → access time: nanoseconds
  2. Remote (global cache) → access time: micros.
  3. “On disk”
     - Flash cache → access time: microseconds
     - Disk controller cache → access time: micros.
     - Spinning disk → access time: milliseconds



Oracle RAC Fundamentals

Maximum “3” way communication to access data

- In the worst case,

  The requester asks
  - for data held in a remote instance

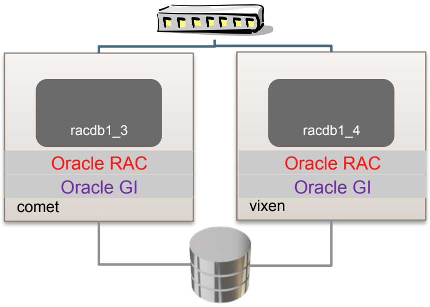
  mastered in a third instance

Agenda

- Introduction
- Fundamentals
- **Network Recommendations**
- Application Considerations
- Testing
- Identifying Issues
- Monitoring & Analysis
- Summary and Q&A

Network Recommendations

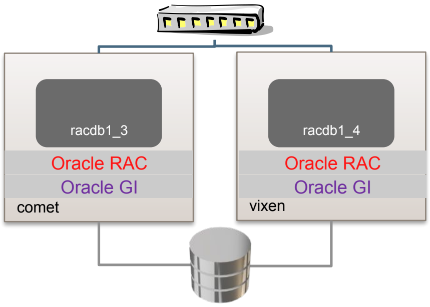
What is "normal"?



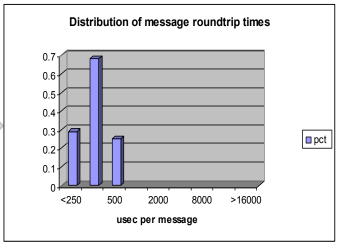
Block size RT (usec)	2K	4K	8K	16K
UDP/GE	300	310	360	460
RDS/IB	100	130	160	200

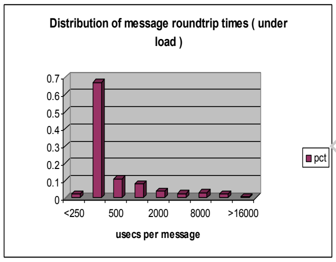
Network Recommendations

Interconnect Latency Variation



Mean: 250 µsecs





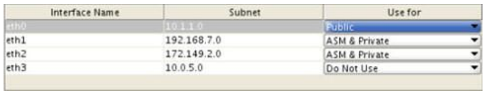
Mean: 800 µsecs
5% > 8 ms

- Interconnect Utilization or Network Configuration
- High CPU Utilization and Process Concurrency
- LMS utilization and priorities

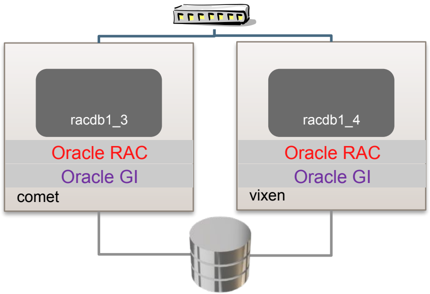
ORACLE

Network Recommendations

How to prepare to get to a “normal” state?

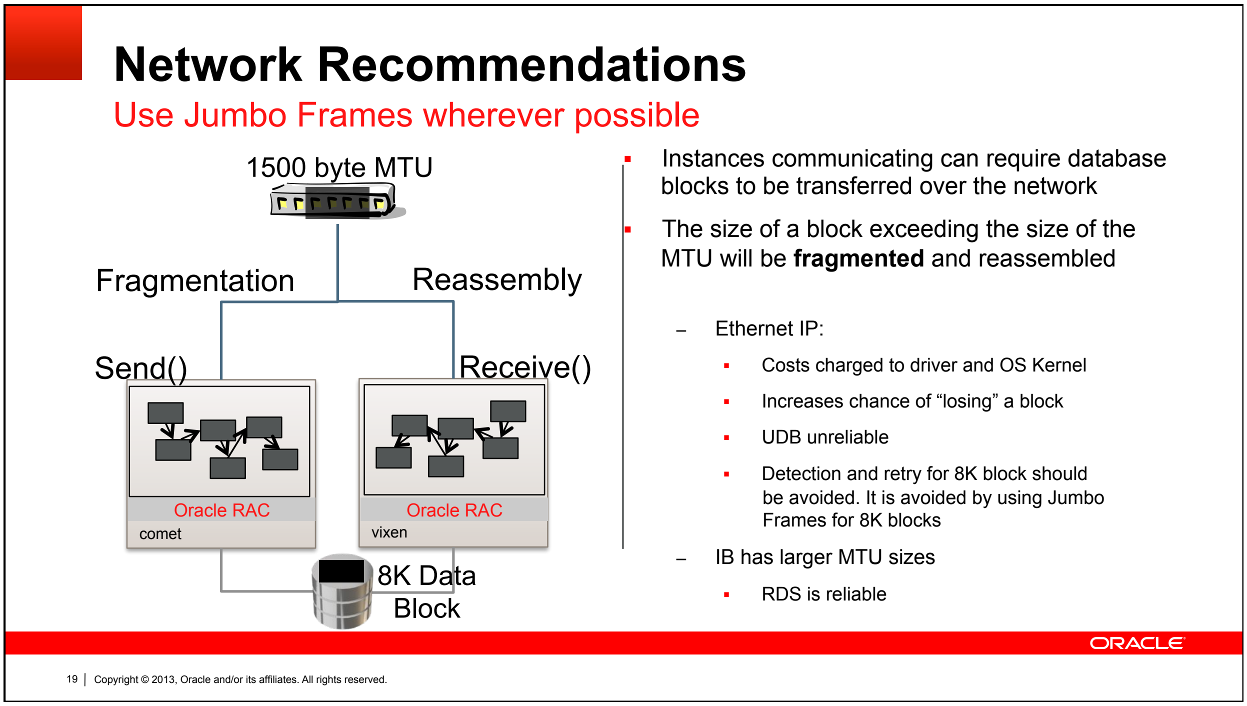


Interface Name	Subnet	Use for
eth0	10.1.1.0	Public
eth1	192.168.7.0	ASM & Private
eth2	172.149.2.0	ASM & Private
eth3	10.0.5.0	Do Not Use



- Size Interconnect for aggregated throughput
  - ROT approx1Gb/sec per 32 cores
- 10Gb/sec or IB generally good for database consolidations and large SMP nodes
- Bonded 1GbE solutions with load balancing are supported
  - Use same type NICs for LB and failover
  - Use different subnets
- Use Jumbo frames wherever possible
- Check for IP & NIC send and receive buffer size
- Interconnect should be stress tested with Iperf or Netperf

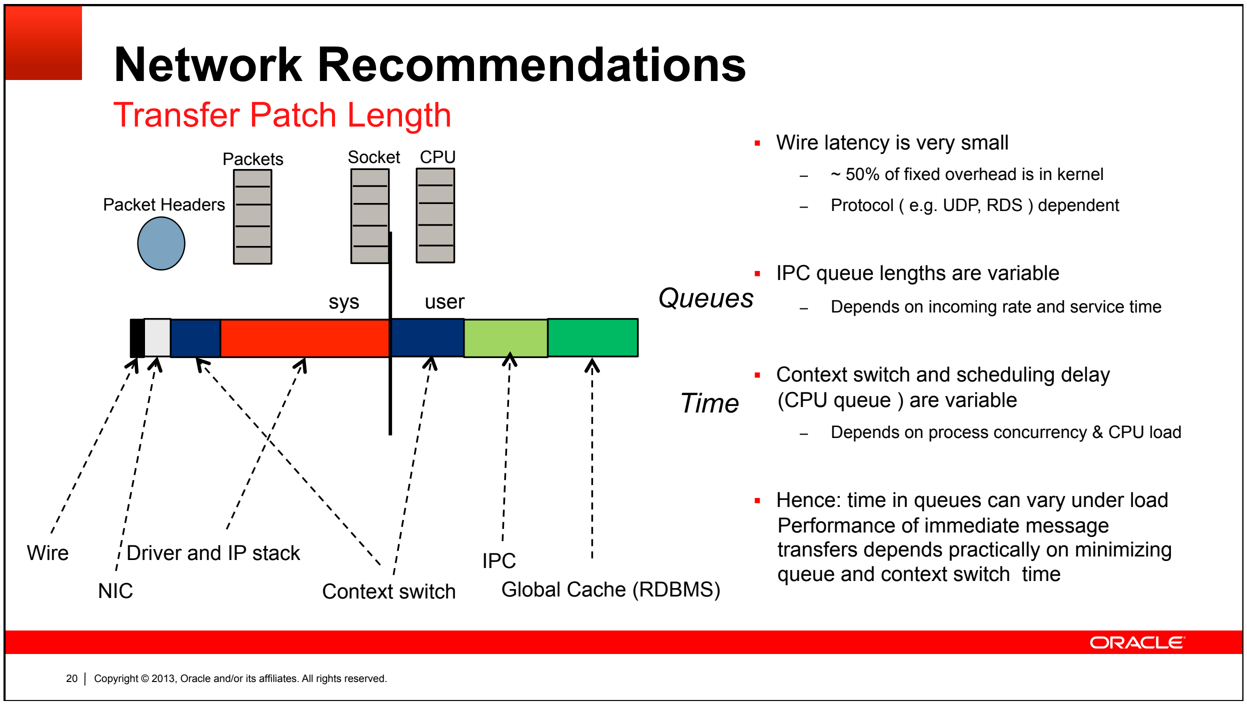
ORACLE



Network Recommendations

Use Jumbo Frames wherever possible

- Instances communicating can require database blocks to be transferred over the network
- The size of a block exceeding the size of the MTU will be **fragmented** and reassembled
  - Ethernet IP:
    - Costs charged to driver and OS Kernel
    - Increases chance of "losing" a block
    - UDB unreliable
    - Detection and retry for 8K block should be avoided. It is avoided by using Jumbo Frames for 8K blocks
  - IB has larger MTU sizes
    - RDS is reliable



Network Recommendations

Transfer Patch Length

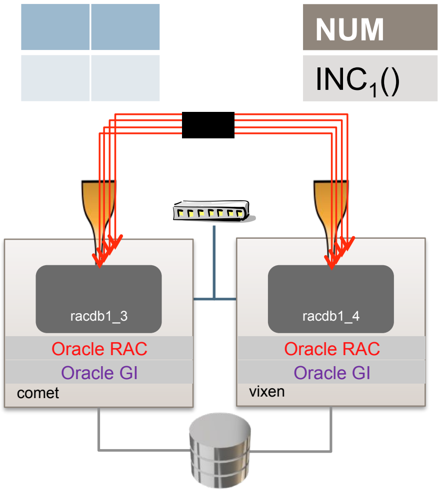
- Wire latency is very small
  - ~ 50% of fixed overhead is in kernel
  - Protocol (e.g. UDP, RDS) dependent
- IPC queue lengths are variable
  - Depends on incoming rate and service time
- Context switch and scheduling delay (CPU queue) are variable
  - Depends on process concurrency & CPU load
- Hence: time in queues can vary under load Performance of immediate message transfers depends practically on minimizing queue and context switch time

Agenda

- Introduction
- Fundamentals
- Network Recommendations
- **Application Considerations**
- Testing
- Identifying Issues
- Monitoring & Analysis
- Summary and Q&A

Application Considerations

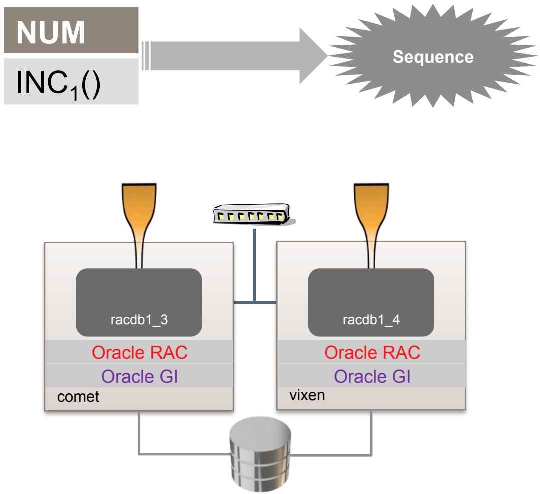
How to avoid “Write Hot Spots” in applications – part 1



- Frequent transactional changes to the same data blocks in all instances may result in “write hot spots”
  - *In 99% of OLTP performance issues, write hot spots occur on indexes*
- Block with pending changes may be “pinged” by other instance
- Pending redo must be written to log before the block can be transferred
- Only for very frequently modified data
- Latency for a deferred block transfer becomes dependent on delay for log IO
- LGWR and LMS interact asynchronously

Application Considerations

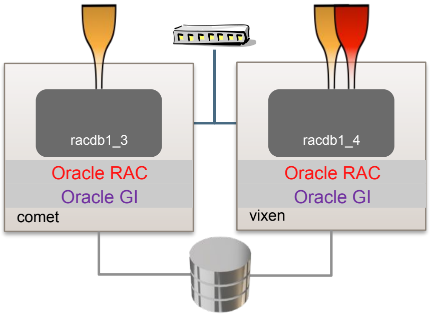
How to avoid “Write Hot Spots” in applications – part 2



- Use **non-ordered & cached** sequences if sequence is used to generate primary key
  - ALTER SEQUENCE S1 … CACHE 10000+
- Symptoms if not cached:
  - Enq; SQ – contention
  - Row cache lock
- Ordered Sequences
  - Do not scale well in Oracle RAC
  - Symptoms:
    - enq: SV – contention / DFS lock handle
  - Solution: Use them only on one instance in active-passive configuration
    - Create multiple per application and route

Application Considerations

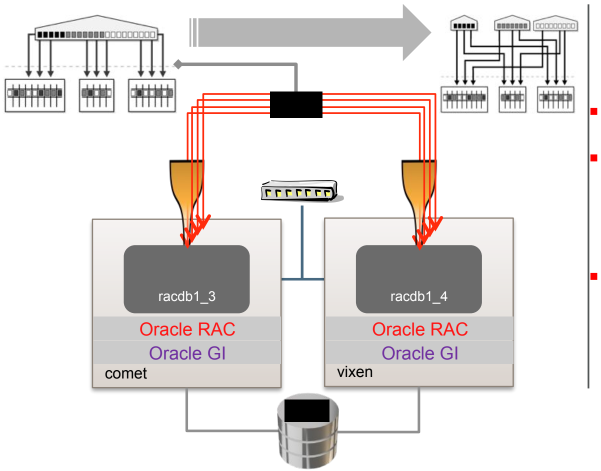
How to avoid “Write Hot Spots” in applications – part 3



- Possible:
  - Consolidate applications to use only one server and route via services
- Optimize log flush :
  - Place redo logs on fast storage if performance critical; e.g. SSDs
  - Separate disks for logs from other IO busy disks
    - Implemented in 11.2.2.4 of Exadata and Oracle Database Appliance by default (Smart Logs and SSDs, respectively)
- Schema tuning only involves minimal modification and is the preferred option

Application Considerations

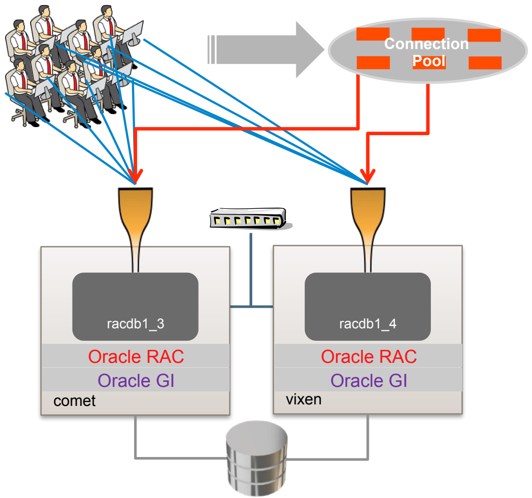
How to avoid "Write Hot Spots" in applications – part 4



- Global hash partitioned indexes
- Locally partitioned indexes
  - **Both solutions achieve better cache locality**
- Drop unused indexes

Application Considerations

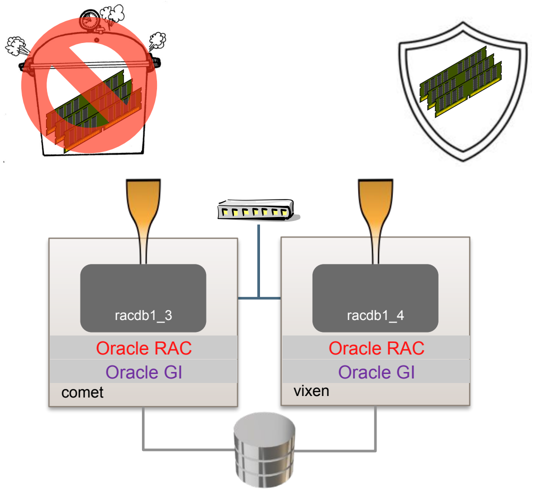
How to avoid Resource Contention in applications



- Foreground processes are in time-share class
- Scheduling delays on high context switch rates on busy systems may increase the variation in the cluster traffic times
- Latch and mutex contention can cause priority inversion issues for critical background procs.
- More processes imply higher memory utilization and higher risk of paging
- Control the number of concurrent processes
  - Use connection pooling
  - Avoid connection storms (pool and process limits)
- Ensure that load is well-balanced over nodes

Application Considerations

Optimize memory locally to prevent cluster-wide impact

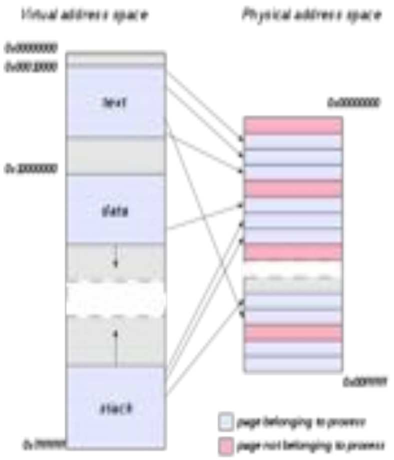


- Avoid memory pressure!
  - Paging and Swapping activity on one node affects performance on all nodes
  - **Severe Paging and Swapping activity on one node can cause instance evictions**
    - *#1 cause for service disruptions in clusters*
- **Must** use Huge pages for SGA (Linux)
  - Saves memory for page tables
  - Pins pages for SGA
- **Use Memory Guard**
  - QoS feature – available in monitoring only mode
  - Prevents new connections from coming in to a server that is already under memory pressure

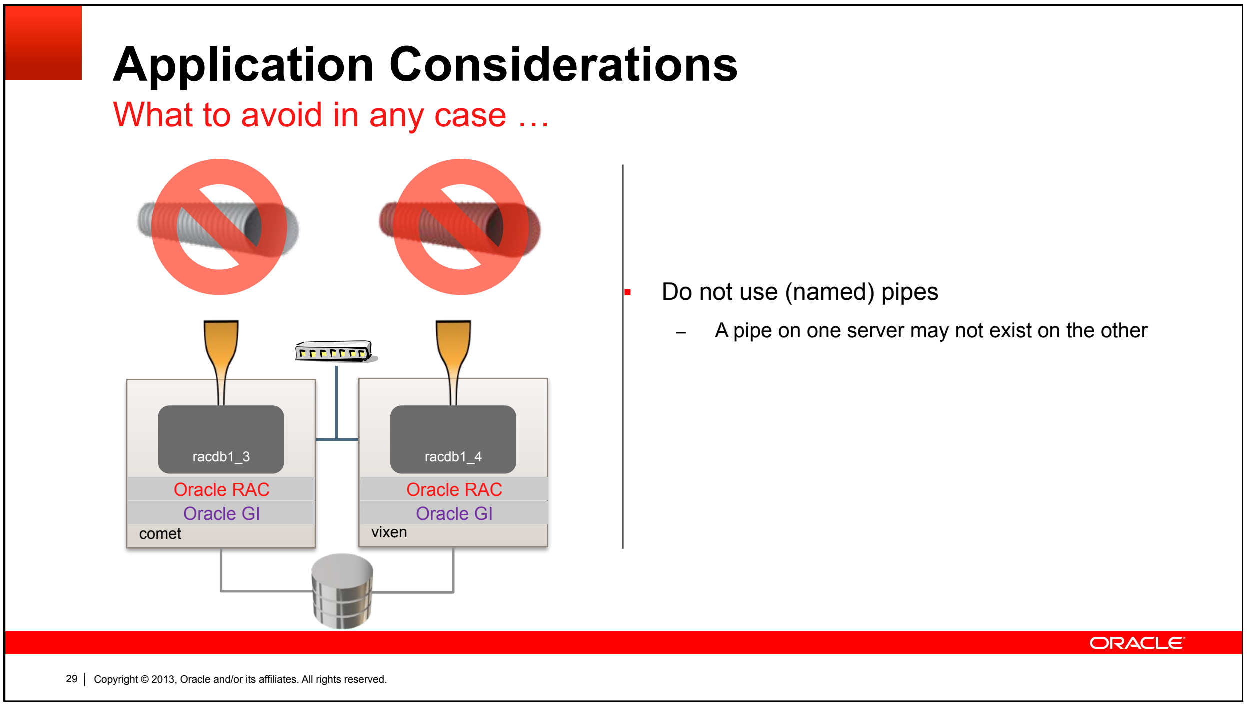
---

Application Considerations

Configure Huge Pages for Oracle RAC



- Use Huge pages for SGA (Linux)
  - Dramatic reduction in memory for page tables
  - SGA pages pinned in memory
- **More information:**
  - My Oracle Support note 361323.1 – HugePages on Linux: What It Is... and What It Is Not...
  - My Oracle Support note 401749.1 – Shell Script to Calculate Values Recommended Linux HugePages / HugeTLB Configuration
- Engineered systems provide templates for pre-configuration of huge pages for the SGA

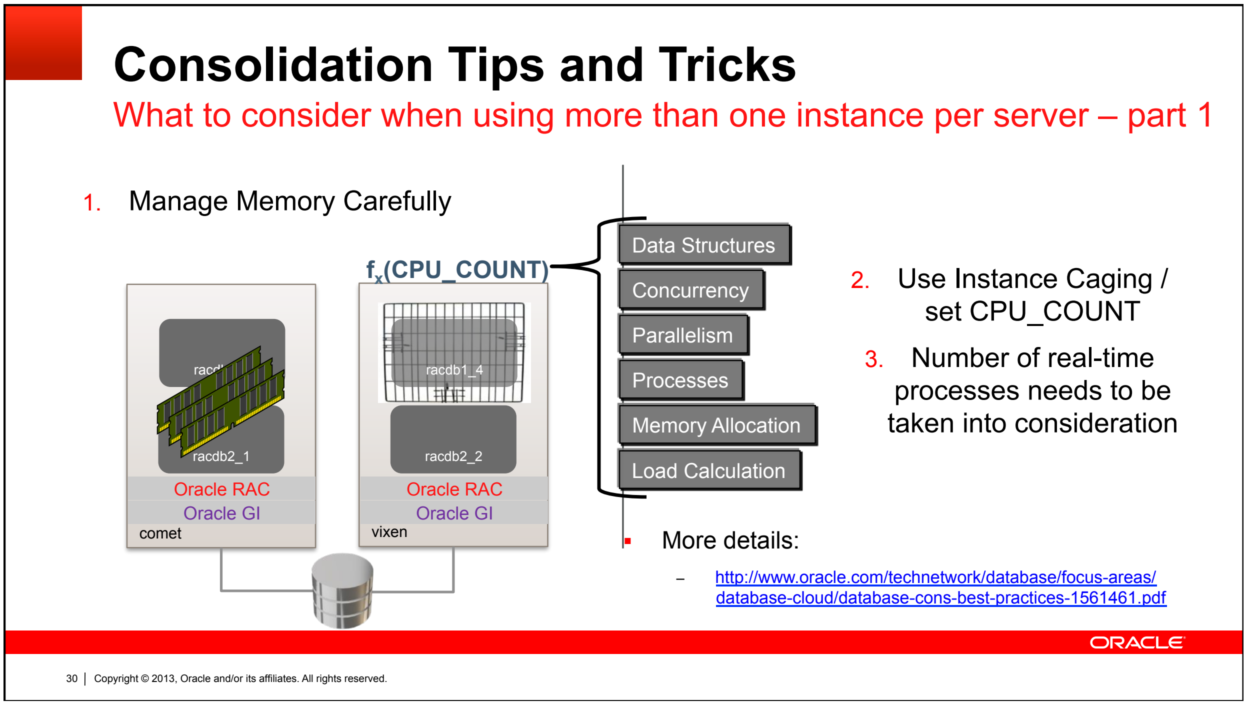


Application Considerations

What to avoid in any case …

- Do not use (named) pipes
  - A pipe on one server may not exist on the other

29 |



Consolidation Tips and Tricks

What to consider when using more than one instance per server – part 1

1. Manage Memory Carefully

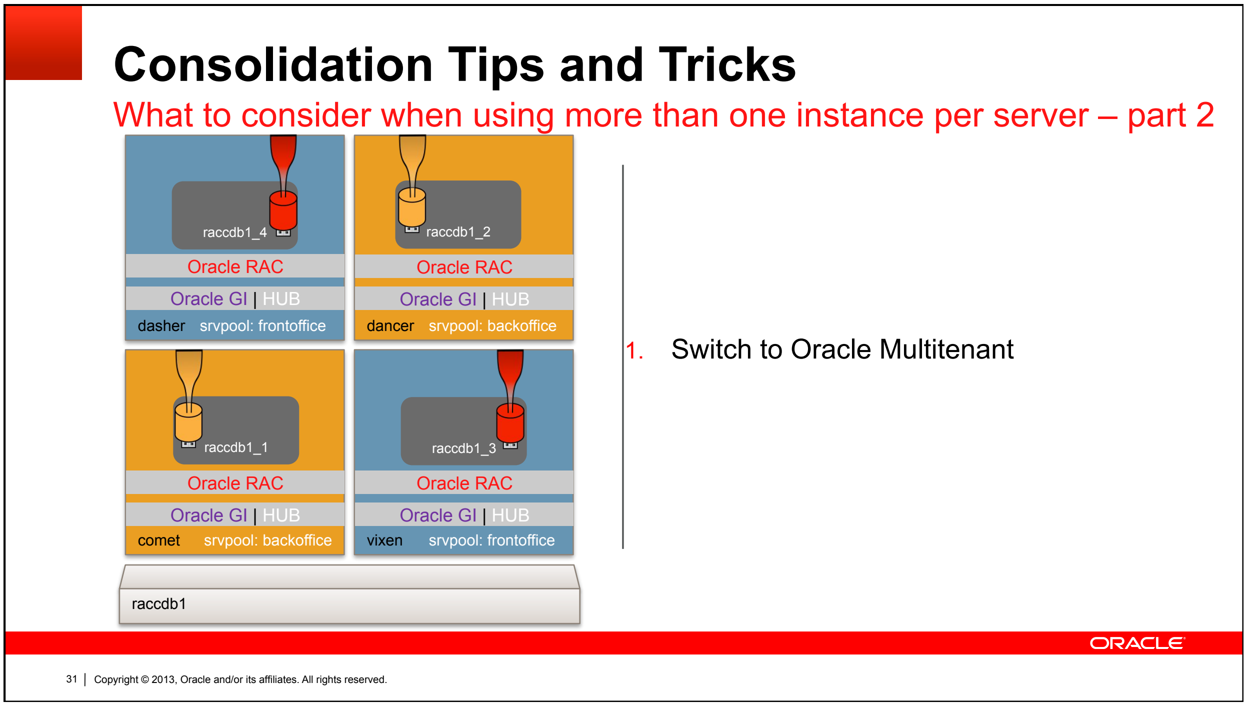
f_x(CPU_COUNT)

- Data Structures
- Concurrency
- Parallelism
- Processes
- Memory Allocation
- Load Calculation

2. Use Instance Caging / set CPU_COUNT
3. Number of real-time processes needs to be taken into consideration

- More details:
  - http://www.oracle.com/technetwork/database/focus-areas/database-cloud/database-cons-best-practices-1561461.pdf

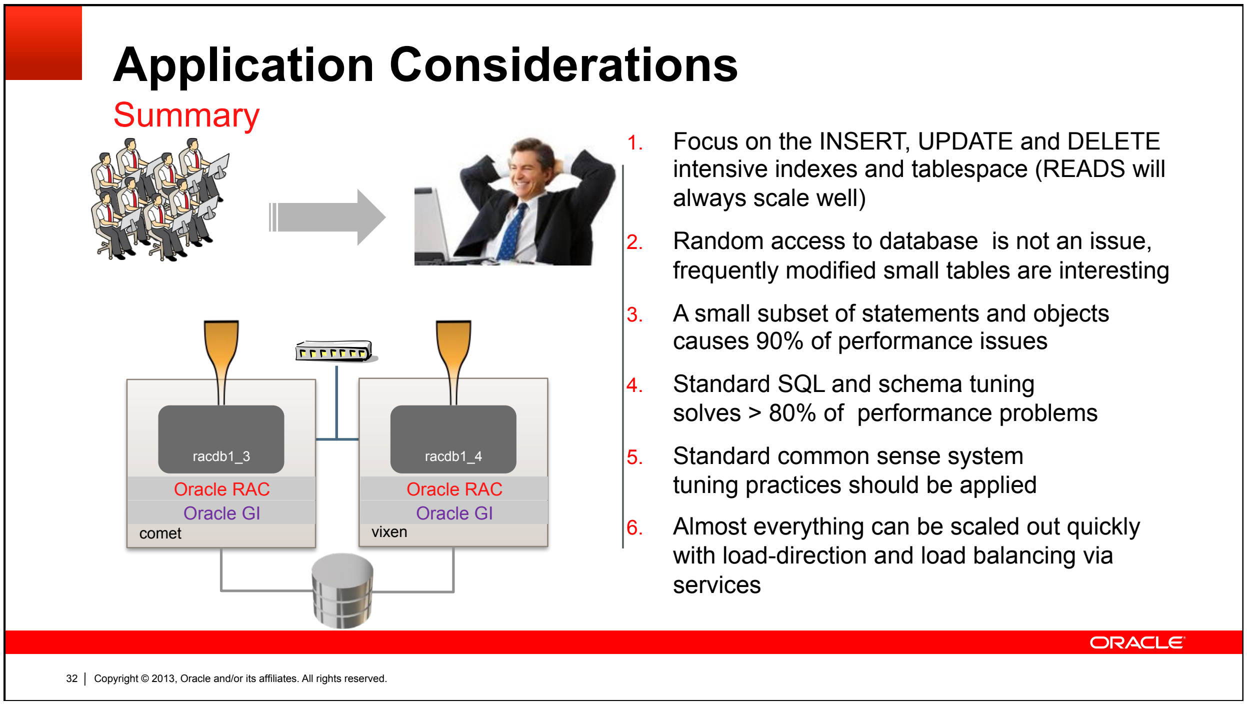
30 |



Consolidation Tips and Tricks

What to consider when using more than one instance per server – part 2

1. Switch to Oracle Multitenant



Application Considerations

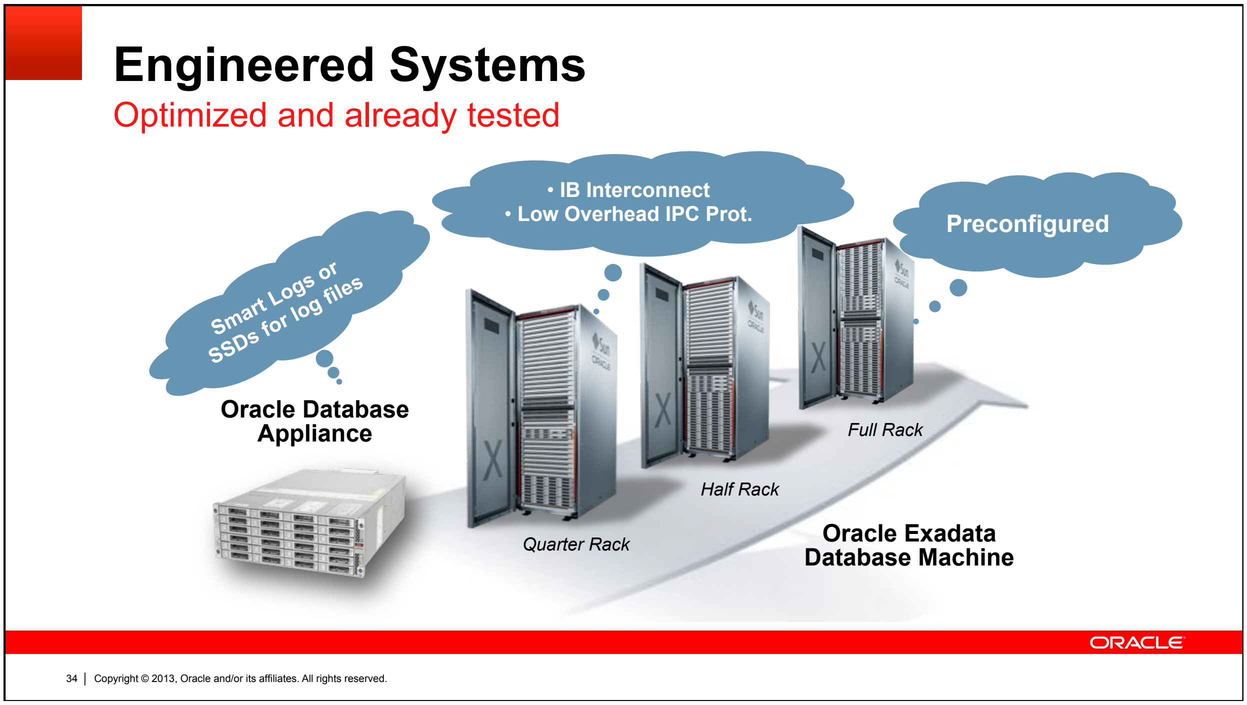
Summary

1. Focus on the INSERT, UPDATE and DELETE intensive indexes and tablespace (READS will always scale well)
2. Random access to database is not an issue, frequently modified small tables are interesting
3. A small subset of statements and objects causes 90% of performance issues
4. Standard SQL and schema tuning solves > 80% of performance problems
5. Standard common sense system tuning practices should be applied
6. Almost everything can be scaled out quickly with load-direction and load balancing via services

Agenda

- Introduction
- Fundamentals
- Network Recommendations
- Application Considerations
- **Testing**
- Identifying Issues
- Monitoring & Analysis
- Summary and Q&A

ORACLE



Engineered Systems

Optimized and already tested

Smart Logs or SSDs for log files

- IB Interconnect
- Low Overhead IPC Prot.

Preconfigured

Oracle Database Appliance

Quarter Rack

Half Rack

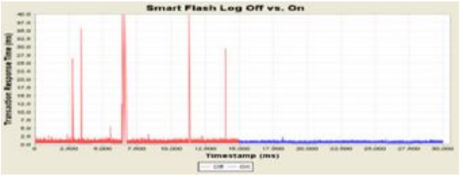
Full Rack

Oracle Exadata Database Machine

ORACLE

Exadata Smart Flash Log

Accelerate Transaction Response Time Using Flash

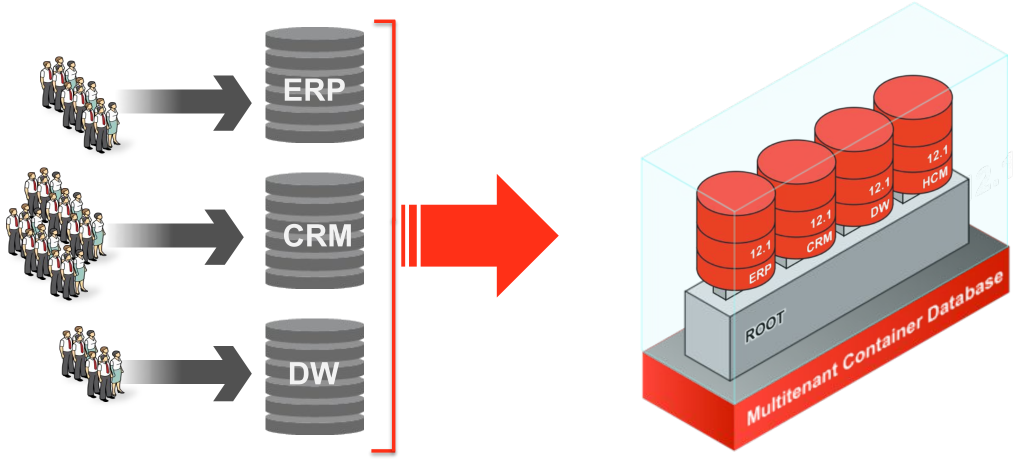


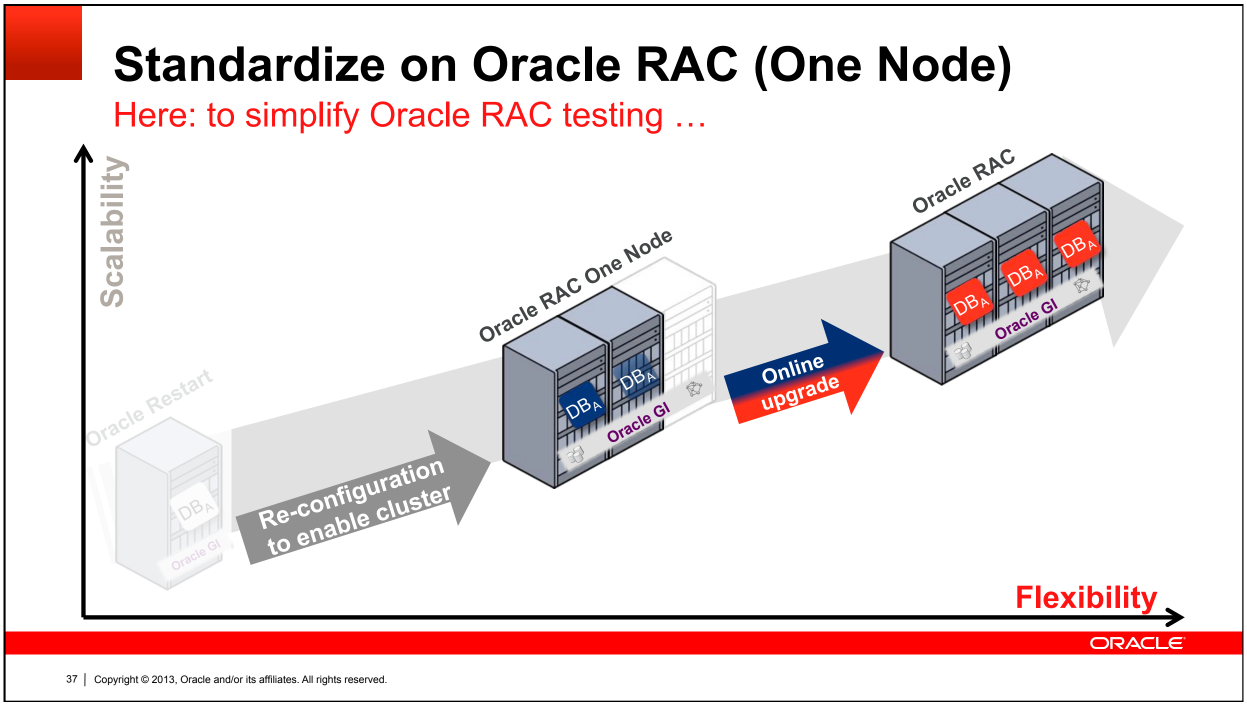
Default	Smart Log Enabled
• Choppy Response • High Outliers	• 3x faster response • Much lower outliers

- Uses Flash for Database Logs in a clever way
  - Flash is fast but has slow outliers
- Smart Flash Log feature transparently uses Flash as a parallel write cache to disk controller cache
  - Whichever write completes first wins (disk or flash)
- Better response time and more throughput
- Uses almost no flash capacity (0.1% of capacity)
- Automatically activated with 11.2.2.4.2

Oracle Real Application Testing

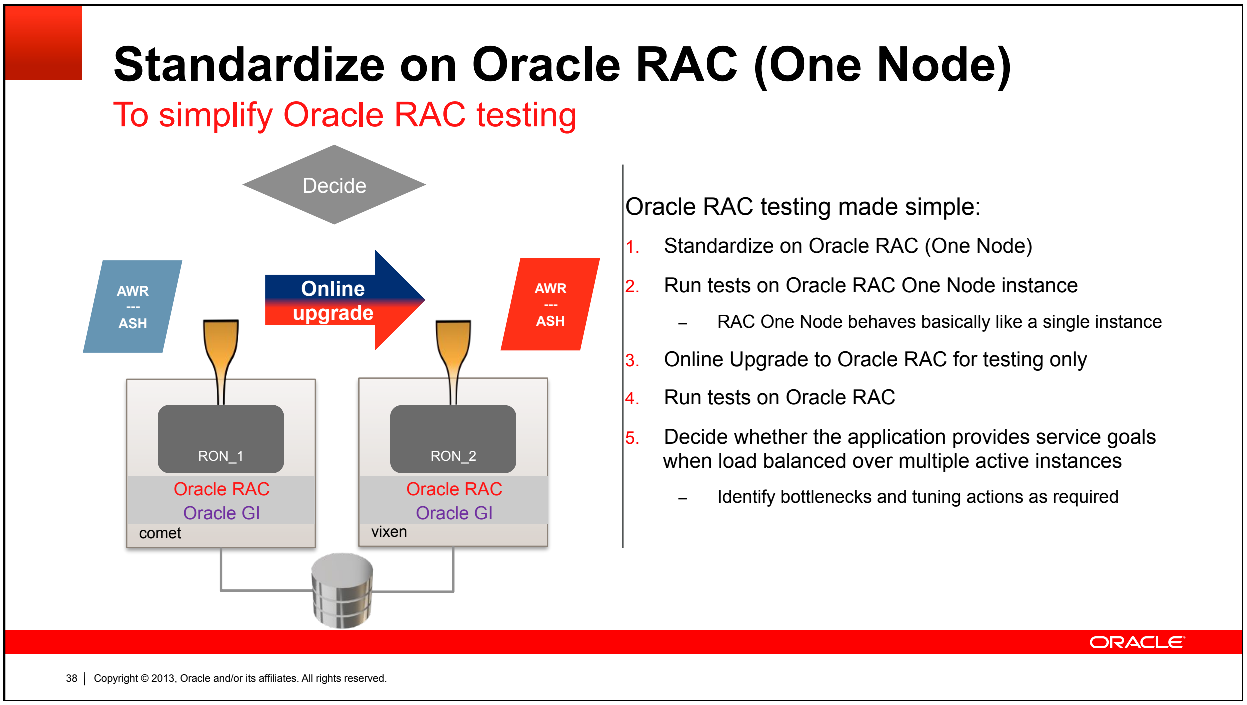
Gather and deploy workload on any database type…





Standardize on Oracle RAC (One Node)

Here: to simplify Oracle RAC testing …



Standardize on Oracle RAC (One Node)

To simplify Oracle RAC testing

Oracle RAC testing made simple:

1. Standardize on Oracle RAC (One Node)
2. Run tests on Oracle RAC One Node instance
   – RAC One Node behaves basically like a single instance
3. Online Upgrade to Oracle RAC for testing only
4. Run tests on Oracle RAC
5. Decide whether the application provides service goals when load balanced over multiple active instances
   – Identify bottlenecks and tuning actions as required

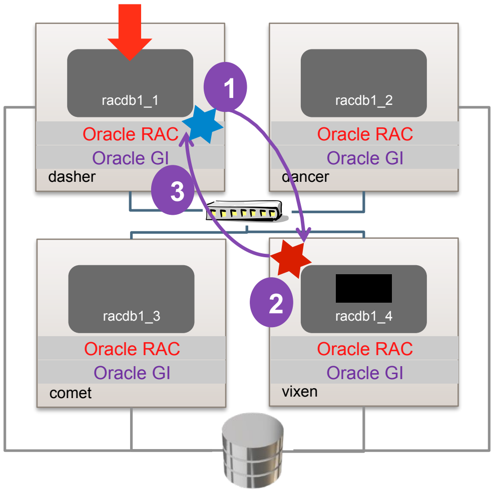
Agenda

- Introduction
- Fundamentals
- Network Recommendations
- Application Considerations
- Testing
- **Identifying Issues**
- Monitoring & Analysis
- Summary and Q&A

ORACLE

Identifying Issues

Immediate Global Cache Access – no contention



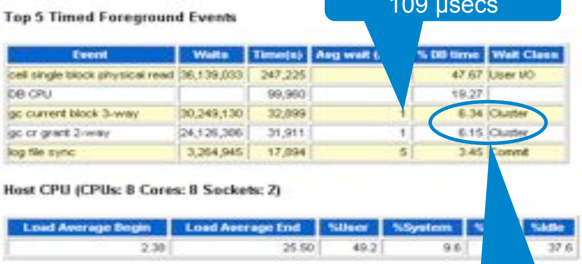
1. Shadow process on node dasher requests block from remote instance
2. LMS on remote instance sends block or grant
3. Shadow process / instance receives block

★ Shadow Process ★ LMS

ORACLE

Identifying Issues

No contention Global Cache Access – how it looks in AWR



Top 5 Timed Foreground Events

Event	Waits	Time(s)	Avg wait (	% DB time	Wait Class
cell single block physical read	36,139,033	247,225		47.67	User I/O
DB CPU		99,960		19.27	
gc current block 3-way	30,249,130	32,899	1	6.34	Cluster
gc cr grant 2-way	24,126,306	31,911	1	6.15	Cluster
log file sync	3,264,945	17,894	5	3.45	Commit

Host CPU (CPUs: 8 Cores: 8 Sockets: 2)

Load Average Begin	Load Average End	%User	%System	%	%Idle
2.30	25.50	49.2	9.6		37.6

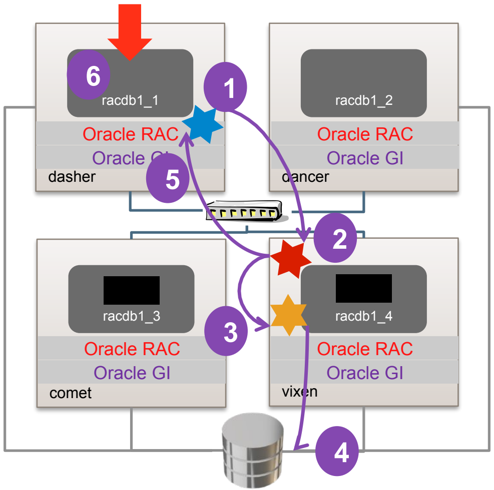
Accurate average: 109 µsecs

- Latency in Cluster has small impact
- Average Performance is good

- Factors Affecting Performance of Immediate Global Cache Access
  - Machine Load
    - Process concurrency for CPU
    - Scheduling
    - CPU utilization
  - Interconnect Bandwidth
    - Total bandwidth utilization for the database(s)
  - LMS processes
    - Real time
    - CPU busy
- **No application tuning required**

Identifying Issues

Global Cache Access – with application contention



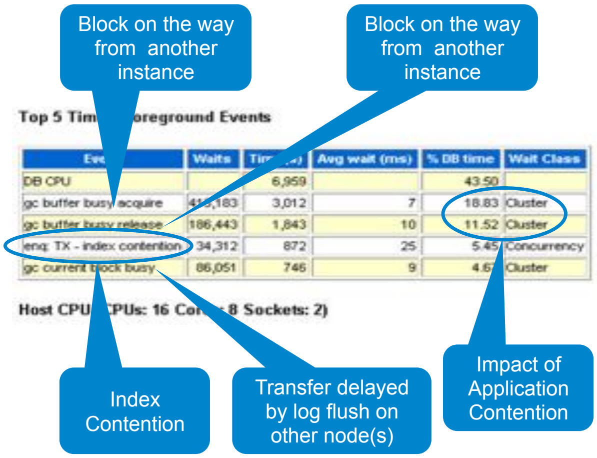
1. Shadow process on node dasher has a local cache miss and tries getting data from another instance
2. LMS on node vixen receives the request
3. LMS on node vixen posts the LGWR to write the redo for the data as the data buffer is dirty in “vixen’s cache”
4. LGWR on node vixen performs the log IO
5. Once the log IO is complete, LMS picks up the buffer handle and sends the block directly to the requesting process on node dasher.
6. The shadow process on node dasher wakes up and continues with the data

The Redo log IO latency will dominate the roundtrip time. (block is busy)

★ Shadow Process ★ LMS ★ LGWR

Identifying Issues

Global Cache Access with application contention – how it looks in AWR



Top 5 Tim oreground Events

Eve	Waits	Tim	Avg wait (ms)	% DB time	Wait Class
DB CPU		6,959		43.50	
gc buffer busy acquire	41[illegible],183	3,012	7	18.83	Cluster
gc buffer busy release	186,443	1,843	10	11.52	Cluster
enq: TX - index contention	34,312	872	25	5.45	Concurrency
gc current block busy	86,051	746	9	4.6	Cluster

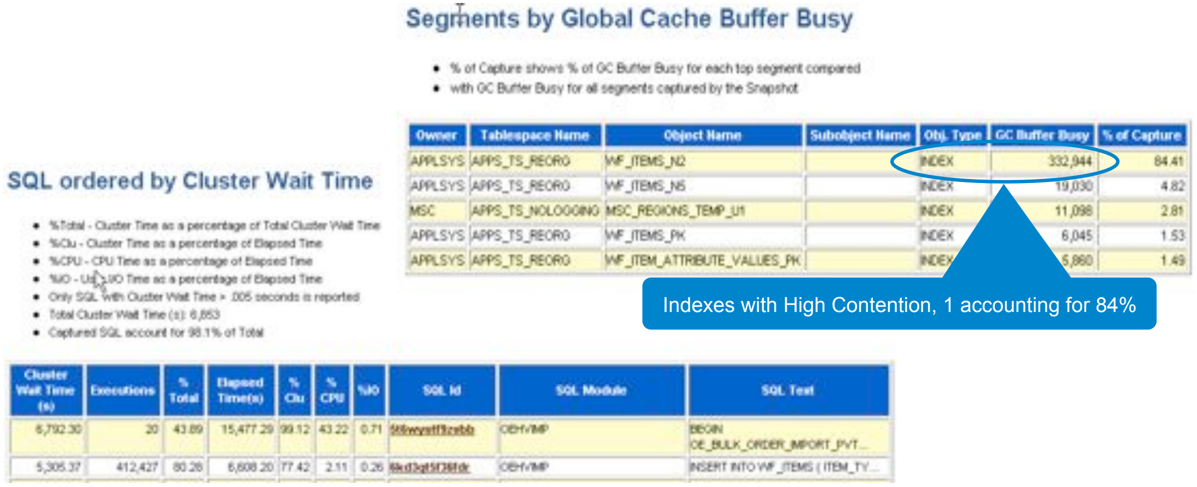
Host CPU CPUs: 16 Co 8 Sockets: 2)

- Factors Affecting Performance
  with Application Contention on Data
  - Log File IO latency
  - LGWR responsiveness
- Schema tuning may be required
  - If the application response time or throughput do not meet objectives

---

Identifying Issues

SQL and Schema Optimization: Identifying SQL incurring highest Cluster Wait Time



Segments by Global Cache Buffer Busy

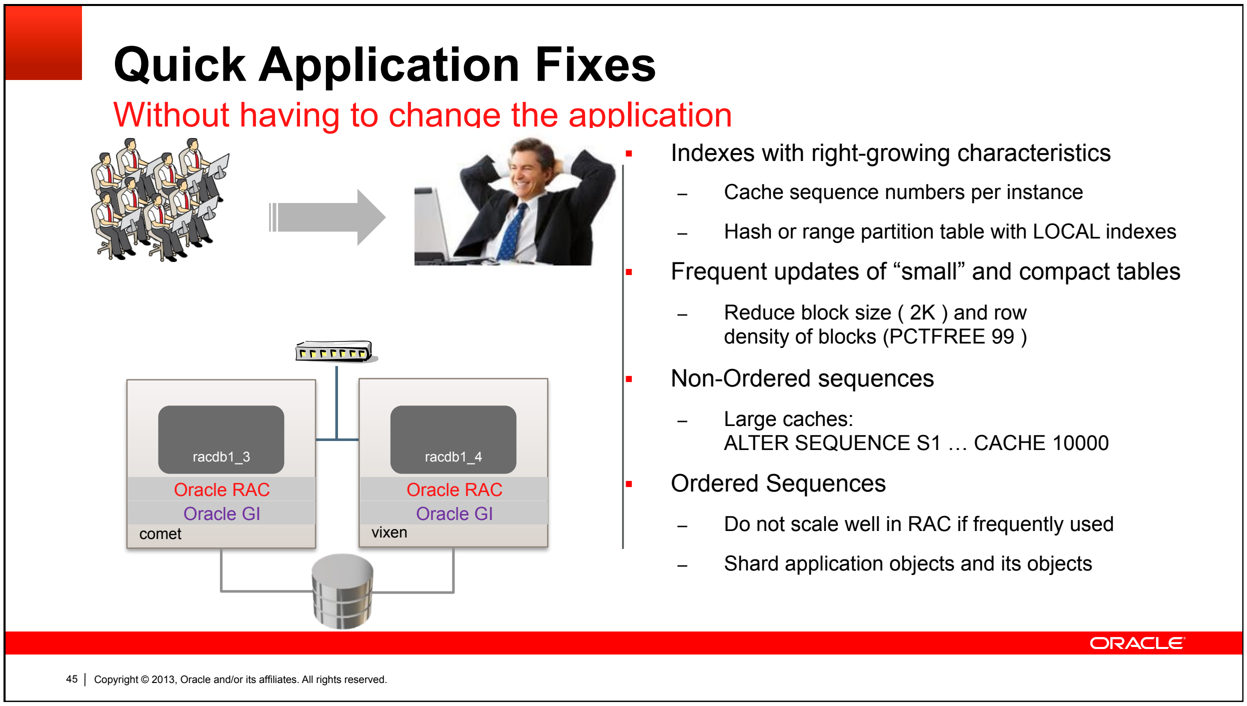
- % of Capture shows % of GC Buffer Busy for each top segment compared
- with GC Buffer Busy for all segments captured by the Snapshot

Owner	Tablespace Name	Object Name	Subobject Name	Obj. Type	GC Buffer Busy	% of Capture
APPLSYS	APPS_TS_REORG	WF_ITEMS_N2		INDEX	332,944	84.41
APPLSYS	APPS_TS_REORG	WF_ITEMS_N5		INDEX	19,030	4.82
MSC	APPS_TS_NOLOGGING	MSC_REGIONS_TEMP_U1		INDEX	11,098	2.81
APPLSYS	APPS_TS_REORG	WF_ITEMS_PK		INDEX	6,045	1.53
APPLSYS	APPS_TS_REORG	WF_ITEM_ATTRIBUTE_VALUES_PK		INDEX	5,860	1.49

SQL ordered by Cluster Wait Time

- %Total - Cluster Time as a percentage of Total Cluster Wait Time
- %Clu - Cluster Time as a percentage of Elapsed Time
- %CPU - CPU Time as a percentage of Elapsed Time
- %IO - User I/O Time as a percentage of Elapsed Time
- Only SQL with Cluster Wait Time > .005 seconds is reported
- Total Cluster Wait Time (s): 6,853
- Captured SQL account for 98.1% of Total

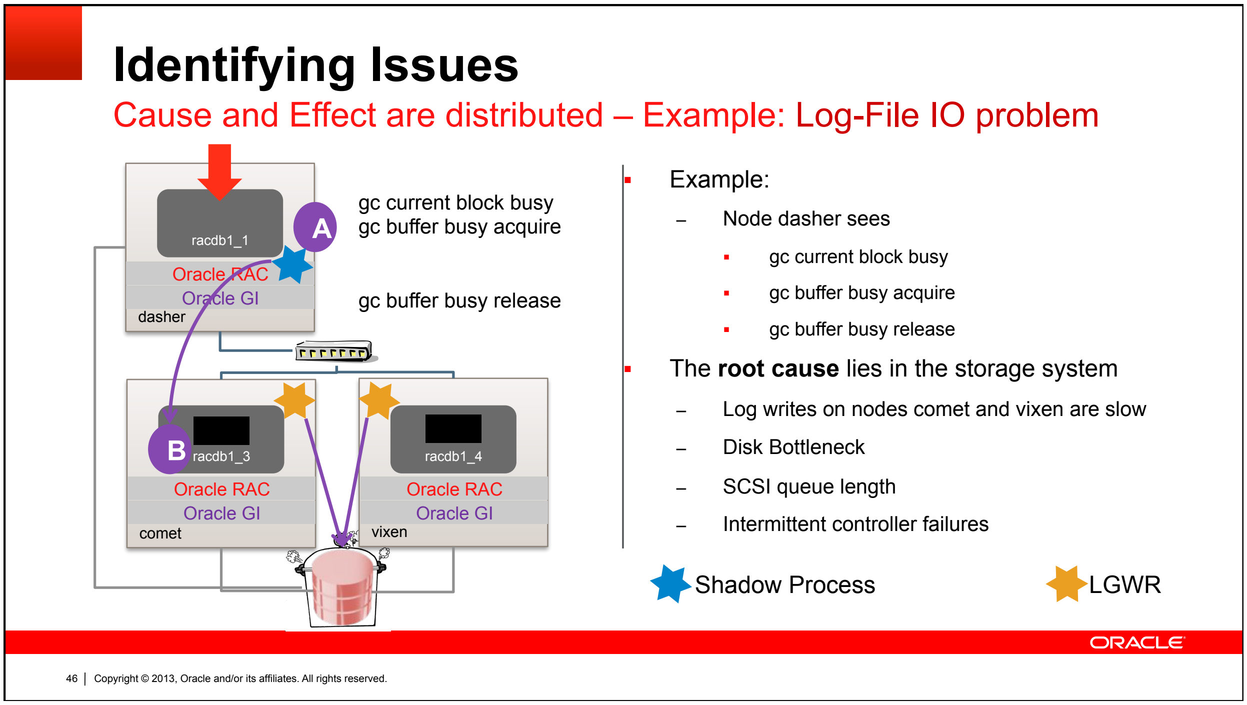
Cluster Wait Time (s)	Executions	% Total	Elapsed Time(s)	% Clu	% CPU	%IO	SQL Id	SQL Module	SQL Text
6,792.30	20	43.89	15,477.29	99.12	43.22	0.71	5t8wysff3zsbb	OEHVIMP	BEGIN OE_BULK_ORDER_IMPORT_PVT...
5,305.37	412,427	80.28	6,608.20	77.42	2.11	0.26	6kd3gt5f36fdr	OEHVIMP	INSERT INTO WF_ITEMS (ITEM_TY...



Quick Application Fixes

Without having to change the application

- Indexes with right-growing characteristics
  - Cache sequence numbers per instance
  - Hash or range partition table with LOCAL indexes
- Frequent updates of “small” and compact tables
  - Reduce block size (2K) and row density of blocks (PCTFREE 99)
- Non-Ordered sequences
  - Large caches: ALTER SEQUENCE S1 … CACHE 10000
- Ordered Sequences
  - Do not scale well in RAC if frequently used
  - Shard application objects and its objects



Identifying Issues

Cause and Effect are distributed – Example: Log-File IO problem

gc current block busy
gc buffer busy acquire

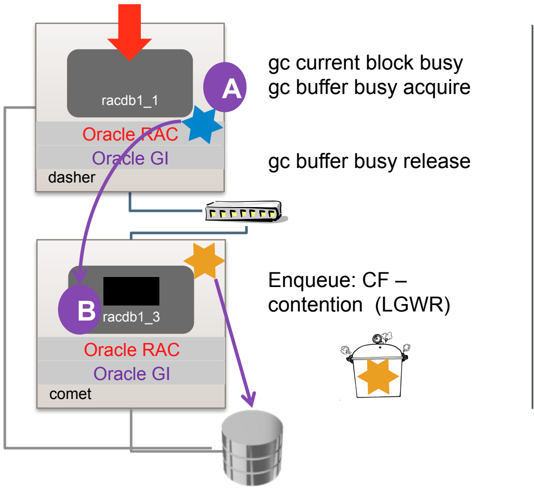
gc buffer busy release

- Example:
  - Node dasher sees
    - gc current block busy
    - gc buffer busy acquire
    - gc buffer busy release
- The **root cause** lies in the storage system
  - Log writes on nodes comet and vixen are slow
  - Disk Bottleneck
  - SCSI queue length
  - Intermittent controller failures

★ Shadow Process ★ LGWR

Identifying Issues

Cause and Effect are distributed – Example: LGWR stall



gc current block busy
gc buffer busy acquire

gc buffer busy release

Enqueue: CF –
contention (LGWR)

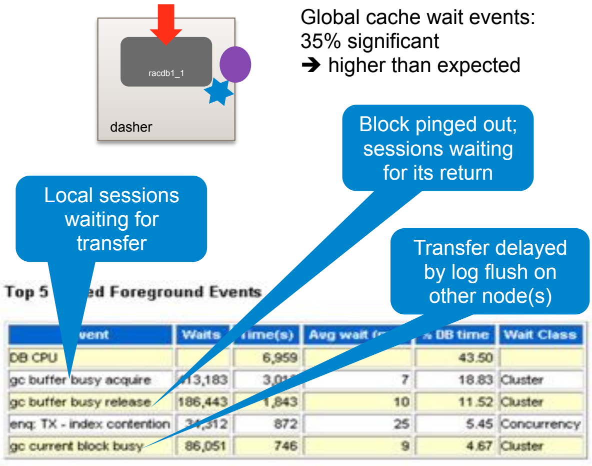
- Example:
  - Node dasher sees
    - gc current block busy
    - gc buffer busy acquire
    - gc buffer busy release
- The **root cause** is a blocked LGWR
  - LGWR waits for enq: CF – contention which may be held by another background process
  - A background process may hold a CF enqueue while updating the control file

★ Shadow Process ★ LGWR

Identifying Issues

Cause and Effect are distributed – How to read the Global Impact

Global cache wait events:
35% significant
➔ higher than expected



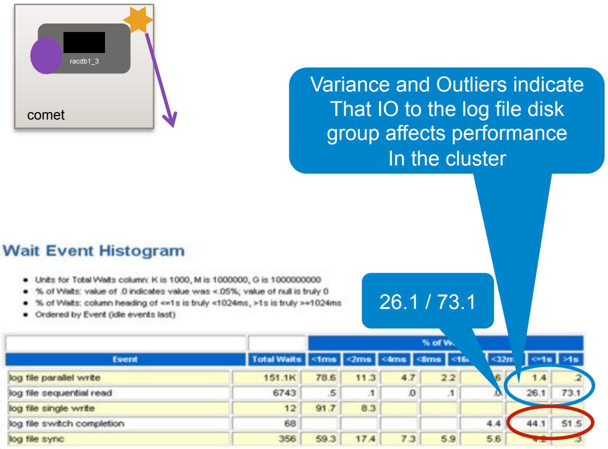
Local sessions waiting for transfer

Block pinged out; sessions waiting for its return

Transfer delayed by log flush on other node(s)

Top 5 ed Foreground Events

vent	Waits	ime(s)	Avg wait (	% DB time	Wait Class
DB CPU		6,959		43.50	
gc buffer busy acquire	13,183	3,01	7	18.83	Cluster
gc buffer busy release	186,443	1,843	10	11.52	Cluster
enq: TX - index contention	34,312	872	25	5.45	Concurrency
gc current block busy	86,051	746	9	4.67	Cluster



Variance and Outliers indicate That IO to the log file disk group affects performance In the cluster

26.1 / 73.1

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)

		% of W							
Event	Total Waits	<1ms	<2ms	<4ms	<8ms	<16	<32m	<=1s	>1s
log file parallel write	151.1K	78.6	11.3	4.7	2.2		.6	1.4	.2
log file sequential read	6743	.5	.1	.0	.1		.0	26.1	73.1
log file single write	12	91.7	8.3						
log file switch completion	68						4.4	44.1	51.5
log file sync	356	59.3	17.4	7.3	5.9		5.6	4.2	.3

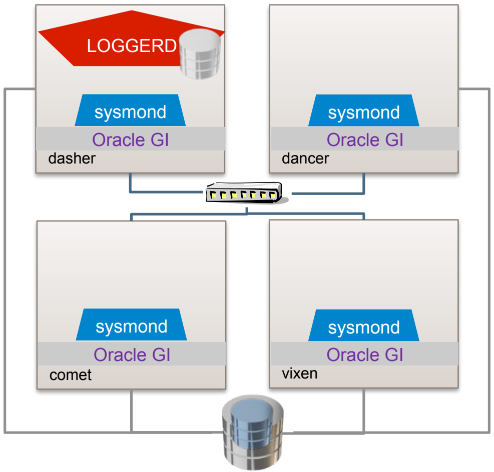
Agenda

- Introduction
- Fundamentals
- Network Recommendations
- Application Considerations
- Testing
- Identifying Issues
- **Monitoring & Analysis**
- Summary and Q&A

ORACLE

Monitoring Essentials

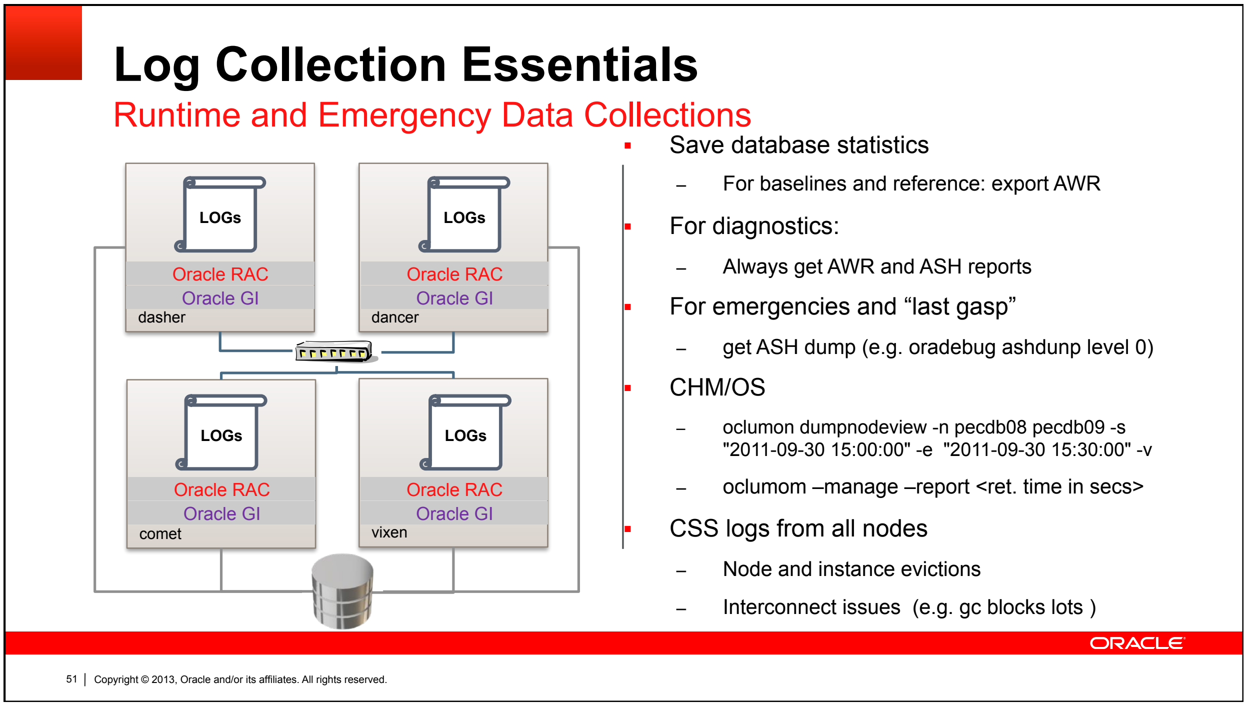
Use Cluster Health Monitor for OS (CHM/OS)



1. First failure diagnostics: Collects as much system metrics and data as feasible
2. Collects system metrics on all nodes in the cluster synchronized
3. Persistent storage of all collected data in a database
4. Integrated with Oracle Grid Infrastructure starting with 11.2.0.2 (12*c*: uses **GIMR**)
5. Robust collection in situations of resource starvation
6. Data dump or visualization

sysmond **LOGGERD**

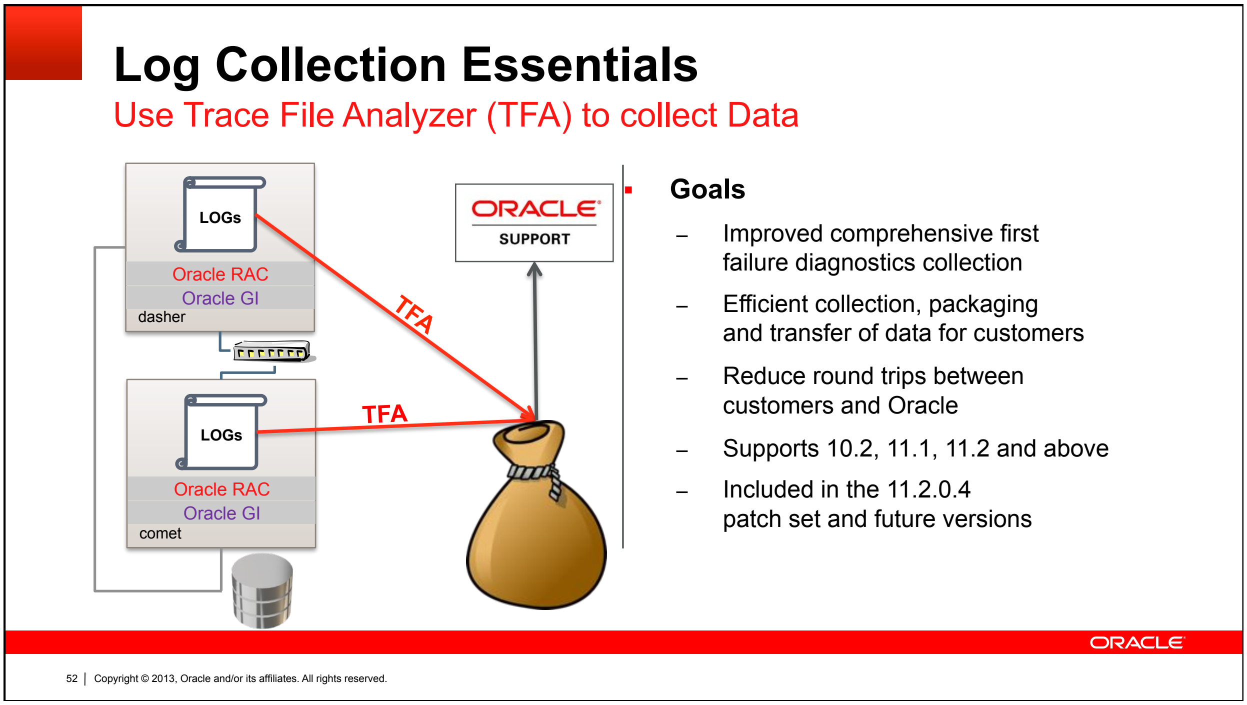
ORACLE



Log Collection Essentials

Runtime and Emergency Data Collections

- Save database statistics
  - For baselines and reference: export AWR
- For diagnostics:
  - Always get AWR and ASH reports
- For emergencies and “last gasp”
  - get ASH dump (e.g. oradebug ashdunp level 0)
- CHM/OS
  - oclumon dumpnodeview -n pecdb08 pecdb09 -s "2011-09-30 15:00:00" -e "2011-09-30 15:30:00" -v
  - oclumom –manage –report <ret. time in secs>
- CSS logs from all nodes
  - Node and instance evictions
  - Interconnect issues (e.g. gc blocks lots)



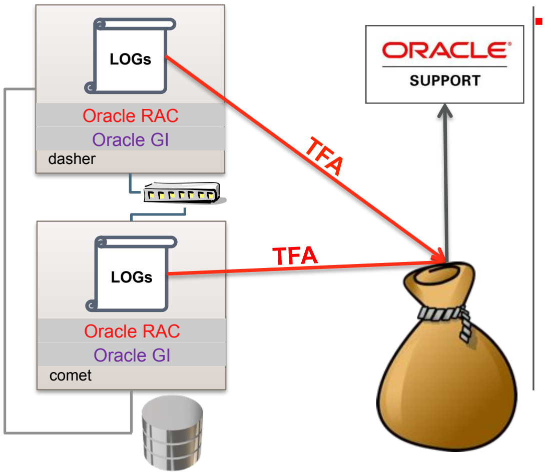
Log Collection Essentials

Use Trace File Analyzer (TFA) to collect Data

- **Goals**
  - Improved comprehensive first failure diagnostics collection
  - Efficient collection, packaging and transfer of data for customers
  - Reduce round trips between customers and Oracle
  - Supports 10.2, 11.1, 11.2 and above
  - Included in the 11.2.0.4 patch set and future versions

Log Collection Essentials

Use Trace File Analyzer (TFA) to collect Data



- **Approach**
  - Collect for all relevant components (OS, Grid Infrastructure, ASM, RDBMS)
  - One command to collect all information
  - Prune large files based on temporal criteria
  - Collect time relevant IPS (incident) packages on RAC nodes
  - Collect time relevant CHM/OS, OSWatcher data on all Oracle RAC nodes
  - On-demand (default) and Event Driven diagnostic collections
    - **TFA Collector – MOS note ID 1513912.1**

TFA – Usage Example

```
$ sudo ./tfactl diagcollect
```

One simple command

```
Sending diagcollect request to host : myhost2
Getting list of files satisfying time range [Tue Sep 03 14:17:43 PDT 2013, Tue Sep 03 18:17:43 PDT 2013]
myhost1: Zipping File: /opt/oracle/oak/oswbb/archive/oswiostat/myhost1_iostat_13.09.03.1500.dat.gz
myhost1: Zipping File: /u01/app/oracle/diag/rdbms/bill/bill1/trace/alert_bill1.log
Trimming file : /u01/app/oracle/diag/rdbms/bill/bill1/trace/alert_bill1.log with original file size : 109kB
myhost1: Zipping File: /opt/oracle/oak/oswbb/archive/oswtop/myhost1_top_13.09.03.1500.dat.gz
myhost1: Zipping File: /opt/oracle/oak/log/myhost1/oak/oakd.log
Trimming file : /opt/oracle/oak/log/myhost1/oak/oakd.log with original file size : 9.2MB
myhost1: Zipping File: /u01/app/11.2.0.3/grid/log/myhost1/gipcd/gipcd.log
myhost1: Zipping File: /u01/app/11.2.0.3/grid/log/myhost1/agent/ohasd/oraagent_grid/oraagent_grid.log
Trimming file : /u01/app/11.2.0.3/grid/log/myhost1/agent/ohasd/oraagent_grid/oraagent_grid.log with original filesize 4.3MB
myhost1: Zipping File: /var/log/messages
…Truncated for brevity
myhost1: Zipping File: /opt/oracle/oak/oswbb/archive/oswslabinfo/myhost1_slabinfo_13.09.03.1800.dat
Collecting ADR incident files..
Total Number of Files checked : 10543
Total Size of all Files Checked : 3.9GB
Number of files containing required range : 68
Total Size of Files containing required range : 129MB
Number of files trimmed : 10
Total Size of data prior to zip : 144MB
Saved 63MB by trimming files
Zip file size : 8.6MB
Total time taken : 47s.

Logs are collected to:
/opt/oracle/tfa/tfa_home/repository/collection_Tue_Sep_3_18_17_24_PDT_2013_node_all/myhost1.tfa_Tue_Sep_3_18_17_24_PDT_2013.zip
/opt/oracle/tfa/tfa_home/repository/collection_Tue_Sep_3_18_17_24_PDT_2013_node_all/myhost2.tfa_Tue_Sep_3_18_17_24_PDT_2013.zip
```

OS Watcher files

Pruning

Relevant files only

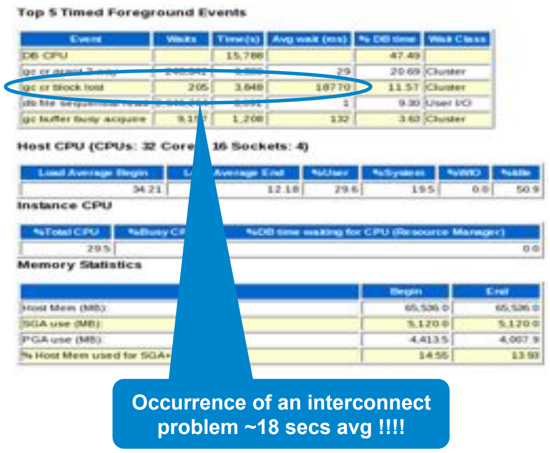
ADR Incident files

144MB pruned and compressed down to 8.6MB

47 seconds! – 1 command, 2 nodes, 4 databases, ASM, Clusterware, OS

Analysis Essentials

Time Series Analysis – AWR Reports are not enough due to average…



Top 5 Timed Foreground Events

Event	Waits	Time(s)	Avg wait (ms)	% DB time	Wait Class
DB CPU		15,788		47.49	
gc cr grant 2-way	[illegible]	[illegible]	29	20.69	Cluster
gc cr block lost	205	3,848	18770	11.57	Cluster
db file sequential read	[illegible]	[illegible]	1	9.30	User I/O
gc buffer busy acquire	[illegible]	1,208	132	3.63	Cluster

Host CPU (CPUs: 32 Core[illegible] 16 Sockets: 4)

Load Average Begin	Load Average End	%User	%System	%WIO	%Idle
34.21	12.18	29.6	19.5	0.0	50.9

Instance CPU

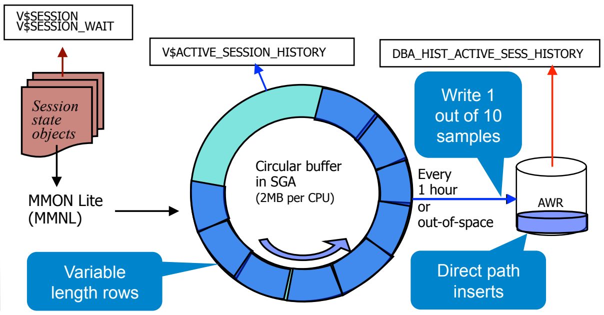
%Total CPU	%Busy CPU	%DB time waiting for CPU (Resource Manager)
29.5	[illegible]	0.0

Memory Statistics

	Begin	End
Host Mem (MB):	65,536.0	65,536.0
SGA use (MB):	5,120.0	5,120.0
PGA use (MB):	4,413.5	4,007.9
% Host Mem used for SGA+	14.55	13.93

Occurrence of an interconnect problem ~18 secs avg !!!!

- **Solution**
  - ASH reports



V$SESSION
V$SESSION_WAIT

Session state objects

MMON Lite (MMNL)

V$ACTIVE_SESSION_HISTORY

Circular buffer in SGA (2MB per CPU)

DBA_HIST_ACTIVE_SESS_HISTORY

Write 1 out of 10 samples

Every 1 hour or out-of-space

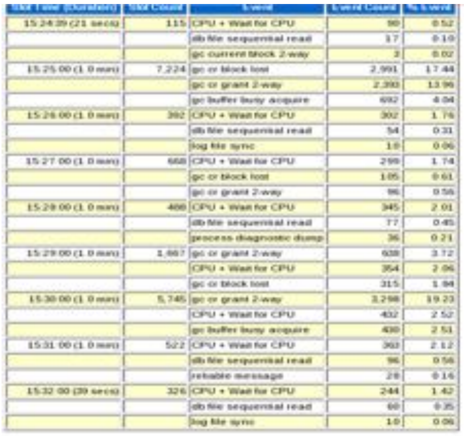
AWR

Variable length rows

Direct path inserts

Analysis Essentials

Time Series Analysis – AWR Reports are not enough …

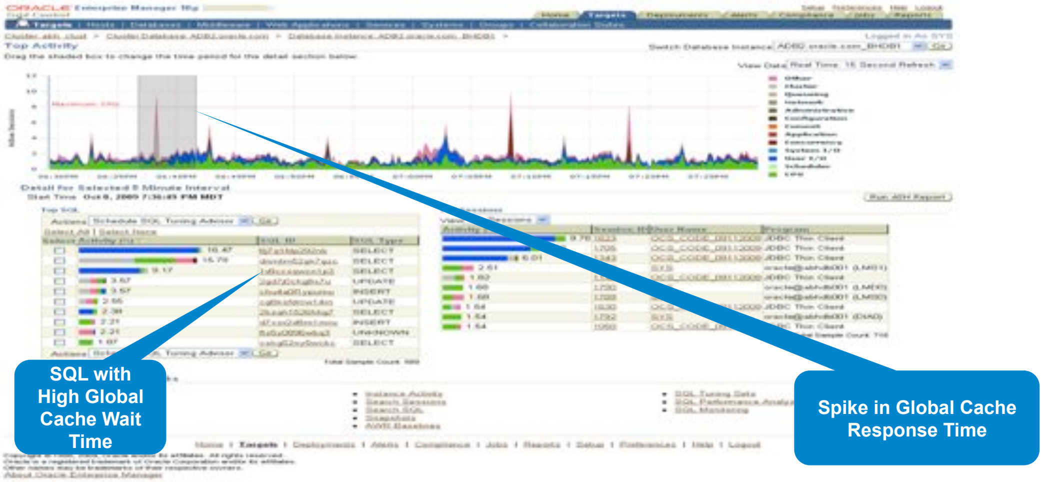


Slot Time (Duration)	Slot Count	Event	Event Count	% Event
15:24:39 (21 secs)	115	CPU + Wait for CPU	90	0.52
		db file sequential read	17	0.10
		gc current block 2-way	3	0.02
15:25:00 (1.0 min)	7,224	gc cr block lost	2,991	17.44
		gc cr grant 2-way	2,393	13.96
		gc buffer busy acquire	692	4.04
15:26:00 (1.0 min)	392	CPU + Wait for CPU	302	1.76
		db file sequential read	54	0.31
		log file sync	10	0.06
15:27:00 (1.0 min)	668	CPU + Wait for CPU	299	1.74
		gc cr block lost	105	0.61
		gc cr grant 2-way	96	0.56
15:28:00 (1.0 min)	488	CPU + Wait for CPU	345	2.01
		db file sequential read	77	0.45
		process diagnostic dump	36	0.21
15:29:00 (1.0 min)	1,887	gc cr grant 2-way	638	3.72
		CPU + Wait for CPU	354	2.06
		gc cr block lost	315	1.84
15:30:00 (1.0 min)	5,745	gc cr grant 2-way	3,298	19.23
		CPU + Wait for CPU	432	2.52
		gc buffer busy acquire	430	2.51
15:31:00 (1.0 min)	522	CPU + Wait for CPU	363	2.12
		db file sequential read	96	0.56
		reliable message	28	0.16
15:32:00 (39 secs)	326	CPU + Wait for CPU	244	1.42
		db file sequential read	60	0.35
		log file sync	10	0.06

```
2011-10-03 15:25:25.662: [    CSSD][39]clssnmPollingThread:
node pecdb07 (1) is impending reconfig, flag 199692, misstime
15626
2011-10-03 15:25:25.662: [    CSSD][39]clssnmPollingThread:
node 2, pecdb08, ninfmisstime 267, misstime 267, skgxnbit 4,
vcwmisstime 0, syncstage 0
2011-10-03 15:25:25.662: [    CSSD][39]clssnmPollingThread:
local diskTimeout set to 27000 ms, remote disk timeout set to
27000, impending reconfig status(1)
2011-10-03 15:25:25.663: [    CSSD]
[35]clssnmvDHBValidateNCopy: node 1, pecdb07, has a disk
HB, but no network HB, DHB has rcfg 207057058, wrtcnt,
17468860, LATS 534607234, lastSeqNo 17462672, uniqueness
1317671435, timestamp 1317673524/229299074
…
2011-10-03 15:30:21.811: [    CSSD]
[35]clssnmvDHBValidateNCopy: node 1, pecdb07, has a disk
HB, but no network HB, DHB has rcfg
```

Analysis Essentials

Temporary Cluster Wait Spike



SQL with High Global Cache Wait Time

Spike in Global Cache Response Time

Agenda

- Introduction
- Fundamentals
- Network Recommendations
- Application Considerations
- Testing
- Identifying Issues
- Monitoring & Analysis
- **Summary and Q&A**

Summary and Q&A

Some more sessions of interest

Monday

- Understanding Oracle RAC Internals – 12:15 PM – 1:15 PM; Moscone South – 104
- Come to see the "Meeting Service Objectives with Oracle Real Application Clusters" Demo – Moscone South, Left / SL-021

Tuesday

- Oracle RAC 12c Best Practices – 10:30 AM - 11:30 am ; Moscone South – 104
- Oracle Flex Cluster: Optimized Resource Management for the Cloud – 12:00 PM - 1:00 PM; Moscone South – 104

Wednesday

- Oracle RAC Practical Performance Management and Tuning – 11:45 AM - 12:45 PM Moscone South – 104
- Building a Test/Dev Infrastructure with Oracle Automatic Storage Management Cluster File System 3:30 PM - 4:30 PM Moscone South - 200

Thursday

- Next-Generation Oracle Automatic Storage Management – 12:30 PM - 1:30 PM Moscone South – 103
- Maximize Availability by Using Database Services with Oracle RAC – 12:30 PM - 1:30 PM Moscone South – 103
- The Oracle Multitenant Option Meets Oracle RAC – 3:30 PM - 4:30 PM Moscone South – 103

ORACLE