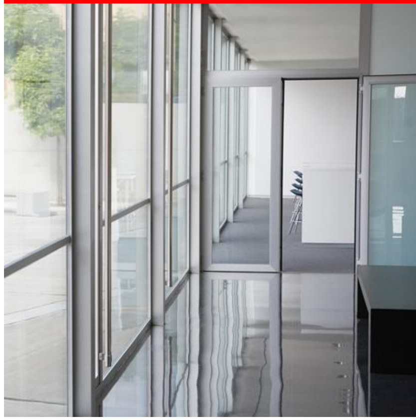
ORACLE®

Clusters Accelerated - a Study

Sumanta Chatterjee

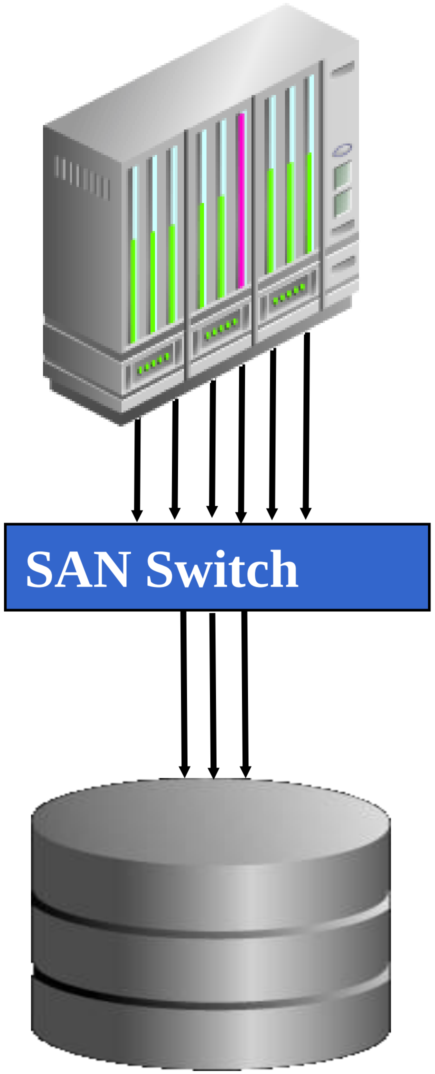
The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development,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

Program Agenda



- Motivations for Exadata
- Hardware components
- Smart Software too!
- Next Frontiers

Traditional DB Configuration



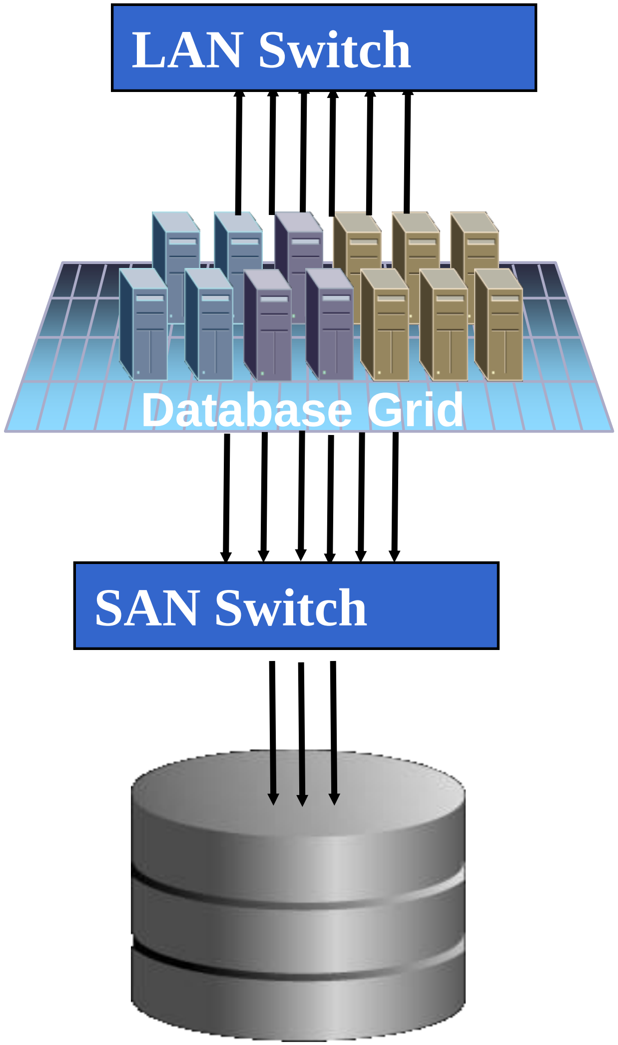
Monolithic SMP Server

- High-Cost Scale-Up
- Limited Scalability

Monolithic Storage Array

- High-Cost Scale-Up

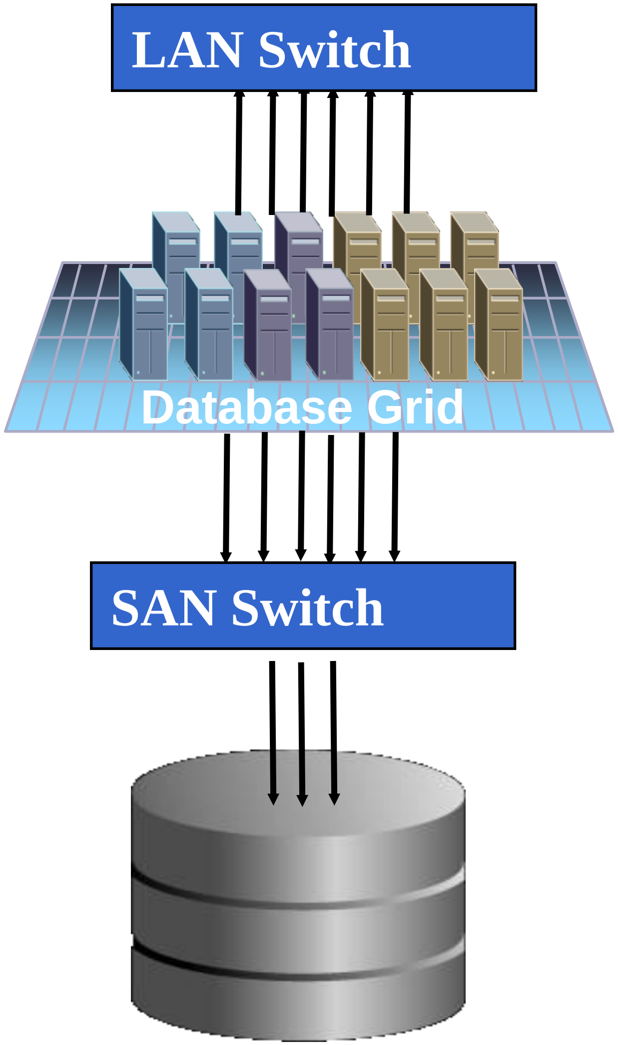
Oracle Brings Grid to Server Tier



- Real Application Clusters (RAC)
  - Released in Oracle9i in 2001
- Still the ONLY solution that enables:
  - Scale-out using low cost servers
    - Single system image
  - Highly available architecture
  - Runs real-world applications unchanged
    - Including ultra-complex applications like ERP, CRM, HR, etc.
- Today, thousands of production customers

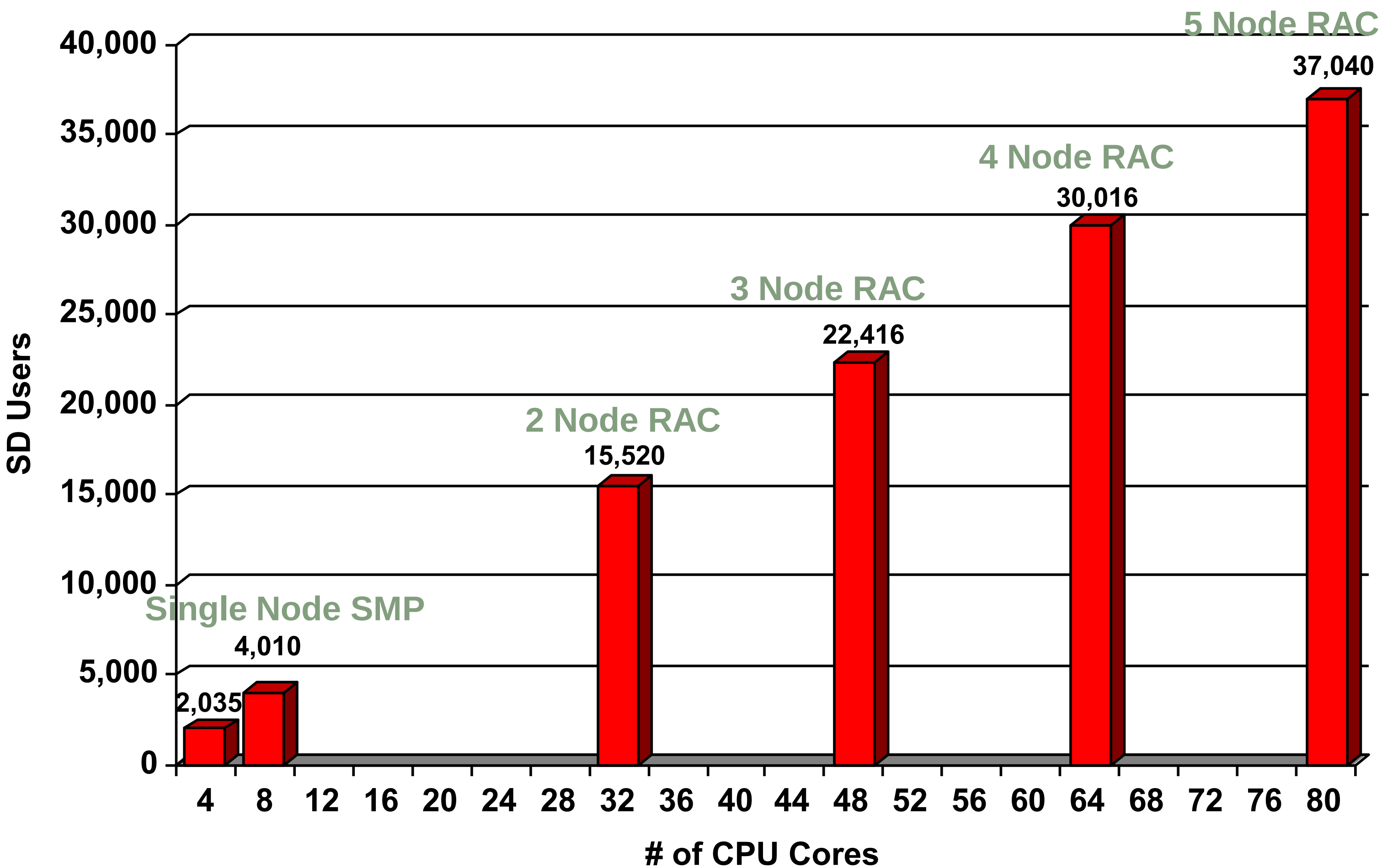
ORACLE

Works Great for OLTP Applications



- Cache Fusion reduces I/Os and allows nodes to serve requests from local cache
  - Low bandwidth across LAN
  - Gigabit Ethernet bandwidth is sufficient
- OLTP queries satisfied using indexed access
  - Disk requests are single-block random I/Os
    - Disk seeks are the storage bottleneck
  - Low usage of storage network
    - One SAN link can serve 50K IO per sec.
    - More than enough even for huge system
  - Storage arrays with hundreds of disks common

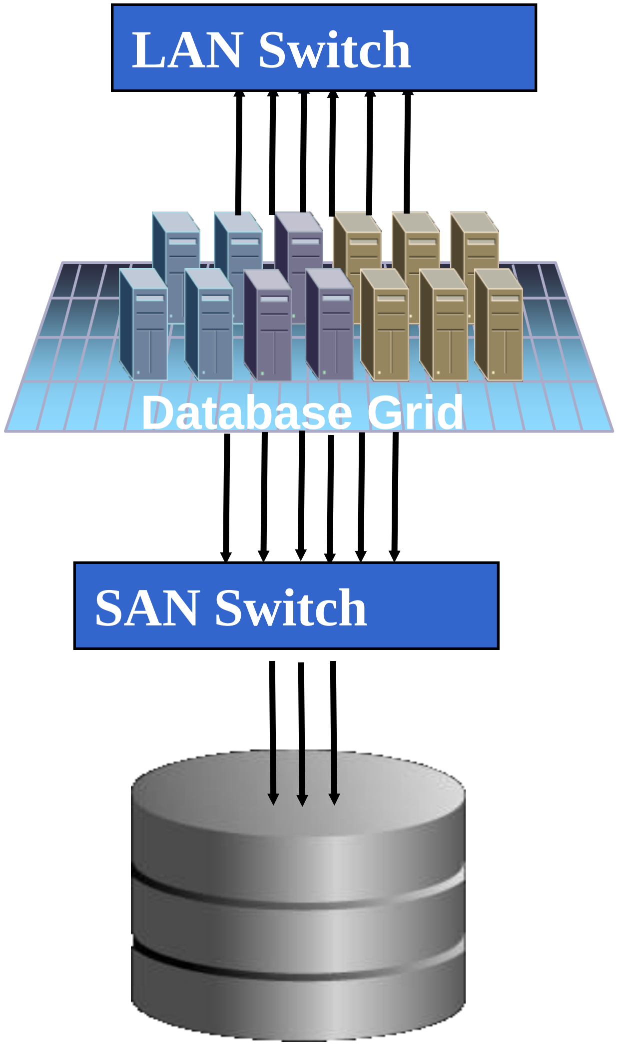
Best OLTP Scalability and Performance
World Record SAP SD Benchmark Results



Near Perfect Scaling across SMP and Cluster

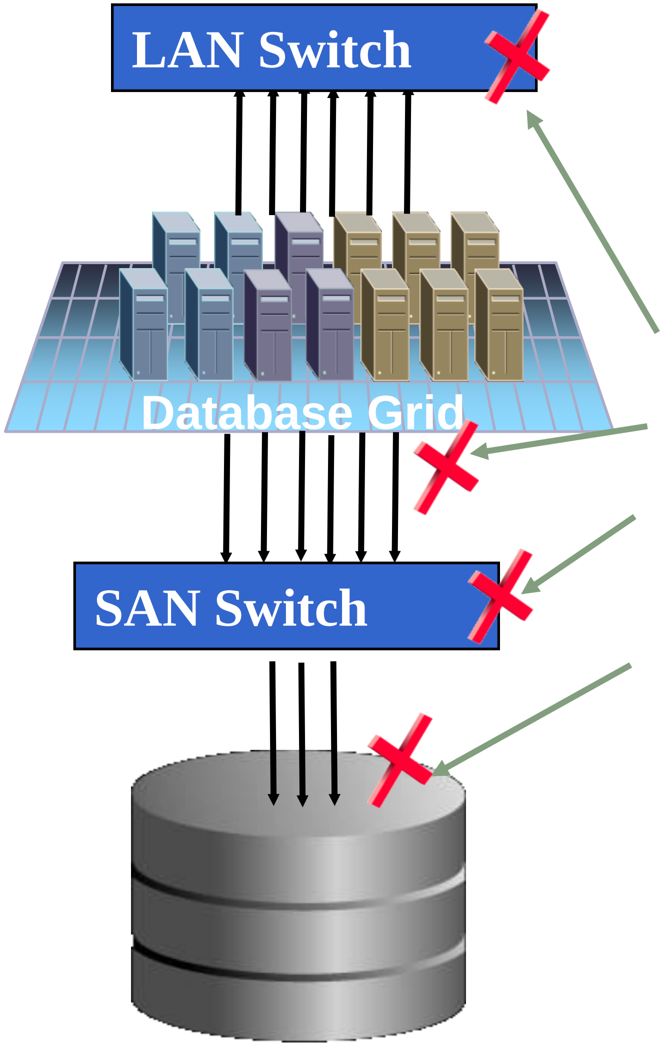
ORACLE

Data Warehouse Workloads



- Two kinds of Data Warehouse workloads:
- Predictable many user workloads
  - Can be highly optimized using smart database technologies such as
    - Partitioning, Bitmap Indexing, Join indexing, OLAP cubes, Materialized Views, Result Caches, etc.
  - Smart technologies give large speedup so hardware needs are kept controlled
- Unpredictable workloads needing huge data scans
  - Require very high performance scans and joins of huge amounts of data
  - Still uses some smart technologies like partitioning but to a lesser extent

Bottlenecks for Huge Data Scans



- Want many Gigabytes per second of I/O
  - Large systems want 10's of Gigabytes (not bits) of bandwidth to hundreds of disks
- Many bottlenecks prevent this today
- LAN switches can't handle load of large joins
- Server nodes need many SAN adapters
- Storage switch cost and SAN complexity increase dramatically
- Large storage arrays cannot deliver bandwidth of hundreds of disks
  - Bottleneck on storage heads and connections to SAN switches
- Result is poor performance for huge data scans

Grid Based Storage Accelerator

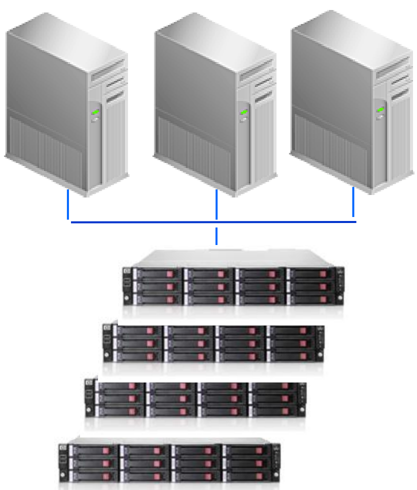
A Storage Cell is an appliance containing

- Disks
- Intel x86 server
- Infiniband Network

Oracle provides Hardware and Software

Scale-Out Architecture

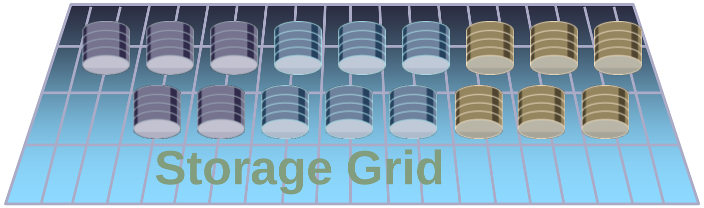
- Scalable to hundreds of cells
- Dynamic Provisioning
- Fault Tolerant
- Load Balancing



- Cell software serves blocks to hosts, but can also perform smart data scans
  - Move code to data for large scans
  - Large speedup for data warehouses, reporting, backup
- Grid Consolidation
  - Consolidated pool of storage improves utilization and flexibility
  - Transaction/Job level Quality of Service
- Mission Critical Availability and Protection
  - Disaster recovery, backup, point-in-time recovery, data validation

ORACLE

Exadata Changes the Game



- A new approach and architecture
- Delivers database intelligence and modern Grid technology in the storage tier
  - Using state of the art industry standard hardware
- Eliminates all bottlenecks preventing high performance data scans
- Dramatic performance and price improvement for data warehouse
- Simplicity of appliance seamlessly integrated with the database

ORACLE

Storage Cell

- A Storage Cell is a database storage appliance
  - Does not support non-database storage
- A cell ships complete with all hardware and software components pre-installed
- Runs Oracle cell software, Oracle Infiniband protocol, Oracle Enterprise Linux, and Sun hardware management software
- Absolutely no custom hardware
  - All parts are off the shelf high-volume
- Equipped with 4 X 96 GB Sun Flash F20 PCI-e cards

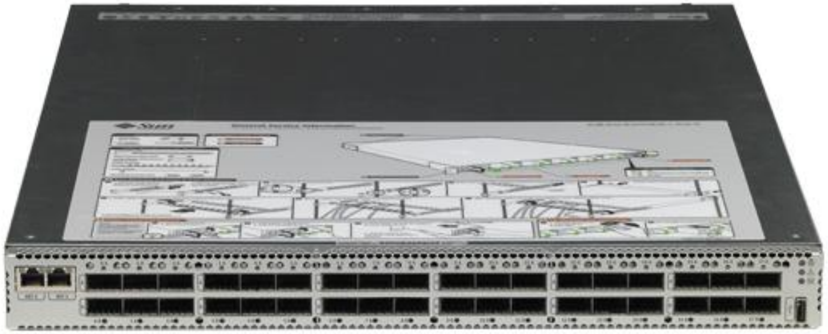
ORACLE

Fully Optimized Configuration

- Configuration and software is highly optimized for fast data processing
  - Capable of streaming and processing full bandwidth of all disks
- CPUs are collocated with disks to allow offloading of high data throughput operations such as table scans of compressed tables
- Components carefully matched to avoid bottlenecks
  - Aggregate disk bandwidth
    - Over 800MB/sec
  - Disk controller bandwidth
    - Over 1000 MB/sec
  - CPU power – 2 Six-core Intel 64 bit
    - Capable of processing over 3000 MB/sec of compressed table data
  - Network
    - Infiniband bandwidth sufficient to stream disk bandwidth

ORACLE

InfiniBand Network



- Unified InfiniBand Network
  - Storage Network
  - RAC Interconnect
  - External Connectivity (optional)
- High Performance, Low Latency Network
  - 80 Gb/s bandwidth per link (40 Gb/s each direction)
  - SAN-like Efficiency (Zero copy, buffer reservation)
  - Simple manageability like IP network
- Protocols
  - Zero-copy Zero-loss Datagram Protocol (ZDP RDSv3)
    - Linux Open Source, Low CPU overhead (Transfer 3 GB/s with 2% CPU usage)
  - Internet Protocol over InfiniBand (IPoIB) for external connectivity
    - Looks like normal Ethernet to host software (tcp/ip, udp, http, ssh,…)

ORACLE

InfiniBand Network

- Uses Sun Datacenter 36-port Managed QDR (40Gb/s) InfiniBand switches
  - Runs subnet manager and automatically discovers network topology
  - Only one subnet manager active at a time
  - 2 “leaf” switches to connect individual server IB ports
  - 1 “spine” switch in Full Rack and Half Rack for scaling out to additional Racks
- Database Server and Exadata Servers
  - Each server has Dual-port QDR (40Gb/s) IB HCA
  - Active-Passive Bonding – Assign Single IP address
    - Performance is limited by PCIe bus, so active-active not needed
  - Connect one port from the HCA to one leaf switch and the other port to the second leaf switch for redundancy

ORACLE

Flash in the Exadata Storage Server

- Flash vs Disk tradeoff
  - 10X-100X better performance but 10X more expensive
- Exadata Goal is get performance of Flash but price point of disk.
- 4 x 96GB Sun F20 Flash Accelerator PCIe Cards in each storage server
  - 384 GB of Flash per Storage Server
- Choice of PCIe form factor over SSD for performance reasons
  - No disk controller bottleneck

ORACLE

 |

Fully Assembled
Exadata Hardware Architecture

Scaleable Grid of industry standard servers for Compute and Storage

- Eliminates long-standing tradeoff between Scalability, Availability, Cost

Database Grid

- 8 Dual-processor x64 database servers

or

- 2 Eight-processor x64 database servers

Intelligent Storage Grid

- 14 High-performance low-cost storage servers

- 100 TB High Performance disk,
  or
  336 TB High Capacity disk
- 5.3 TB PCI Flash
- Data mirrored across storage servers

InfiniBand Network

- Redundant 40Gb/s switches
- Unified server & storage network

ORACLE

X2-2 and X2-8 Full Rack

	X2-8 Full Rack	X2-2 Full Rack
Database Servers	**2**	**8**
Cores (Total)	128 (2.26 GHz)	96 (2.93 GHz)
Memory (Total)	2048 GB	768 GB
1 GbE Ports (Total)	16	32
10 GbE Ports(Total)	16	16
InfiniBand Switches	**3**	
Exadata Storage Servers	**14**	
Flash (Total)	5.3 TB	
Raw Storage (Total)	100 TB or 336 TB	
Raw Disk Data Bandwidth	**25 GB/s***	
Raw Flash Data Bandwidth	**50 GB/s**	
Flash IOPS (8k Reads)	**1,000,000**	

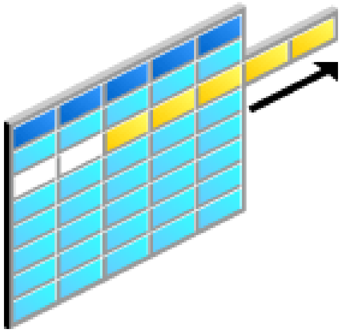
* Using High Performance 15K RPM disks

Exadata Software

SOFTWARE.
HARDWARE.
COMPLETE.

Exadata Smart Scans

- Exadata cells implement smart scans to greatly reduce the data that needs to be processed by database
  - Only return relevant rows and columns to database
  - Offload predicate evaluation
- Data reduction is usually very large
  - Column and row reduction often decrease data to be returned to the database by 10x
- Join Filtering
  - Bloom filters used for join filtering in storage



Encrypted Smart Scan

- Features
  - Smart Scans on Encrypted Tablespaces
  - Smart Scans on Encrypted Columns
  - Takes advantage of AES-NI in X5600 hardware
- Benefits
  - CPU utilization on database node dramatically improves due to offload
  - Less data shipped to database nodes
  - Query fully encrypted database by moving decryption from software to storage cell hardware

Smart Incremental Backup

– Block Change Tracking maintains the set of blocks changed with a bitmap that has 1 bit per 32k and does a large (approx 1M) IO if needed.
– When a large IO for incremental backup is done at exadata, exadata filters out most of the data and returns only the data that needs to be a part of the incremental backup.

Change Tracking File Content for 1MB

0 0 1 0 1 0 1 1 0 0 1 0 1 0 1 1 0 0 1 0 1 0 1 1 0 0 1 0 1 0 0 0 0

Smart Incremental backup Request

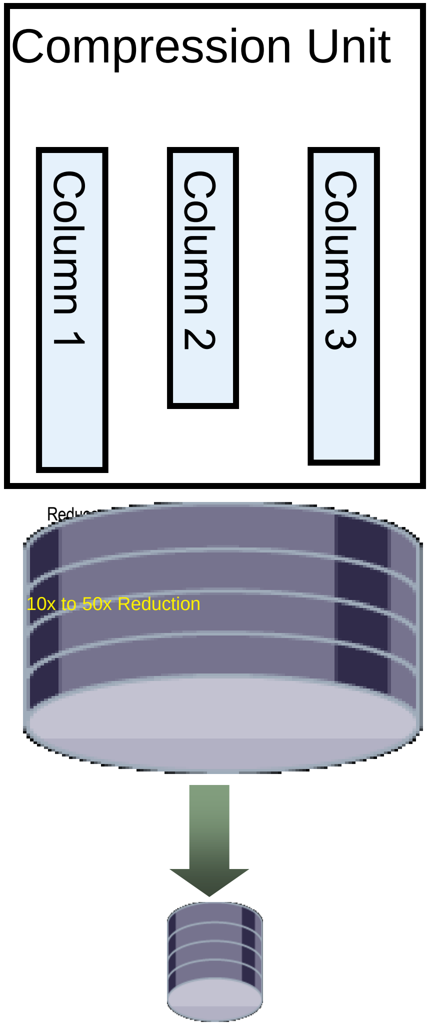
ORACLE

Smart File Creation

- Files created by the database are initialized
- Full blocks initialized by database and written to storage
- With Exadata, only metadata is sent by Database to Exadata storage
- Initialization is done by the Exadata storage server software on the drives
- Tremendous reduction in IO between database to storage

ORACLE

Hybrid Columnar Compression



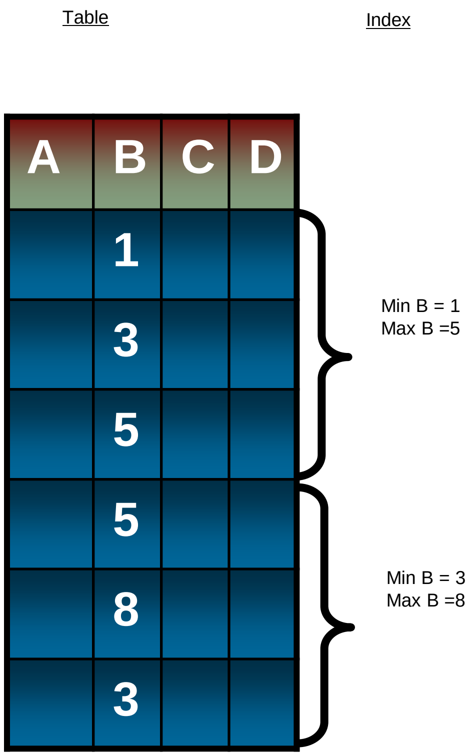
- Useful for data that is bulk loaded and queried
- Tables are organized into Compression Units (CUs)
  - CUs are larger than database blocks
  - Usually around 32K
- Within Compression Unit, data is Organized by Column instead of by row
  - Column organization brings similar values close together, enhancing compression

Hybrid Columnar Compression

- Modes of Compression
  - Query Mode – 10x average savings
  - Archive Mode – 15x average savings
- Smart Scans on HCC tables in Exadata
  - Column Projection
  - Decompression
  - Row Filtering

ORACLE

Storage Index – Basic example



- Exadata Storage Indexes maintain summary information about table data in memory
  - Store MIN and MAX values of columns
  - Typically one index entry for every MB of disk
- Eliminates disk I/Os if MIN and MAX can never match “where” clause of a query
- Completely **automatic and transparent**
- Example: Select count(*) from table where b = 1;

Storage Index with Partitions Example

Orders Table

Order#	Order_Date Partitioning Column	Ship_Date	Item
1	2007	2007	
2	2008	2008	
3	2009	2009	

- Queries on Ship_Date do not benefit from Order_Date partitioning
  - However Ship_date and Order# are highly correlated with Order_Date
  - e.g. Ship dates are usually near Order_Dates and are never less
- Storage index provides partition pruning like performance for queries on Ship_Date and Order#
  - Takes advantage of ordering created by partitioning or sorted loading

ORACLE

Storage Index with Joins Example

```
Select count(*) from fact, dim
where fact.m=dim.m and dim.name='Camry'
```

Dimension

Name	M
Accord	1
Camry	3
Civic	5
Prius	8

Bloom filter constructed
with min/max for M

Perform IO and
apply bloom filter

Skip IO
Due to Storage Index

Fact

A	M	C	D
	1		
	3		
	5		
	5		
	5		
	5		

Smart Flash Cache

- Understands different types of I/Os from database
  - Skips caching I/Os to mirror copies
  - Skips caching backups
  - Skips caching data pump I/O
  - Skips caching tablespace formatting
  - Resistant to table scans
  - Control File Reads and Writes are cached
  - File header reads and writes are cached
  - Data Blocks and Index blocks are cached

ORACLE

Smart Flash Cache Keep Directive

- DBA can enforce that an object is kept in flash cache
  - `ALTER TABLE calldetail STORAGE (CELL_FLASH_CACHE KEEP)`
- Can be set like other storage clause values
  - At table level, partition level, during creation time etc.
- Table scans on objects marked with cell_flash_cache keep run through the flash cache
  - Disk bandwidth full rack – 25GB/s
  - Flash bandwidth full rack – 50GB/s

Smart Flash Cache Benefits

- Smart Flash Cache w/ HCC compressed table
  - Converts 5TB of flash into 50TB of flash cache
- Flash Cache does not use space for redundancy
  - Better utilization of premium storage
- Scans through flash cache take advantage of disks too!
  - Disks 25GB/s Flash 50GB/s
  - Flash cache > 69GB/s (featured in exadata demo)
- 1.5 Million 8k I/Os per second on a full rack at sub millisecond latency

ORACLE

Flash Cache Advantages

- Reacts much quicker than storage migration of LUNs
- You can control movement/placement at the database level and specify logical tables/indexes to put in cache (not LUNs)
- Since it is a cache, you do not need to add extra redundancy. This increases the effective size by a lot.
- Combined with Exadata compression you get a lot more effective flash
- Exadata flash is accessed through InfiniBand providing higher throughput
- We use a scale-out architecture that actually provides the full benefit of flash without bottlenecking in the controller.

CPU Resource Management

- Manage CPU usage of workloads within a database
  - Fine-grained, application-level scheduling
  - Control number of processes that are running at any moment in time
  - Resource plan specifies
    - Minimum, guaranteed CPU per workload
    - Maximum CPU utilization per workload
  - Critical for consolidation
  - Prevents system instability caused by excessive loads

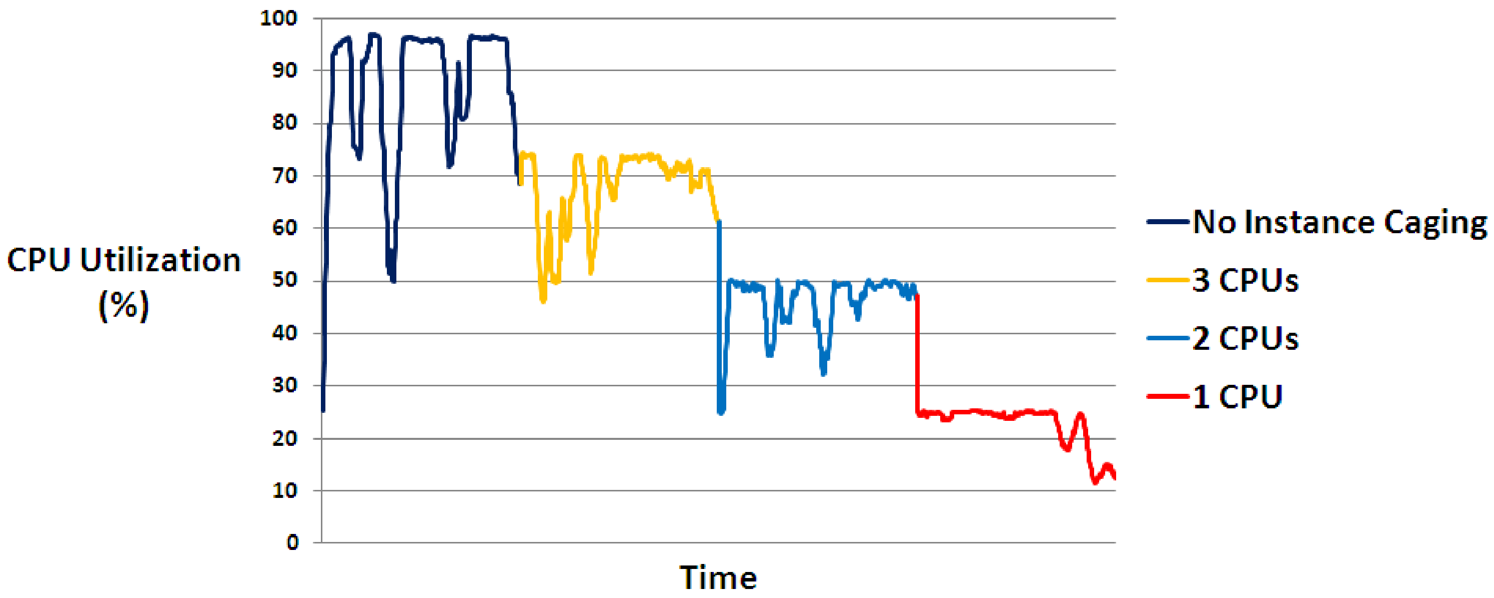
<u>Day Time Plan</u>

	Allocation	Limit
OLTP	70%	
Reports	20%	50%
Maintenance	10%	

ORACLE

Instance Caging

- Oracle feature for “caging” or limiting the amount of CPU that a database instance can use at any time
– Fine-grained application-level scheduling
- Important tool for server consolidation
  - Easy to configure
  - Available on all platforms

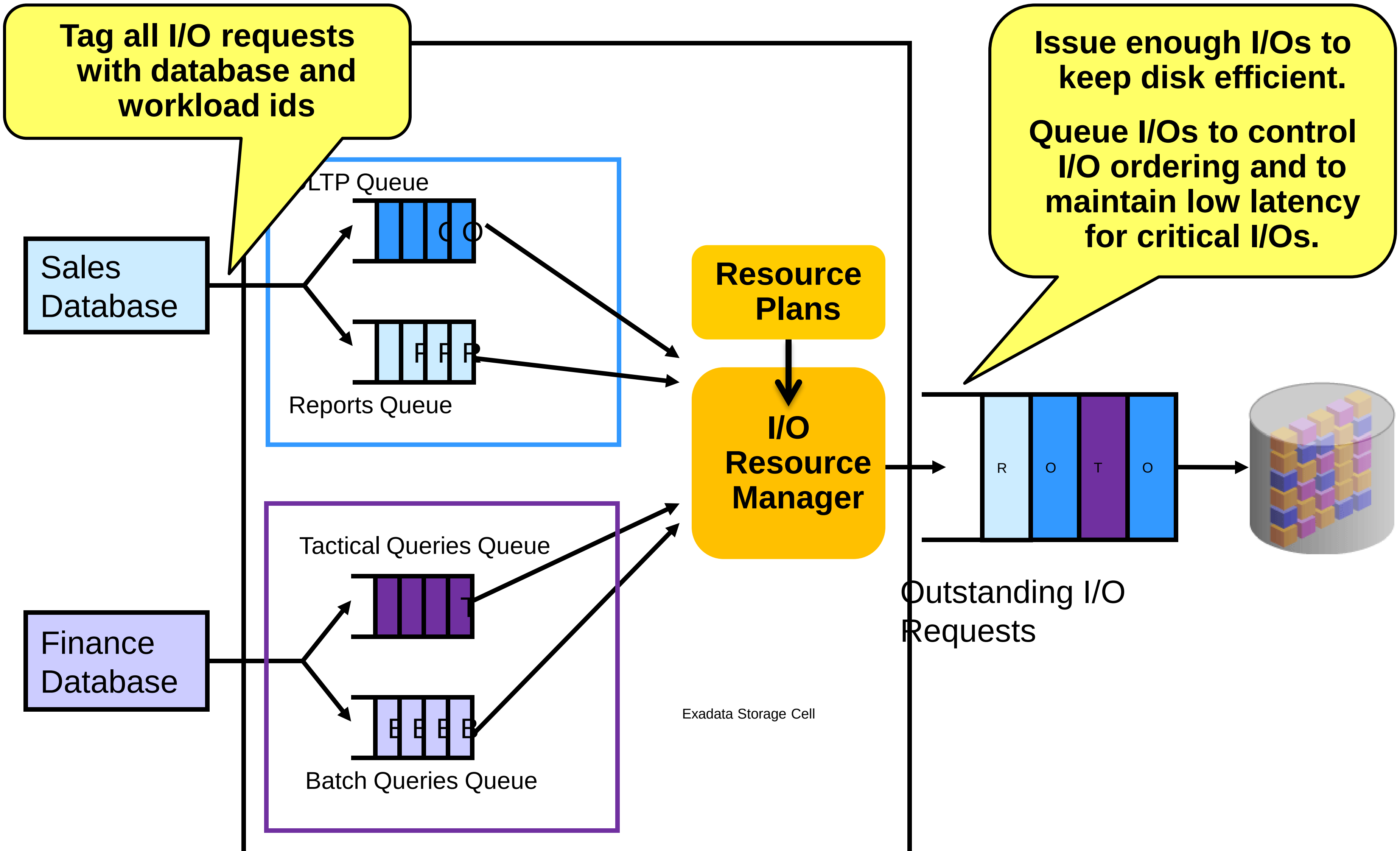


Exadata I/O Resource Manager

- Manages how databases and workloads share disk bandwidth
  - Storage can be shared for consolidated or cloud environments
  - Storage can be shared by real-time and low-priority applications
- Policy specified through a Resource Plan
  - Specifies min and max resource allocations per database
  - Specifies min and max resource allocations per workload
- Fine-grained scheduling is key!
  - Tightly control disk latency by managing disk queue lengths
  - Key for good OLTP performance with concurrent DSS
  - DSS workloads can capitalize on lulls in critical I/O loads
- Hardware and software integration is key!
  - Storage must know who and why each I/O was issued

ORACLE

Exadata I/O Resource Manager

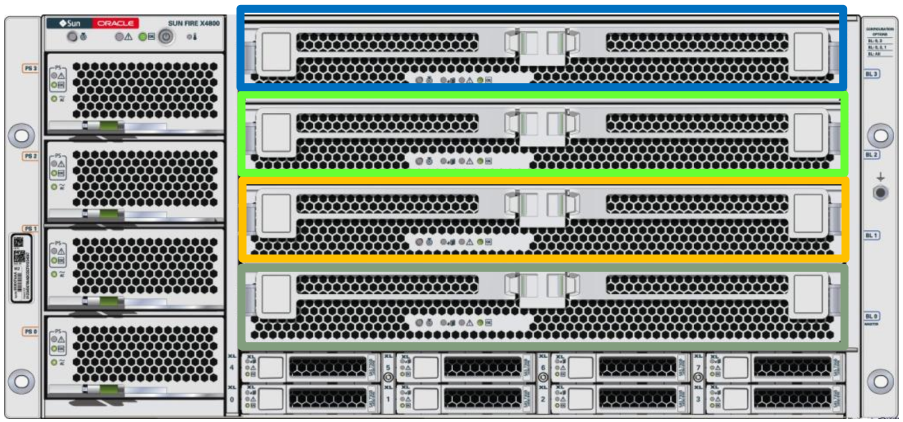


NUMA – Headache?

- In my experience, if you are not careful, in a NUMA systems, you could end up with unexpected scalability
- What happened to the old SMP?
- Check what type of cache snooping algorithms are being used
- Revisit all of your major shared memory algorithms
- Can be very useful as a consolidation platform
- NUMA affinity is key

ORACLE

NUMA Affinity on X2-8



- Affinity is inherent the hardware design
- Software takes advantage of it
- Process running on red node uses memory from red node
- Process running on red node has affinity to send its traffic through red Infiniband card since latency is lower
- New automatically generated file cellaffinity.ora describes the affinity to Oracle Database

Let’s look at latency

The Operating Environment is:

- 1000s of tasks.
- 100s of cores.
- 10s of nodes.
- Gigabytes of network bandwidth.
- Multiple O/Ss.
- General time sharing O/S scheduler.

Problems we see

- Very high scheduling latencies.
- Very high network overhead / cost.
- Very expensive to use messages for distributed synchronization.
  – Atomically update remote host memory state.

Very high scheduling latencies

- Deep scheduling queues with 100s of waiters
  - Head of line blocking..
- Poor SMP efficiency
  - Cross CPU scheduling / scaling / latency
- Bugs! Scheduler changes continuously.

ORACLE

15 usecs for 500 Byte msg
One Way Latency

1. sendmsg() and ctx switch out.
2. Wire time (< 1usec).
3. Interrupt + driver tasking msg + wake target task.
4. Scheduler latency-- Find a CPU to run
5. Ctx switch in and recvmsg().
6. 30 usecs round trip !

ORACLE

15 usecs 500 Byte msg One Way Latency– reality check!

- Grows to hundreds of usecs on loaded system – 400 – 500 usecs. Outliers of 100s of milliseconds
- Reducing wire latency is not solution.
  - Going from 20 to 40 to 100 gbits *does not* help.
- Kills cluster scaling.
- We need consistent fixed cost operations.

New Oracle IPC

- RDS is great for bulk data transfers
  - relatively long operation times for I/Os 100 usecs
  - Event based model. Client yields while waiting..
- New IPC model projecting 75% reduction in latencies

  With udp, rds, rc, xrc transports

  50% shorter code paths.
- rc + xrc are based on libibverbs.

ORACLE

RAC requires much lower latencies

- Moving to MPI like interfaces for some operations.
  - User mode busy wait for completion
  - Stalls hardware thread execution… and polls for CQ completions.
- Introducing Remote Memory Access Model
  - Clients operate on a declared data structure.. As if structure is always local..
  - Transparent where location of structure is
  - Uses RDMA read, write, + atomics..

ORACLE

Oracle RMA model

- Barrier / data sync operations
  - Dirty read, Dirty write
  - Consistent Read - remote host can be updating local memory while remote reader is reading it..
  - Consistent Update – local host can be reading data while remote host is updating it..
  - Serialized Read / Write – lock, read, write, unlock
- Atomics
  - Fetch add, Compare swap, variable sized data..
  - Transactional.. If updater dies in middle of update.. Update is rolled back..

Oracle RMA model

- Dirty Read – 2 usecs.. 256 bytes..
- Consistent Read – 3 usecs for 256 bytes..
  - Uses CRC for small data structures.
  - Uses sequence of 3 RDMA fenced reads for larger structs..
- PGAS model for "fixed" RMA object space.
  - All nodes contribute chunk of shmem address space.
  - Same size on all nodes.
  - All nodes can access shmem on all other nodes
  - All objects are at same offset in PGAS.. For each node

ORACLE

Oracle MSGQ

- Based on RDMA writes to remote rings
- Allocate ring in private or shared memory
- Client has access to remote ring and inserts variable sized messages with RDMA writes..
- 4 usecs update latency for 1KB messages

Q&A

Hardware and Software

ORACLE

Engineered to Work Together

ORACLE®