

1

Oracle Exadata Storage Server Overview

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

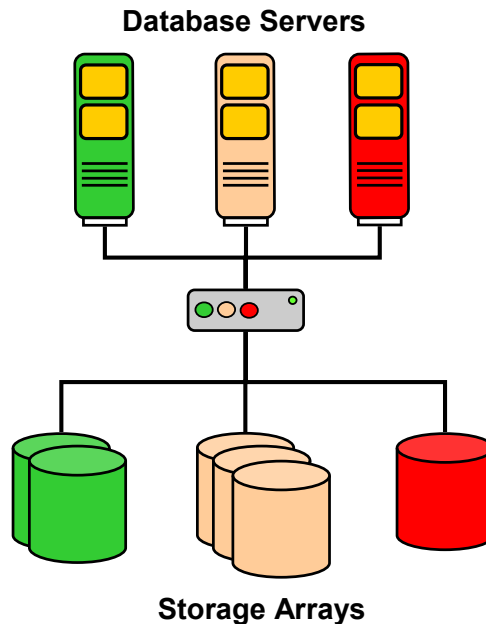
Objectives

After completing this lesson, you should be able to:

- Contrast the Exadata storage architecture with traditional shared storage offerings
- Describe the hardware components of the Sun Oracle Exadata Storage Server and Sun Oracle Database Machine
- Detail the capabilities of the Exadata Storage Server
- Describe the main advantages of using Exadata compared to traditional storage servers

ORACLE

Traditional Database Storage Deployment



ORACLE

1 - 3

Copyright © 2009, Oracle. All rights reserved.

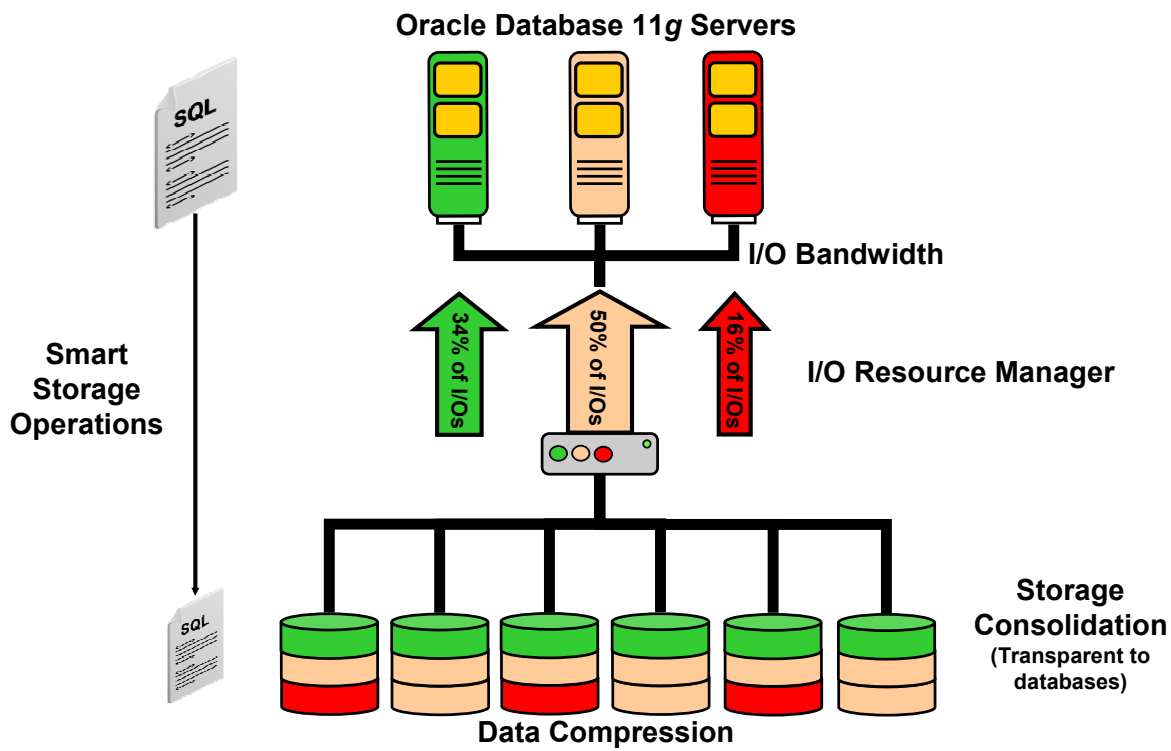
Traditional Database Storage Deployment

The nature of traditional storage products encourages inefficient deployments of storage for each database in the information technology (IT) infrastructure.

The above graphic illustrates the traditional deployment approach for multiple databases. Each database has an isolated allocation of storage resources and its bandwidth is limited by the hardware allocated to it. The isolation and dedication of hardware resources to individual databases can simultaneously lead to unused space and unused I/O bandwidth for some databases, and over committed bandwidth with insufficient free space in others. The right balance is almost never achieved because real-world workloads are very dynamic. As a result, managing and configuring traditional storage products for use with a database is also time consuming and labor intensive.

Large storage arrays are used today for many mission-critical deployments. These large storage arrays must be partitioned and have their bandwidth and space allocated across the databases and applications sharing the storage array. In addition, given that these storage arrays house vast quantities of mission-critical data, they must be highly engineered, and consequentially very expensive, to deliver high levels of reliability and availability. This results in very high total cost of ownership when traditional large storage arrays are used in real-world deployments. Enterprise-class storage arrays are not only costly to procure, but they also require highly specialized skills to manage and maintain.

Exadata Storage Deployment



1 - 4

Copyright © 2009, Oracle. All rights reserved.

ORACLE

Exadata Storage Deployment

The above graphic illustrates the general deployment approach with Exadata storage:

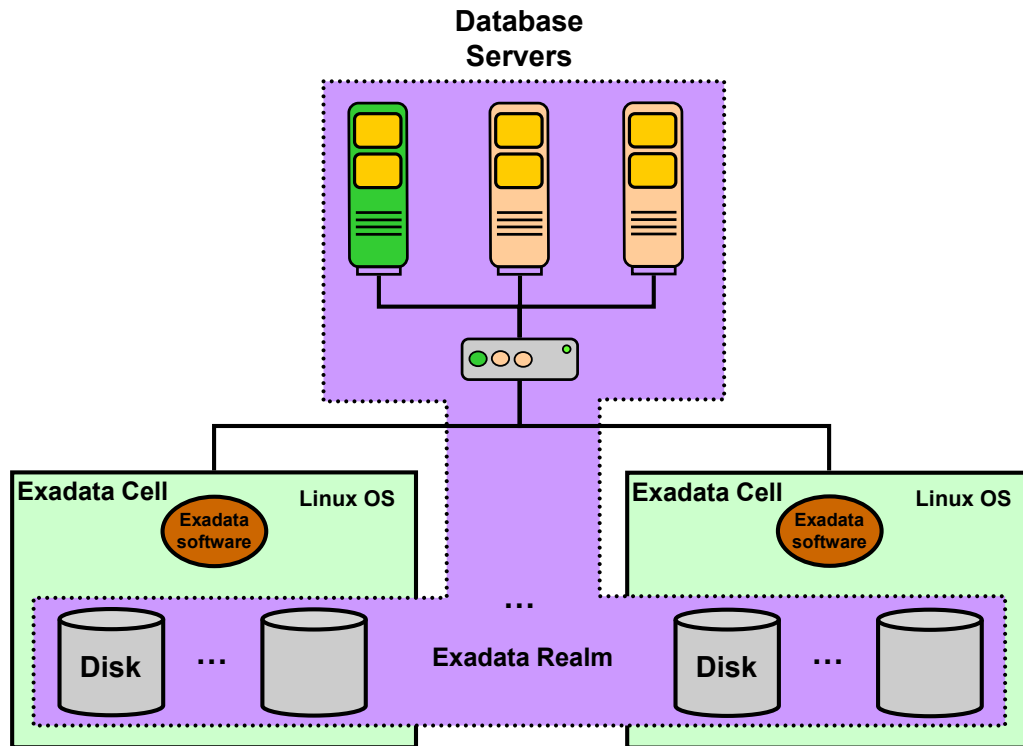
- You can use Exadata to consolidate your storage environment. Using Exadata, multiple databases can use storage from a single pool. Exadata uses Oracle Automatic Storage Management (ASM) to evenly distribute the storage load for every database across every available disk in the storage pool. Every database can use all available disks to achieve superior I/O rates. Exadata V2 requires the use of the Oracle Database 11g Release 2 (at least 11.2.0.1). Exadata works equally well with single-instance or Oracle Real Application Clusters (RAC) databases. Users and database administrators use the same tools and knowledge they are already familiar with. Exadata sets a new standard for modular enterprise-capable grid storage. Being based on industry-standard components and technologies, Exadata is inexpensive to deploy. In addition, tight integration with the full suite of Oracle Database high-availability features, ensures that the reliability and integrity needs of mission-critical environments are met.
- A key advantage of Exadata is the ability to offload some database processing to the Exadata storage server. With Exadata, the database can offload single table scan predicate filters and projections, join processing based on bloom filters, along with CPU intensive decompression and decryption operations. This ability is known as SQL processing offload or smart scan. In addition to smart scan, Exadata has other smart storage capabilities including the ability to offload incremental backup optimizations, file creation operations and more. This approach

Exadata Storage Deployment (continued)

yields substantial CPU, memory and I/O bandwidth savings in the database server resulting in potentially massive performance improvements.

- Exadata V2 introduces Exadata Hybrid Columnar Compression. This feature provides very high levels of data compression implemented inside the Exadata storage server. Exadata Hybrid Columnar Compression allows the database to reduce the number of I/Os required to scan a table. For example, for data with a compression ratio of 10 to 1, the I/Os required to scan the data are reduced 10 to 1 as well.
- Exadata ensures that I/O resources are made available whenever, and to whichever, database needs them based on priorities and policies that you can define. The Database Resource Manager (DBRM) has been enhanced to manage intradatabase and interdatabase I/O resource usage to ensure that your defined service-level agreements (SLAs) are met when multiple databases share Exadata storage.
- Finally, even for queries that do not use smart scan, Exadata has many advantages over conventional storage. Exadata is highly optimized for fast processing of large queries. It has no bottlenecks at the controller or other levels inside the storage server and uses a state-of-the-art InfiniBand network that has much higher throughput than conventional storage networks.

Exadata Implementation Architecture Overview



Exadata Implementation Architecture Overview

The environment for a typical Exadata-based database grid is shown above.

Each Exadata cell is a self-contained appliance that houses disk storage and runs the Exadata storage server software provided by Oracle. Databases are typically deployed across multiple Exadata cells, and multiple databases can share Exadata cells. The databases and Exadata cells communicate with each other via a high-speed InfiniBand network. The collection of Exadata cell disks shared between a set of databases is referred to as an Exadata realm. The set of disks shown in the above graphic is an example of a realm. Realms ensure isolation and, therefore, protection, across a given set of databases. Mechanisms are provided to move disks and whole cells between realms in a controlled and safe manner.

Like any storage array, the Exadata storage cell is a computer with CPUs, memory, a bus, disks, Host Channel Adapters (HCA), and the other components normally found in a server. It also runs an operating system (OS), which in the case of Exadata is Linux. The Oracle-provided software resident in the Exadata cell runs under this operating system. The OS is accessible in a restricted mode to administer and manage the Exadata cell.

Introducing the Sun Oracle Exadata Storage Server

Sun Exadata Storage Server Hardware



Hardware by



Software by



- Building block of massively parallel Exadata Storage Grid
 - Up to 1.5 GB/sec raw data bandwidth
 - Up to 75,000 I/Os per second using flash
- Sun Fire™ X4275 Server
- Software pre-installed
 - Oracle Exadata Storage Server Software
 - Oracle Enterprise Linux
 - Drivers and Utilities
- Single Point of Support from Oracle
 - 3 year, 24 x 7, 4 Hr On-site response

ORACLE

1 - 7

Copyright © 2009, Oracle. All rights reserved.

Introducing the Sun Oracle Exadata Storage Server

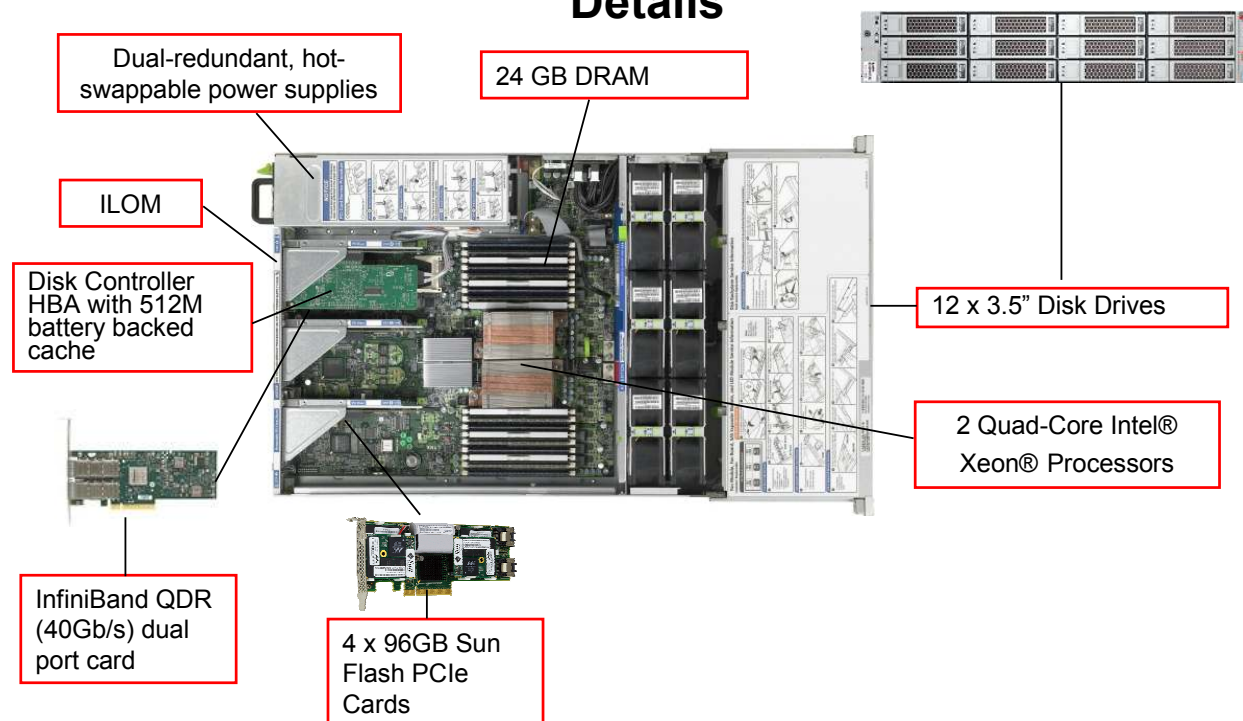
The Sun Oracle Exadata Storage Server is a storage product that is highly optimized for use with the Oracle database. Exadata delivers outstanding I/O and SQL processing performance for data warehousing and OLTP applications.

Exadata is a joint offering from Oracle and Sun and is a combination of software and hardware. Sun provides the hardware technology used in the Exadata Storage Server, it is based on the Sun Fire X4275 server. Oracle provides the storage server software to impart database intelligence to the storage, and tight integration with the Oracle Database and its features. The Exadata Storage Server comes with a 3-yr parts/ 3-yr labor/ 3-yr on-site 24x7 4-hour response warranty provided by Oracle. Oracle is the first point of contact for support of both hardware and software, and manages issues to resolution.

Each storage server is called a cell. A cell is the building block for a storage grid. You simply add more cells when you need more capacity or more I/O bandwidth. Each cell is shipped with all the hardware and software components preinstalled including the Oracle Exadata Storage software, Oracle Enterprise Linux x86_64 as the operating system, InfiniBand protocol drivers and the Sun hardware management software.

Each cell is a purely dedicated storage appliance for Oracle Database files although you can use Database File System (DBFS), a feature of the Oracle Database, to store your business files inside the database.

Sun Oracle Exadata Storage Server Hardware Details



ORACLE

1 - 8

Copyright © 2009, Oracle. All rights reserved.

Sun Exadata Storage Server Hardware Details

Here is an internal view of the Sun Exadata Storage Server hardware. This hardware is based on the Sun Fire X4275 Server. The key parts of the server are:

- 2 Quad-Core Intel Xeon E5540 Processors
- 24GB memory used internally to buffer blocks and send messages back and forth
- LSI MegaRaid SAS2 Controller with 512 MB Battery Backed Write Cache
- Dual-port 4X QDR (40Gb/s) InfiniBand card
- Dedicated port for Sun Integrated Lights Out Management (ILOM)
- Gigabit Ethernet connectivity for cell software administration
- Redundant power supplies
- 12 disks: Choice of either 12x600 GB SAS (15K RPM) disks (7.2 TB total) or 12x2TB SATA disks (24 TB total).
- 4x96 GB Sun Flash PCIe cards (384 GB total)

Note: SAS stands for Serial Attached SCSI, SATA stands for Serial ATA and ATA stands for Advanced Technology Attachment.

Sun Oracle Exadata Storage Server Specifications

	SAS Disk	SATA Disk
Exadata Smart Flash Cache ¹	384 GB	384 GB
Raw Disk Capacity ¹	7.2 TB	24 TB
Uncompressed Data Capacity ²	2 TB	7 TB
Raw Disk Throughput (MBPS)	1,500	850
Effective Throughput with Flash (MBPS)	3,600	3,600
Effective Throughput with Flash and Compression (MBPS)	36,000	36,000
Disk I/Os per Second (IOPS)	3,600	1,440
Flash I/Os per Second (IOPS)	75,000	75,000

1 - Raw capacity calculated using 1 GB = 1000 x 1000 x 1000 bytes and 1 TB = 1000 x 1000 x 1000 x 1000 bytes.

2 - User Data: Actual space for uncompressed end-user data, computed after single mirroring (ASM normal redundancy) and after allowing space for database structures such as temporary space, logs, undo space, and indexes. Actual user data capacity varies by application. User Data capacity calculated using 1 TB = 1024 * 1024 * 1024 * 1024 bytes.

ORACLE

1 - 9

Copyright © 2009, Oracle. All rights reserved.

Sun Oracle Exadata Storage Server Specifications

The Sun Oracle Exadata Storage Server is offered in two configurations: with SAS disks, or with SATA disks. The SAS option provides maximum performance while the SATA option provides maximum capacity. The table in the slide lists the key capacity and performance specifications for both cell configuration options.

Note: MBPS stands for megabytes per second, IOPS stands for I/Os per second.

Note: These metrics do not take into account compression. As a result they are quite conservative. With compressed data, you can achieve much higher rates. In all cases, actual performance will vary by application.

InfiniBand Network



- InfiniBand is the Exadata storage network:
 - Provides highest performance available - 40 Gb/sec each direction
 - Widely used in high-performance computing since 2002
- InfiniBand looks like normal Ethernet to host software:
 - All IP-based tools work transparently - tcp/ip, udp, http, ssh, ...
- InfiniBand has efficiency of a SAN:
 - Zero copy and buffer reservation abilities of storage network
- InfiniBand used for both storage and RAC interconnect:
 - Less configuration, lower cost, higher performance
- Uses high-performance ZDP InfiniBand protocol (RDS V3):
 - Zero-copy, zero-loss Datagram protocol
 - Developed by Oracle and available as Linux Open Source
 - Very low CPU overhead
 - Transfers 1 GB/sec of data with 2% CPU utilization

ORACLE

InfiniBand Network

InfiniBand is the only storage network supported by Exadata because of its performance and proven track record in high-performance computing. InfiniBand works like normal Ethernet but much faster. It has the efficiency of SAN using zero copy and buffer reservation. Zero copy means that data is transferred across the network without intermediate buffer copies in the various network layers. Buffer reservation is used so that the hardware knows exactly where to place buffers ahead of time. These are the two main characteristics that distinguish InfiniBand from normal Ethernet.

InfiniBand is also supported as a unified network fabric for Exadata storage and the Oracle RAC interconnect. This facilitates easier configuration and fewer cables and switches. You can also use it for high-performance external connectivity, such as to connect backup or ETL servers.

On top of InfiniBand, Exadata uses the ZDP (Zero Data loss UDP) protocol. ZDP is open source software that is developed by Oracle. It is like UDP but more reliable. Its full technical name is RDS (Reliable Datagram Sockets) V3. This protocol has a very low CPU overhead and allows for 1 GB/sec of data transfer with only two percent CPU utilization.

Each Exadata cell is configured with one dual-port InfiniBand card designed to be connected to two separate InfiniBand switches for high availability. Each InfiniBand link is able to carry the full data bandwidth of the entire cell which means you can loose an entire network without loosing any performance.

Sun Oracle Database Machine Full Rack

- 8 Sun Fire™ X4170 Oracle Database servers
- 14 Exadata Storage Servers (All SAS or all SATA)
- 3 Sun Datacenter InfiniBand Switch 36
 - 36-port Managed QDR (40Gb/s) switch
- 1 Cisco Ethernet switch (48-port)
- Keyboard, Video, Mouse (KVM) hardware
- 2 Power Distribution Units (PDUs)
- Single Point of Support from Oracle
 - 3 year, 24 x 7, 4 Hr On-site response



ORACLE

Sun Oracle Database Machine Full Rack

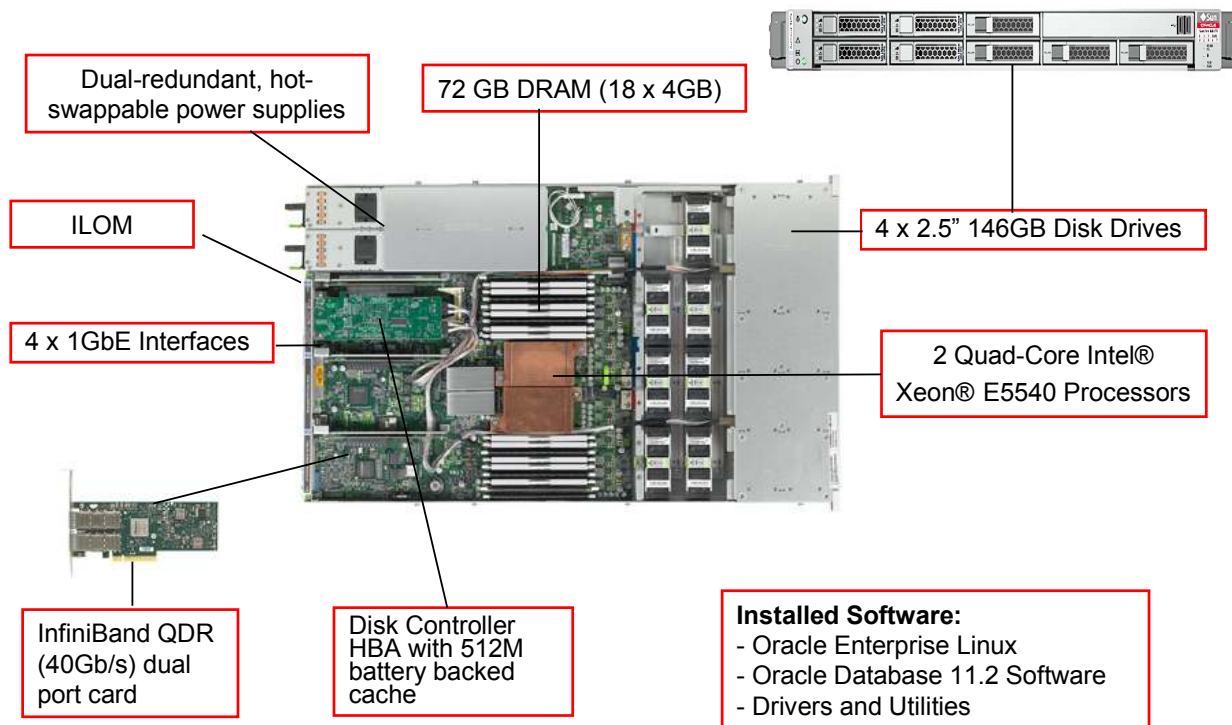
In addition to offering the Exadata V2 storage server as a standalone storage product, Oracle offers a complete Exadata-based database solution called the Sun Oracle Database Machine.

The Sun Oracle Database Machine provides an optimal solution for all database workloads, ranging from scan-intensive data warehouse applications to highly concurrent online transaction processing (OLTP) applications. The Sun Oracle Database Machine delivers extreme performance in a highly-available, highly-secure environment. Using the unique clustering and workload management capabilities of the Oracle Database, the Sun Oracle Database Machine is well-suited for consolidating multiple databases into a single grid. Delivered as a complete pre-optimized, and pre-configured package of software, servers, and storage, the Sun Oracle Database Machine is simple and fast to implement.

The contents of a Sun Oracle Database Machine Full Rack configuration is listed on the slide.

Note: By itself, the Sun Oracle Database Machine does not include any Oracle software licenses. Appropriate licensing of the Oracle Database and Oracle Exadata Storage Server software is required. In addition, the Sun Oracle Database Machine is highly complementary with clustering and a parallel operations so Oracle recommends licensing Oracle Real Application Clusters and Oracle Partitioning in this environment.

Sun Database Server Hardware Details

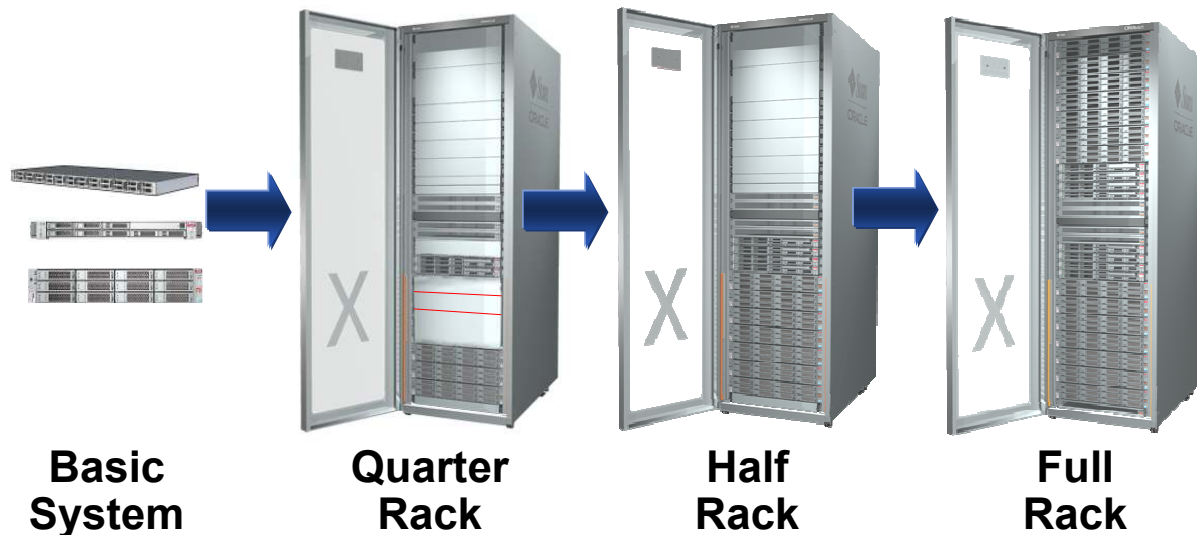


Sun Database Server Hardware Details

The Oracle Database Machine uses standardized database server hardware based on the Sun Fire X4170 Server. The slide shows a view of the database server hardware. The key components of each server are:

- 2 Quad-Core Intel Xeon E5540 processors
- 72 GB RAM
- 4 x 146 GB 10K RPM SAS disks
- Disk controller HBA with 512 MB battery-backed write cache
- Dual-port QDR (40 Gb/s) InfiniBand Host Channel Adapter (HCA)
- 4 embedded Gigabit Ethernet ports
- 1 Ethernet port for Sun Integrated Lights Out Manager (ILOM) for remote management
- Oracle Enterprise Linux

Start Small and Grow



Start Small and Grow

The Sun Oracle Database Machine is offered in four configurations designed to suit a wide range of requirements.

A basic system is one database server, one storage server, and one InfiniBand switch. It is the smallest possible configuration. A basic system is not supplied with a rack or a KVM and is not pre-cabled. A basic system is not fault tolerant.

Beyond a basic system, there are three rack-mounted offerings:

- A quarter rack has two database servers, and three Exadata cells.
- A half rack has four database servers, and seven Exadata cells.
- A full rack has eight database servers, and 14 Exadata cells.

All of the rack-mounted Sun Oracle Database Machine products come with 2 PDU's and 2 InfiniBand switches to provide fault-tolerance. In addition, they are all pre-cabled and supplied with a KVM and Ethernet switch to complete the configuration.

Sun Oracle Database Machine Capacity

	Quarter Rack	Half Rack	Full Rack
Raw Disk¹	SAS	21 TB	50 TB
	SATA	72 TB	168 TB
Raw Flash¹	1.1 TB	2.6 TB	5.3 TB
User Data²	SAS	6 TB	14 TB
	SATA	21 TB	50 TB

1 - Raw capacity calculated using 1 GB = 1000 x 1000 x 1000 bytes and 1 TB = 1000 x 1000 x 1000 x 1000 bytes.

2 - User Data: Actual space for uncompressed end-user data, computed after single mirroring (ASM normal redundancy) and after allowing space for database structures such as temporary space, logs, undo space, and indexes. Actual user data capacity varies by application. User Data capacity calculated using 1 TB = 1024 * 1024 * 1024 * 1024 bytes.

ORACLE

Sun Oracle Database Machine Capacity

The above table gives you some key product capacity information depending on the Sun Oracle Database Machine configuration you choose.

Sun Oracle Database Machine Performance

		Quarter Rack	Half Rack	Full Rack
Raw Disk Throughput ¹	SAS	4.5 GB/s	10.5 GB/s	21 GB/s
	SATA	2.5 GB/s	6 GB/s	12 GB/s
Raw Flash Throughput ¹		11 GB/s	25 GB/s	50 GB/s
Flash IOPS ²		225,000	500,000	1,000,000

1 – Bandwidth is peak physical disk scan bandwidth, assuming no compression. .

2 – IOs per second (IOPS) – Based on IO requests of size 8K

ORACLE

Sun Oracle Database Machine Performance

The above table gives you some key performance information depending on the Sun Oracle Database Machine configuration you choose.

Note: These metrics do not take into account compression. As a result they are quite conservative. With compressed data, you can achieve much higher rates. In all cases, actual performance will vary by application.

Scale Performance and Capacity



Scalable

- Scale to 8 racks by just adding cables
- Scale beyond 8 racks by adding more Infiniband switches
- Scale to hundreds of storage servers to support multi-petabyte databases

Redundant and Fault Tolerant

- Failure of any component is tolerated
- Data is mirrored across storage servers

ORACLE

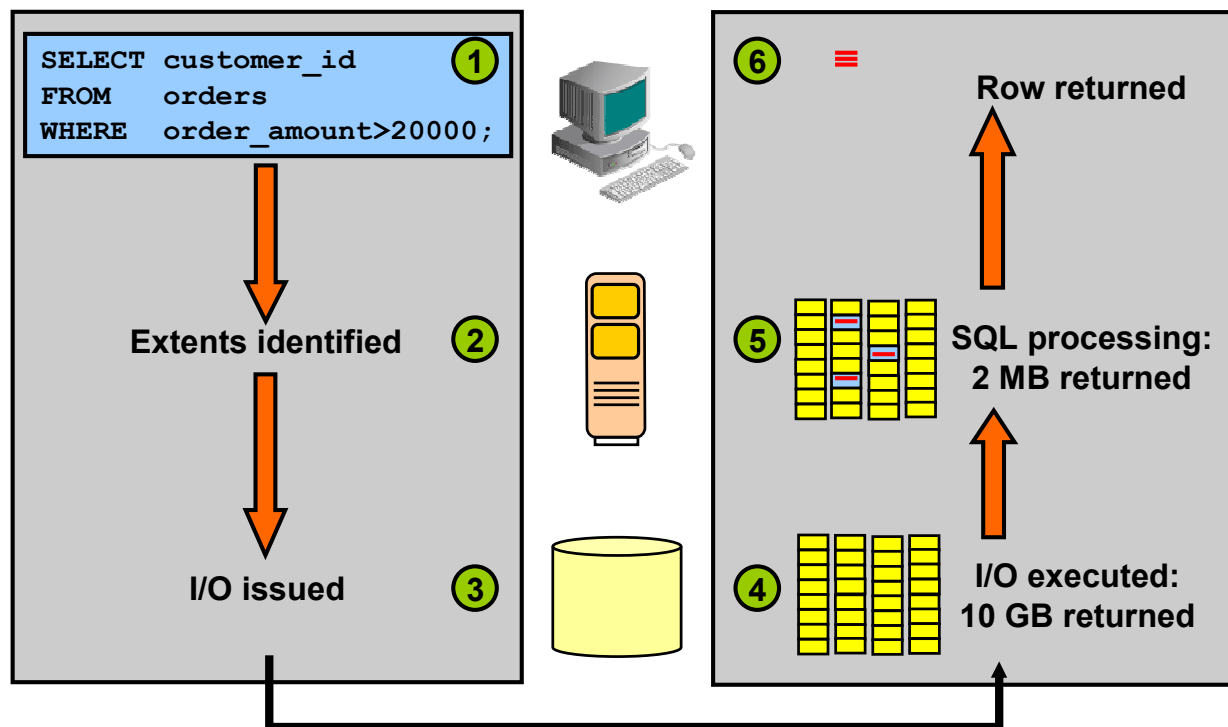
Scale Performance and Capacity

If you need to scale beyond the performance or capacity of an Oracle Database Machine Full Rack configuration, you can scale up to eight racks by simply adding some cabling between them. There are enough free ports in the InfiniBand switches and enough bandwidth for eight Database Machines running at full performance and with full redundancy.

If you want to extend beyond eight racks, you need to add InfiniBand switches.

Exadata implementations can scale to hundreds of storage servers in order to support multi-petabyte databases. Exadata is designed to tolerate the loss of any component, with all the data mirrored across storage servers.

Classic Database I/O and SQL Processing Model



1 - 17

Copyright © 2009, Oracle. All rights reserved.

ORACLE

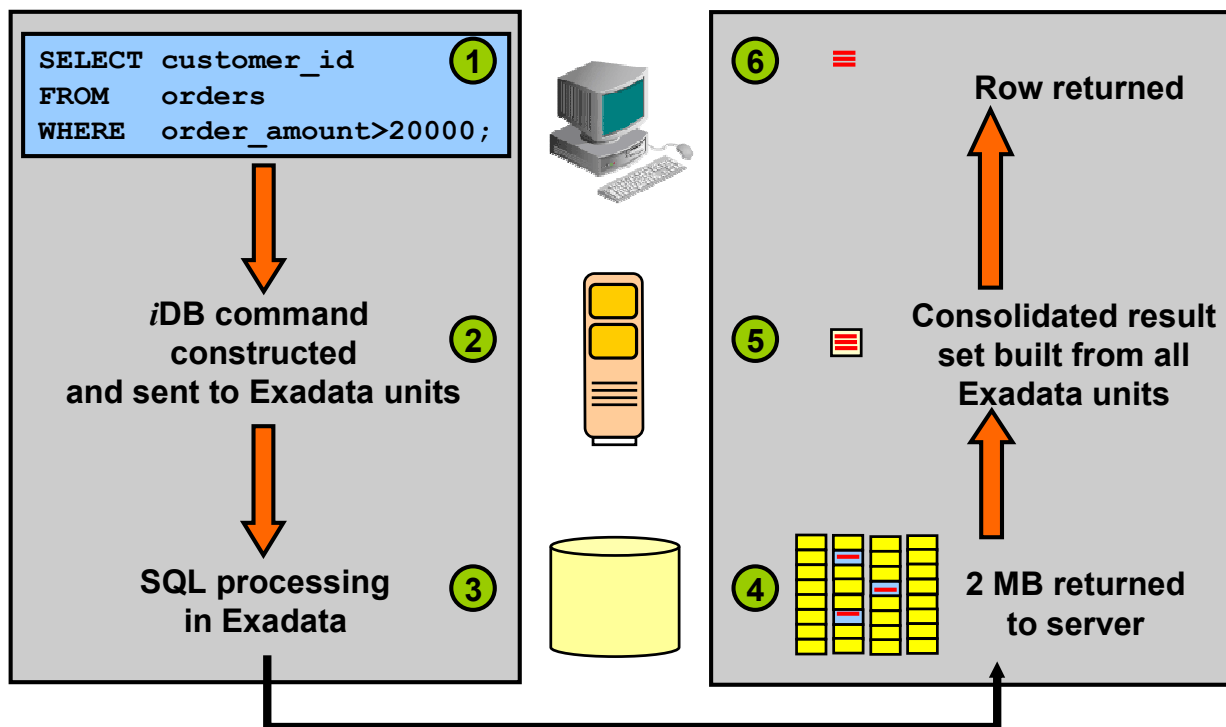
Classic Database I/O and SQL Processing Model

With traditional storage all the database intelligence resides in the software on the database server. To illustrate how SQL processing is performed in this architecture, an example of a table scan is shown in the above graphic.

1. The client issues a `SELECT` statement with a predicate to filter a table and return only the rows of interest to the user.
2. The database kernel maps this request to the file and extents containing the table.
3. The database kernel issues the I/Os to read all the table blocks.
4. All the blocks for the table being queried are read into memory.
5. SQL processing is conducted against the data blocks searching for the rows that satisfy the predicate.
6. The rows are returned to the client.

As is often the case with the large queries, the predicate filters out most of the rows in the table. Yet all the blocks from the table need to be read, transferred across the storage network, and copied into memory. Many more rows are read into memory than required to complete the requested SQL operation. This generates a large amount of unproductive I/O, which wastefully consumes resources and impacts application throughput and response time.

Exadata Smart Scan Model



ORACLE

1 - 18

Copyright © 2009, Oracle. All rights reserved.

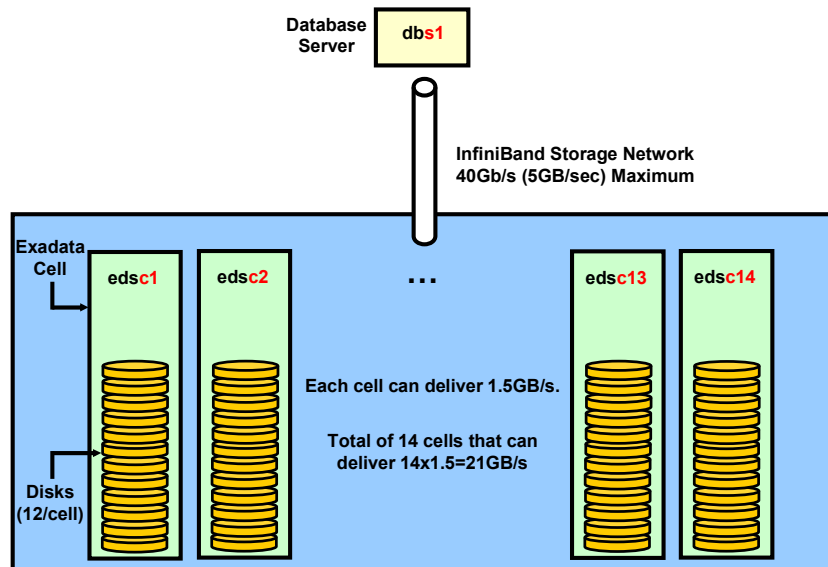
Exadata Smart Scan Model

Using Exadata storage, database operations are handled differently. Queries that perform table scans can be processed within Exadata and return only the required subset of data to the database server. Row filtering, column filtering, some join processing, and other functions can be performed within Exadata. Exadata uses a special direct-read mechanism for smart scan processing. The above graphic illustrates how a table scan operates with Exadata storage:

1. The client issues a `SELECT` statement to return some rows of interest.
2. The database kernel determines that Exadata storage is available and constructs an `iDB` command representing the SQL command and sends it to the Exadata storage. `iDB` is a unique Oracle data transfer protocol that is used for Exadata storage communications.
3. The `CELLSRV` component of the Exadata software scans the data blocks to extract the relevant rows and columns which satisfy the SQL command.
4. `CELLSRV` returns to the database instance an `iDB` message containing the requested rows and columns of data. These results are not block images, so they are not stored in the buffer cache.
5. The database kernel consolidates the result sets from across all the Exadata cells. This is similar to how the results from a parallel query operation are consolidated.
6. The rows are returned to the client.

Moving SQL processing off the database server frees server CPU cycles and eliminates a massive amount of unproductive I/O transfers. These resources are free to better service other requests. Queries run faster, and more of them can be processed.

Exadata Smart Scan Scale-Out Example



ORACLE

1 - 19

Copyright © 2009, Oracle. All rights reserved.

Exadata Smart Scan Scale-Out Example

The example on the next three slides illustrates the power of smart scan in a quantifiable manner using a typical implementation in which multiple Exadata storage servers scale-out to share a workload.

In this scenario you have 14 Exadata cells. Assuming that each Exadata cell can deliver 1.5 gigabytes per second, the potential delivery power of all the Exadata cells is 21 gigabytes per second.

The database server, depicted in the upper portion of the slide, is connected to the InfiniBand storage network which can deliver a maximum of 40 gigabits per second (or 5 gigabytes per second).

Exadata Smart Scan Scale-Out Example

**Without
smart scan**

```
select /*+ full(lineitem) */ count(*)  
from lineitem  
where l_orderkey < 0;
```

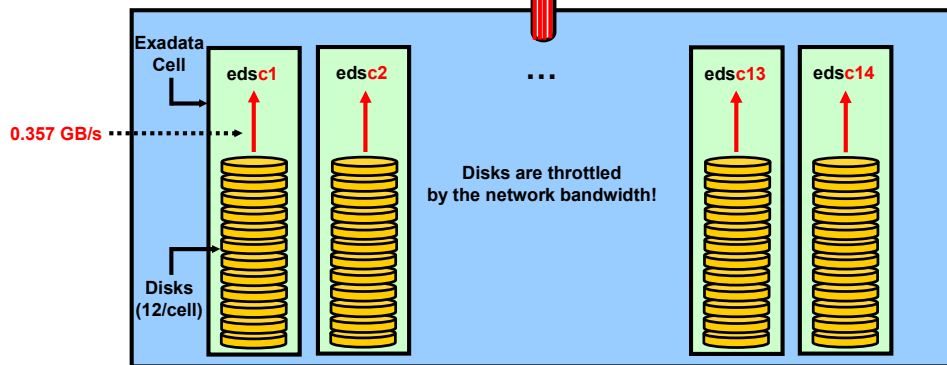
Database
Server

db_{s1}

Database asks to retrieve all
blocks by doing a full table scan,
and then filters matching rows.

If table is 4800 GB in size, the
complete scan would take
approximately **16 minutes**.

If table is evenly distributed
across all disks, each cell
cannot send more than $40 / 14 =$
 $2.85 \text{ Gb/s} = \mathbf{0.357 \text{ GB/s}}$
to the database instance.



ORACLE

1 - 20

Copyright © 2009, Oracle. All rights reserved.

Exadata Smart Scan Scale-Out Example (continued)

Now assume a 4800 gigabyte table is evenly spread across the 14 Exadata cells and a query is executed which requires a full table scan. As is commonly the case, we will assume that the query returns a small set of result records.

Without smart scan capabilities, each Exadata Cell behaves as a traditional storage server by delivering database blocks to the client database.

Because the storage network is bandwidth-limited to 40 gigabits per second, it is not possible for the Exadata cells to deliver all their power. In this case, each cell cannot deliver more than 0.357 gigabytes per second to the database and it would take approximately 16 minutes to scan the whole table.

Exadata Smart Scan Scale-Out Example

**With
smart scan**

```
select /*+ full(lineitem) */ count(*)  
from lineitem  
where l_orderkey < 0;
```

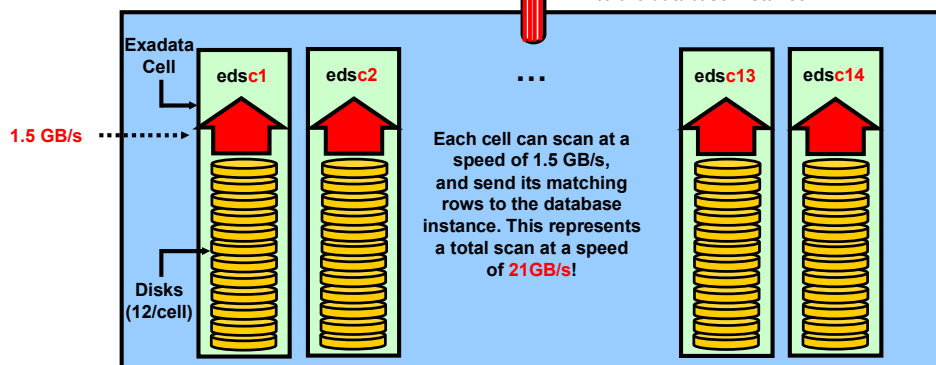
Database
Server

db^s1

Database asks Exadata cells
to send back all matching rows.

If table is 4800 GB in size, the complete
table scan will complete in less than
four minutes!

If table is evenly distributed
across all disks, each cell
cannot send more than $40 / 14 =$
 $2.85 \text{ GB/s} = \mathbf{0.357 \text{ GB/s}}$
to the database instance.



ORACLE

Exadata Smart Scan Scale-Out Example (continued)

Now consider if smart scan is enabled for the same query. The same storage network bandwidth limit applies. However, this time the database is no longer asking for blocks, but for matching rows. So, each Exadata Cell can process its part of the table at full speed, that is 1.5 gigabytes per second. In this case the entire table scan would be completed in less than four minutes.

Exadata Smart Storage Capabilities

- Predicate filtering:
 - Only the rows requested are returned to the database server rather than all rows in a table.
- Column filtering:
 - Only the columns requested are returned to the database server rather than all columns in a table.
- Join processing:
 - Simple star join processing is performed within Exadata.

ORACLE

1 - 22

Copyright © 2009, Oracle. All rights reserved.

Exadata Smart Storage Capabilities

The following database functions are integrated within the Exadata storage layer:

- Exadata enables predicate filtering for table scans. Rather than returning all the rows for the database to evaluate, Exadata returns only the rows that match the filter condition. For example, when the following SQL statement is issued, only rows where the employees' hire date is after the specified date are sent from Exadata to the database kernel:
`SELECT * FROM employee_table WHERE hire_date > '1-Jan-2003';`
The conditional operators that are supported include =, !=, <, >, <=, >=, IS [NOT] NULL, LIKE, [NOT] BETWEEN, [NOT] IN, EXISTS, IS OF type, NOT, AND, OR.
- Exadata provides column filtering, also called column projection, for table scans. Only the requested columns are returned to the database server rather than all columns in a table. For example, when the following SQL statement is issued, only the employee_name and employee_number columns are returned from Exadata to the database kernel:
`SELECT employee_name, employee_number FROM employee_table;`
For tables with many columns, or columns containing LOBs, the I/O bandwidth saved can be very large. When used together, the combination of predicate and column filtering dramatically improves performance and reduces I/O bandwidth consumption.
- Exadata performs join processing for star schemas (between large tables and small lookup tables). This is implemented using Bloom Filters, which is a very efficient probabilistic method to determine whether an element is a member of a set.

Exadata Smart Storage Capabilities

- Scans on encrypted data 
- Scoring for Data Mining: 
 - All data mining scoring functions are offloaded
 - Up to 10x performance gains

ORACLE

1 - 23

Copyright © 2009, Oracle. All rights reserved.

Exadata Smart Storage Capabilities (continued)

- Oracle Exadata Storage Server performs smart scans on encrypted tablespaces and encrypted columns. For encrypted tablespaces, Exadata can decrypt blocks and return the decrypted blocks to the Oracle Database, or it can perform column projection which returns rows and columns. Significant CPU savings can be made within the database server by offloading the CPU-intensive decryption task to Exadata cells.
- Oracle Exadata Storage Server can perform scoring functions for data mining models. All data mining scoring functions, such as `PREDICTION_PROBABILITY`, are offloaded to Exadata cells for processing. This accelerates warehouse analysis while it reduces database server CPU consumption and the I/O load between the database server and Exadata storage.

Exadata Smart Storage Capabilities

- Backups:
 - I/O for incremental backups is much more efficient because only changed blocks are returned to the database server.
- Create/extend tablespace:
 - Exadata formats tablespace extents.

ORACLE

1 - 24

Copyright © 2009, Oracle. All rights reserved.

Exadata Smart Storage Capabilities (continued)

- The speed and efficiency of incremental database backups are significantly enhanced with Exadata. The granularity of change tracking in the database is much finer with Exadata. With Exadata, changes are tracked at the individual Oracle block level rather than at the level of a large group of blocks. This results in less I/O bandwidth being consumed for backups and faster running backups.
- With Exadata, the create/extend tablespace operation is also executed much more efficiently. Instead of formatting blocks in database server memory and writing them to storage, a single *iDB* command is sent to Exadata instructing it to format the blocks. Host memory usage is reduced and I/O associated with the creation and formatting of the tablespace blocks is eliminated with Exadata.

Exadata Hybrid Columnar Compression

Query Compression

Optimized for Speed

- 10x average storage savings
- 10x scan I/O reduction
- Optimized for query performance

**Smaller Warehouse
Faster Performance**

Archive Compression

Optimized for Space

- 15x average storage savings
 - Up to 50x on some data
- Some access overhead
- For cold or historical data

**Reclaim Disks
Keep Data Online**

Can mix compression types by partition for ILM

ORACLE

Exadata Hybrid Columnar Compression

In addition to the BASIC and OLTP compression capabilities of Oracle Database 11g, Exadata V2 introduces Exadata Hybrid Columnar Compression.

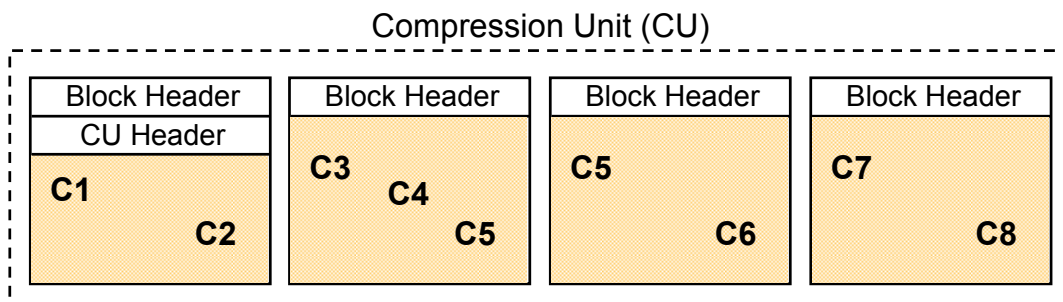
Exadata Hybrid Columnar Compression offers higher compression ratios for direct path loaded data. This compression capability is recommended for data that is not updated frequently. You can specify Exadata Hybrid Columnar Compression at the table, partition, and tablespace level. You can also choose between two types of Exadata Hybrid Columnar Compression, to achieve the proper trade-off between disk usage and CPU overhead, depending on your requirements:

- **QUERY** compression: This type of compression is optimized for query performance, and is intended for data warehouse applications.
- **ARCHIVE** compression: This type of compression is optimized for maximum compression ratios, and is intended for data that does not change.

You can use Exadata Hybrid Columnar Compression on complete tables or in combination with BASIC and OLTP compression by using partitioning.

Note: A compression advisor, provided by the DBMS_COMPRESSION package, helps you determine the expected compression ratio for a particular table with a particular compression method.

Exadata Hybrid Columnar Compression Architecture Overview



- Logical structure spanning multiple database blocks
- Data organized by column during data load
- Each column compressed separately
- Each row is self-contained within a compression unit
- Smart scans are supported

ORACLE

Exadata Hybrid Columnar Compression Architecture Overview

Exadata Hybrid Columnar Compression is a new method for organizing data in database blocks. Tables are organized into sets of rows called compression units (CU). Within a compression unit, data is organized by column and then compressed. The column organization of data brings similar values close together, enhancing compression ratios. Each row is self-contained within a compression unit.

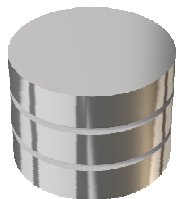
In addition to providing excellent compression, Exadata Hybrid Columnar Compression works in conjunction with smart scan so that column projection and row filtering can be executed along with decompression at the storage level to save CPU cycles on your database servers.

Note: Although the diagram in the slide shows a compression unit containing four data blocks, it should not be assumed that a compression unit always contains four blocks. The size of a compression unit is determined by the Oracle Database based on various factors in order to deliver the most effective compression result.

Exadata Smart Flash Cache

- High performance cache for frequently accessed objects
- Excellent for absorbing repeated random reads
- Allows optimization by application table

300 I/Os per Sec



Tens of Thousands of I/Os per Second



ORACLE

1 - 27

Copyright © 2009, Oracle. All rights reserved.

Exadata Smart Flash Cache

For many years a constraining factor for storage performance has been the number of random I/Os per second (IOPS) that a disk could deliver. To compensate for the fact that even a high performance disk can only deliver a few hundred IOPS, large storage arrays with hundreds of disks are constructed to deliver in excess of 60,000 IOPS.

Sun Oracle Exadata Storage Server provides Exadata Smart Flash Cache, a caching mechanism for frequently-accessed data on each Exadata cell. It is a write-through cache which is useful for absorbing repeated random reads, and very beneficial to OLTP. Using Exadata Smart Flash Cache, a single Exadata cell can support up to 75,000 IOPS, two cells can support up to 150,000 IOPS, and so on.

Exadata Smart Flash Cache focuses on caching frequently accessed data and index blocks, along with performance critical information such as control files and file headers. In addition, DBAs can influence caching priorities using the `CELL_FLASH_CACHE` storage attribute for specific database objects.

Exadata Smart Flash Cache

High performance cache that understands different types of database I/O:

- Frequently accessed data and index blocks are cached
 - Control file reads and writes are cached
 - File header reads and writes are cached
 - DBA can influence caching priorities
- 
- I/Os to mirror copies are not cached
 - Backup related I/O is not cached
 - Data Pump I/O is not cached
 - Data file formatting is not cached
 - Table scans don't monopolize the cache
- 

ORACLE

Exadata Smart Flash Cache (continued)

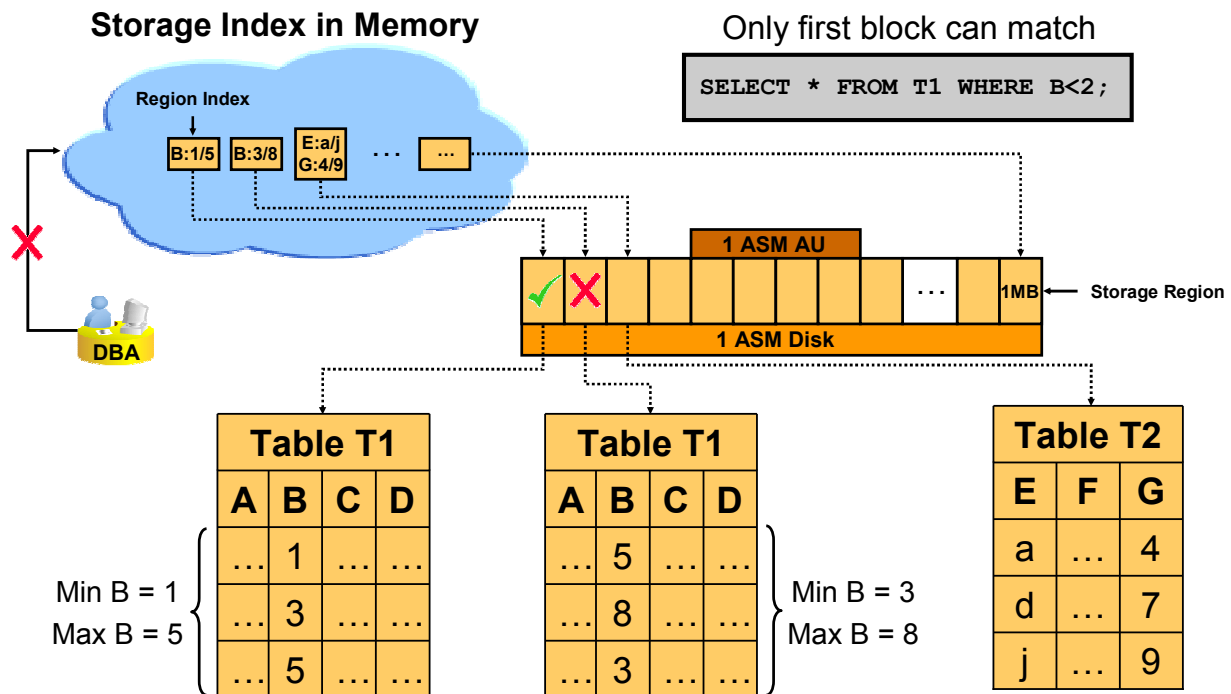
In recent times, vast and expensive storage arrays have introduced equally expensive non-volatile memory caches to improve performance. However, these caches know nothing about the applications using them so their efficiency is limited when compared to their cost.

With Exadata, each database I/O is flagged with meta data indicating the I/O type. Exadata Smart Flash Cache uses this information to make intelligent decisions about how to use the cache. This cooperation ensures the efficient use of Exadata Smart Flash Cache.

For example, with ASM mirroring turned on, multiple copies of each data block must be written to disk to deliver the desired level of data protection. However, there is usually no need to cache the secondary copies of a block because ASM will read the primary copy if it is available. A traditional storage array would not know about this characteristic leading to caching inefficiencies.

Similarly, with traditional storage arrays backups and exports will typically cause all the data to be loaded into the cache even though the operation will not read the data repeatedly. On the other hand, Exadata knows that there is no need to fill the cache with backup and export data. The same is true for data file formatting operations. Finally, Exadata does not flood the cache with data from full table scans, as is the case with most storage arrays.

Exadata Storage Index



Exadata Storage Index

A storage index further reduces the amount of I/O that is required for certain data scans. The storage index keeps track of minimum and maximum values of columns for tables stored on that cell. This information helps Exadata to avoid some I/Os during a large table or index scan.

The storage space inside each cell disk is logically divided into 1MB chunks called storage regions. The boundaries of ASM allocation units (ASM AUs) are aligned with the boundaries of storage regions. For each of these storage regions, data distribution statistics are held in a memory structure called a region index. A storage index is a collection of these region indexes.

The storage statistics represent the data distribution (minimum and maximum values) of columns that are considered well clustered by the Exadata software. Exadata software has heuristics to transparently determine what columns are clustered enough to be included in the storage index.

A storage index works best when the following conditions are true:

- the data has to be roughly ordered by some column
- the query needs a predicate on the storage index column checking for =, <, > or some combination of these.

It is important to note that storage indexes work transparently with no user input. There is no need to create, drop, or tune a storage index. The only way to influence a storage index is to load your tables using pre-sorted data.

Exadata Storage Index (continued)

Also, because the storage index is kept in memory, it disappears when the cell is rebooted. The first queries that run after a cell is rebooted automatically cause the storage index to be rebuilt.

A storage index works for data types whose binary encoding is such that byte-wise binary lexical comparison of two values of that data type is sufficient to determine the ordering of those two values. This includes data types like `NUMBER`, `DATE`, and `VARCHAR2`. However, NLS data types are one example of data types which are not included for storage index filtering.

Storage Index with Partitions Example

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 is highly correlated with ORDER_DATE
- Storage index provides partition pruning like performance for queries on SHIP_DATE
 - Takes advantage of ordering created by partitioning

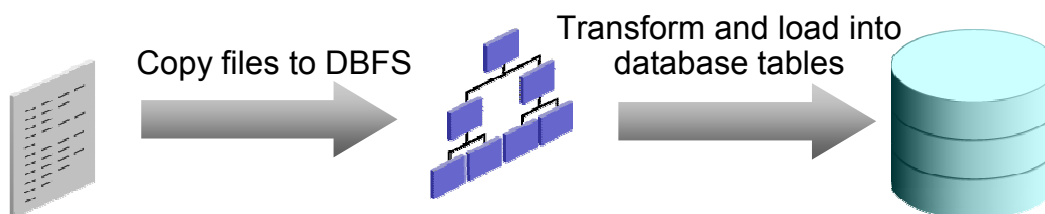
ORACLE

Storage Index with Partitions Example

The example on the slide contains correlated columns. ORDER_DATE is highly correlated with SHIP_DATE. The dates are generally correlated because usually a ship date is close to an order date. If your table is partitioned by ORDER_DATE, and you execute a query using ORDER_DATE as a filter, then partition pruning is used to read only the relevant partitions. However, if you do a query using SHIP_DATE in the WHERE clause, then the system does a full table scan. In this case, the storage index is used to skip all the blocks that do not correspond to your query. This filtering takes place at the storage level. In this case, a natural ordering is automatically created because of the partitioning schema being used and because of the natural correlation that exists between the two columns.

Database File System

- Database File System (DBFS) enables the database to be used as a file system
- Files are stored as SecureFile LOBs inside database tables that are stored in Exadata
 - Protected like any Oracle data – ASM mirroring, Data Guard, Flashback, etc.
- Database Machine comes with DBFS shared Linux file system
 - Shared storage for ETL staging, scripts, reports and other application files
 - 5 to 7 GB/sec file system I/O throughput



ORACLE

Database File System

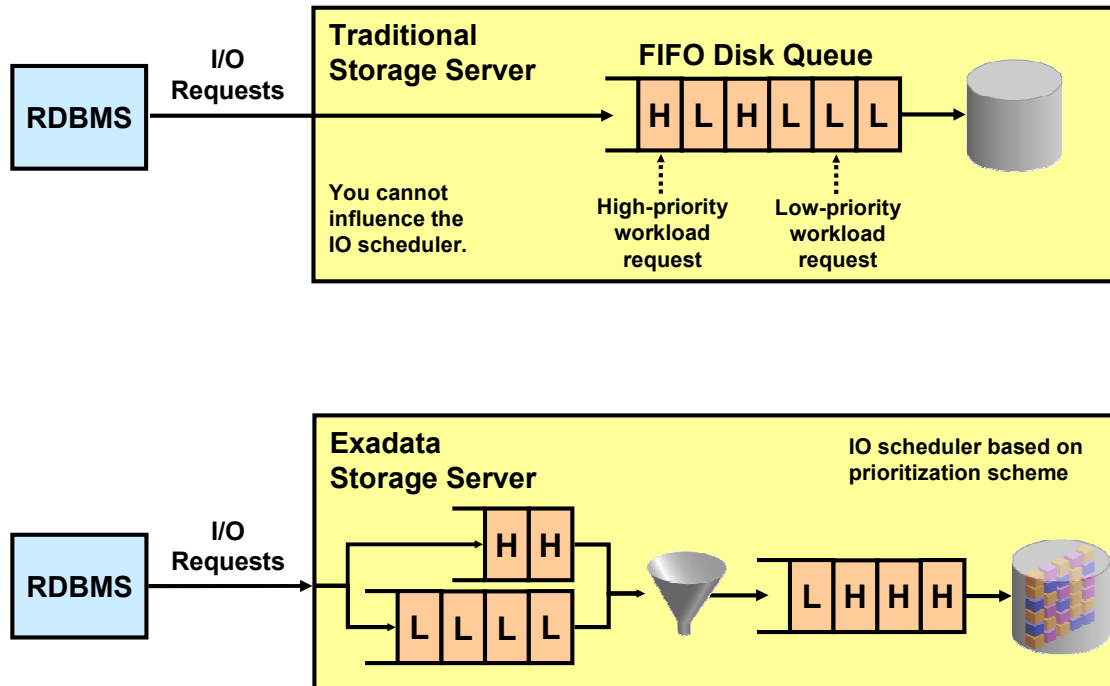
The Oracle Database File System (DBFS) enables the database to be used as a POSIX-compatible file system on Linux. DBFS is an Oracle Database capability that provides Exadata users with an high performance mechanism to load data into the Oracle Database. DBFS can be used to stage your ETL files for example.

Inside DBFS files are stored as SecureFile LOBs. A set of PL/SQL procedures implement the file system access primitives like open, create etc. The `dbfs_client` utility enables mounting of a DBFS file system as a mount point on Linux. It provides the mapping from file system operations to database operations. The `dbfs_client` utility runs completely in user space and interacts with the kernel through the FUSE library infrastructure.

The Oracle Database Machine comes with DBFS preconfigured. Oracle Database File System is available on the database nodes in the `/opt/oracle/dbfs` directory. For Exadata implementations outside of the Oracle Database Machine, an administrator must configure DBFS.

Note: ASM Cluster File System (ACFS) is not supported on an Exadata storage.

I/O Resource Management

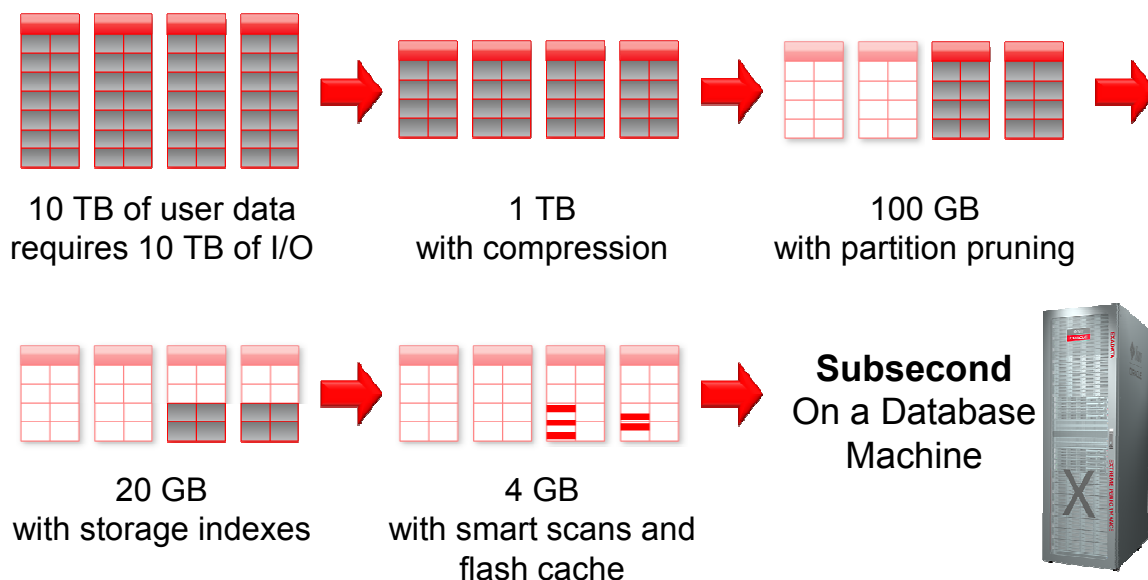


I/O Resource Management

With traditional shared storage, balancing the work of multiple databases sharing the storage subsystem is inherently difficult. This issue is illustrated by the top graphic in the slide, which shows how traditional storage servers handle I/O requests. In essence, they queue I/O requests in a first-in/first-out (FIFO) order which makes no distinction between high-priority and low-priority requests. Exadata allows for allocation of I/O resources based on user-specified priorities and policies. This is illustrated in the graphic at the bottom of the slide where the Exadata I/O scheduler executes I/O requests based on a prioritization scheme. It does that by internally queuing I/O requests to prevent a low-priority but intensive workload from flooding the disks.

I/O resource management is covered in more detail in lesson five of this course.

Benefits Multiply



Data is 10x Smaller, Scans are 2000x faster

ORACLE

Benefits Multiply

This is an example showing you how the main Exadata features that were introduced in this lesson can work together to multiply the benefits in terms of I/O reduction transmitted across the network. Assume you have a 10 terabyte table and somebody runs a query that causes a full table scan but is only interested in a very small result set. Traditionally, the system would have to scan the 10 terabytes of data.

However, using compression, Exadata Hybrid Columnar Compression could reduce that amount of data to just one terabyte. So at this point the system would only need to scan one terabyte of data.

If the table is partitioned, the optimizer could use partition pruning and could eliminate a large proportion of the data. Assuming the query can be satisfied using 10 percent of the available partitions, the optimizer could potentially skip 90 percent of the data. This would leave only 100 gigabytes of data to scan.

Using storage indexes, Exadata might further reduce the amount of data to scan to 20 gigabytes.

Finally, because of smart scan, the amount of data returned to the database could be reduced to 4 gigabytes, which could potentially be fetched out of flash memory.

That means a scan of a 10 terabyte table could be processed in less than a second on a Database Machine.

Exadata Key Benefits for Data Warehousing

- Exadata uses more connections
 - Modular storage cell building blocks organized into massively parallel grid
- Exadata has bigger network pipes
 - InfiniBand network transfers data faster than Fibre Channel
- Exadata transports less data between storage and database
 - Query processing is moved into storage to dramatically reduce data sent to servers while unloading server CPUs
- Exadata Hybrid Columnar Compression 
- In-memory parallel query 

10X to 100X speedup for data warehousing

ORACLE

1 - 35

Copyright © 2009, Oracle. All rights reserved.

Exadata Key Benefits for Data Warehousing

One of the key benefits of Exadata storage is extremely enhanced performance for data warehousing applications. By replacing your existing storage with Exadata, it is possible to get up to 100 times speedup for your data warehousing applications. The larger the data warehouse, the greater the speedup from using Oracle Exadata storage.

Exadata addresses three key dimensions of database I/O that can hamper data warehouse performance.

- Exadata is based on a massively parallel architecture, which provides more connections to deliver more data faster between the database servers and the storage servers.
- Exadata is built using wide network pipes that provide extremely high bandwidth between the database servers and the storage servers. Exadata uses InfiniBand as the storage network which provides a throughput of 40 Gb/sec with very low latency. This is many times the bandwidth provided by traditional SAN HBAs.
- Exadata is database-aware and can transport just the data required to satisfy SQL requests resulting in less data being sent between the database servers and the storage servers.

Basically, Exadata reduces the volume of data transported and moves data faster compared with other storage solutions.

Exadata Key Benefits for Data Warehousing (Continued)

In addition to these benefits, Exadata V2 introduces additional capabilities which can further enhance data warehouse performance.

Exadata V2 introduces Exadata Hybrid Columnar Compression. This feature provides very high levels of data compression implemented inside the Exadata storage server. Exadata Hybrid Columnar Compression benefits large scale scans, commonly used in data warehousing, by efficiently scanning vast volumes of data using a fraction of I/Os. Compression ratios of 10 to 1 are common meaning that a 10 TB table can be scanned using 1 TB of disk I/O.

Exadata's tight integration with the Oracle Database results in an intelligent platform for data warehousing. The complete solution utilizes a range of technologies to deliver the best result, not just relying on one approach to the problem. An example of this is the new in-memory parallel query feature of Oracle Database 11g Release 2.

Normally, a smart scan would be used to execute portions of a query inside Exadata storage and return the minimum amount of data to the database server. In some cases, however, it may be more efficient to read all the required data into the memory on the database servers and process the query that way.

In-memory parallel execution harnesses the aggregated memory in a system to enhance query performance by minimizing or even completely eliminating the physical I/O needed for a parallel operation. Oracle automatically decides if an object being accessed using parallel execution benefits from being cached in the database buffer cache. The decision to cache an object is based on a well defined set of heuristics including size of the object and the frequency that it is accessed. In an Oracle RAC environment, Oracle maps fragments of the object into each of the buffer caches on the active instances. By creating this mapping, Oracle knows which buffer cache to access to find a specific part or partition of an object to answer a given SQL query. Using this information, Oracle Database will prevent multiple instances from reading the same information from disk over and over again, thus maximizing the amount of memory that can be used to cache the objects.

In-memory parallel query harnesses the aggregated memory in a system for parallel operations, enabling it to scale out with the available memory for data caching as the number of nodes in a cluster increases. This new functionality optimizes large parallel operations by minimizing or even completely eliminating the physical I/O needed because the parallel operation can now be satisfied in memory.

In-memory parallel query nicely complements Exadata storage. Using this combination some queries can be efficiently executed with little or no I/O by pinning tables in the database buffer cache while others can harness the power of smart scan inside Exadata storage.

Exadata Key Benefits for OLTP

- Exadata uses more connections
 - Modular storage cell building blocks organized into massively parallel grid
- Exadata has bigger network pipes
 - InfiniBand network transfers data faster than Fibre Channel.
- Exadata Smart Flash Cache
 - High performance cache for frequently accessed objects
 - Excellent for absorbing repeated random reads
 - Allows optimization by application table



300 I/Os per Sec



Tens of Thousands of I/Os per Second



Exadata Key Benefits for OLTP

Some of the fundamental architectural characteristics that are beneficial for data warehousing are equally relevant and beneficial for on-line transaction processing (OLTP). The high-performance, low-latency, InfiniBand network used in conjunction with the massively parallel grid architecture of Oracle Exadata is ideal for supporting many thousands of simultaneous users.

In addition, the introduction of Exadata Smart Flash Cache is of particular benefit to OLTP performance. Exadata Smart Flash Cache allows each Exadata cell to deliver up to 75,000 IOPS. In addition, Oracle Database and Exadata Smart Flash Cache work closely with each other. This cooperation optimizes the usage of Exadata Smart Flash Cache so that only the most frequently-accessed and performance sensitive data is cached. Users have additional control over which database objects should be cached more aggressively than others, and which ones should not be cached at all.

Quiz

Oracle Exadata Storage Server and Oracle Database Machine are two different names that designate the same thing.

1. TRUE
2. FALSE

ORACLE®

1 - 38

Copyright © 2009, Oracle. All rights reserved.

Answers: 2

Quiz

What are three benefits of Oracle Exadata Storage Server compared to traditional storage servers?

1. Disk sizes
2. Smart storage capabilities
3. Network bandwidth
4. RAM capacity
5. Database I/O resource management

ORACLE

1 - 39

Copyright © 2009, Oracle. All rights reserved.

Answers: 2, 3, 5

Summary

In this lesson, you should have learned how to:

- Contrast the Exadata storage architecture with traditional shared storage offerings
- Describe the hardware components of the Sun Oracle Exadata Storage Server and Sun Oracle Database Machine
- Detail the capabilities of the Exadata Storage Server
- Describe the main advantages of using Exadata compared to traditional storage servers

ORACLE