Chris Craft's Oracle Blog

The views expressed on this blog are my own and do not necessarily reflect the views of Oracle Corporation.

What Makes Exadata Faster?

February 9, 2018February 21, 2018 chriscraftus Exadata

In this blog post, we will explore what makes Exadata the fastest database platform in the universe. We see Exadata being compared to servers like IBM's Power Systems. We also see Exadata compared to All-Flash disk arrays such as Dell/EMC Xtremio and Pure Storage. How does Exadata stack up against these technologies? How can Oracle claim Exadata is faster?

Integrated Hardware & Software

To begin with, Exadata is a fully integrated system, while these competing technologies are simply one component. Exadata is comprised of:

- Clustered Servers
- Cluster Interconnect Network
- Storage Network
- Active Storage
- Software

Comparing Exadata to servers ignores the storage. Comparing Exadata to storage ignores the servers. The only valid comparison is to evaluate Exadata against another full-stack configuration of servers, networking, storage, and software.

Unique Integration + Portability

The Oracle DBMS software has always run on virtually ANY hardware platform. Oracle came of age in the era of Open Systems, where Oracle gained significant advantage over competitors by being portable across platforms that supported open standards. To this day, Oracle still runs on Solaris, AIX, Windows, and a wide array of Linux distributions including RedHat, SUSE Linux, and Oracle's own Linux distribution.

While we can expect Oracle to continue supporting Open Systems, that approach to product development has essentially run out of room for innovation. Oracle reached a point in the early 2000's where significant advancements were required to achieve the next level of performance. Integration of the Oracle DBMS software with a specific (Oracle-supplied) hardware configuration was necessary to move beyond the limits of open standards.

What is most surprising about the rise of Exadata is that Oracle has been able to achieve this tighter integration without jepoardizing portability and interoperability with existing platforms. Databases can be migrated back and forth between Exadata, AIX, Windows, Linux, and all platforms that Oracle supports. Those databases simply run better on Exadata.

Fast & Cost Effective

Oracle has focused primarily on the fact that Exadata is FAST and delivers the highest database performance of any platform possible. Database performance is obvious and easy to understand. Database performance has a direct (positive) impact on application performance, delivering benefits to business users that are quite obvious and easy to quantify.

What is less obvious about Exadata is that the same performance advantages also make Exadata extremely cost effective. The cost of Exadata needs to be compared to cost of servers, networking, storage, and software, not against one or another of these components. As we shall see in the following sections, the features that make Exadata fast also combine to make Exadata cost effective as well.

High Availability, Redundancy & Scalability

It's critical to understand that Exadata is a fully redundant system that provides high availability as well as scalability. This fundamental fact of systems architecture underlies the entire design of Exadata. Exadata is also able to deliver *HIGH PERFORMANCE* while still delivering these capabilities.

Other competing systems can deliver high performance, but only by sacrificing one of these attributes of high availability, redundancy, and/or scalability. For example, a system that puts all data in memory (DRAM) can provide high performance, but such a system cannot scale to larger data volumes, and simply cannot provide redundancy and high availability. Likewise, a system with Flash Disks on the internal system bus (such as PCIe) will provide high performance, but without scalability or high availability. These systems cannot scale beyond the amount of data that can be stored internally, and non-shared Flash disk storage eliminates the possibility of clustering.

Fast & Large Servers

From a pure "brute force" hardware standpoint, Oracle offers fast & large servers that equal or exceed anything on the market. Exadata currently comes in 2 flavors of Database Server, with 2-socket and 8-socket options. Exadata is constantly updated with the latest Intel CPU chips. The current generation X7 Exadata machines have 24 core Intel Xeon processors in both the 2-socket and 8-socket versions. Exadata comes with a MINIMUM of 2 database servers for redundancy. The smallest possible Exadata X7-8 (with 8 sockets) has 192 processor cores in each node, for a minimum of 384 cores.

By any measure, Exadata meets or exceeds the raw "brute force" processing power of any competing server on the market. Exadata then EXCEEDS those competitors through vastly better clustering technology that is enabled through hardware/software integration as outlined in this blog.

Large Memory Capabilities (DRAM)

The fast & large database servers within Exadata also come with large memory (DRAM) capabilities as well. The X7-2 model can be configured with as much as 1.5TB per node, or 12TB per rack in a standard "fractional" configuration. The DRAM is configured into the global database buffer cache that is effectively aggregated across nodes.

For the largest DRAM configuration possible on Exadata, the X7-8 database server nodes can be configured with up to 6TB of DRAM per node. We have run 4-node Exadata X7-8 configurations with 24TB of DRAM across all nodes with Oracle In-Memory Option, and larger configurations are certainly possible.

Optimized Device Drivers, O/S, Network, VM, DBMS

Oracle Development puts an enormous amount of effort into holistic optimization of device drivers, the Linux Operating System, Networking, Virtual Machine software, along with the Oracle DBMS software. These efforts are a major factor that has kept Exadata ahead of all other platforms for the past decade.

All-Flash Storage Option

Although most customers choose the Flash + Disk configuration of Exadata, Oracle also offers an All-Flash Storage Option. Oracle has invested heavily in Flash Cache capabilities that are outlined in this blog post, making the Flash + Disk configuration outperform the largest All-Flash disk arrays on the market. Flash + Disk in Exadata provides the best possible combination of price/performance on the market. However, some customers still need an all-Flash capability in some cases.

The All-Flash option of Exadata (known as Extreme Flash or EF) ensures that I/O will always be satisfied at the speed of Flash. The standard Flash + Disk configuration of Exadata will see the vast majority of I/O (95% or more) going to Flash in most cases. However, some I/O still hits spinning disk before the data is finally cached into the Exadata Smart Flash Cache. Oracle offers Extreme Flash for those situations where customers need a guarantee that 100% of I/O will come at Flash speeds all the time.

SPARC SuperCluster

Oracle's SPARC SuperCluster uses the same storage found in Exadata, so the same performance benefits apply to SPARC SuperCluster as well.

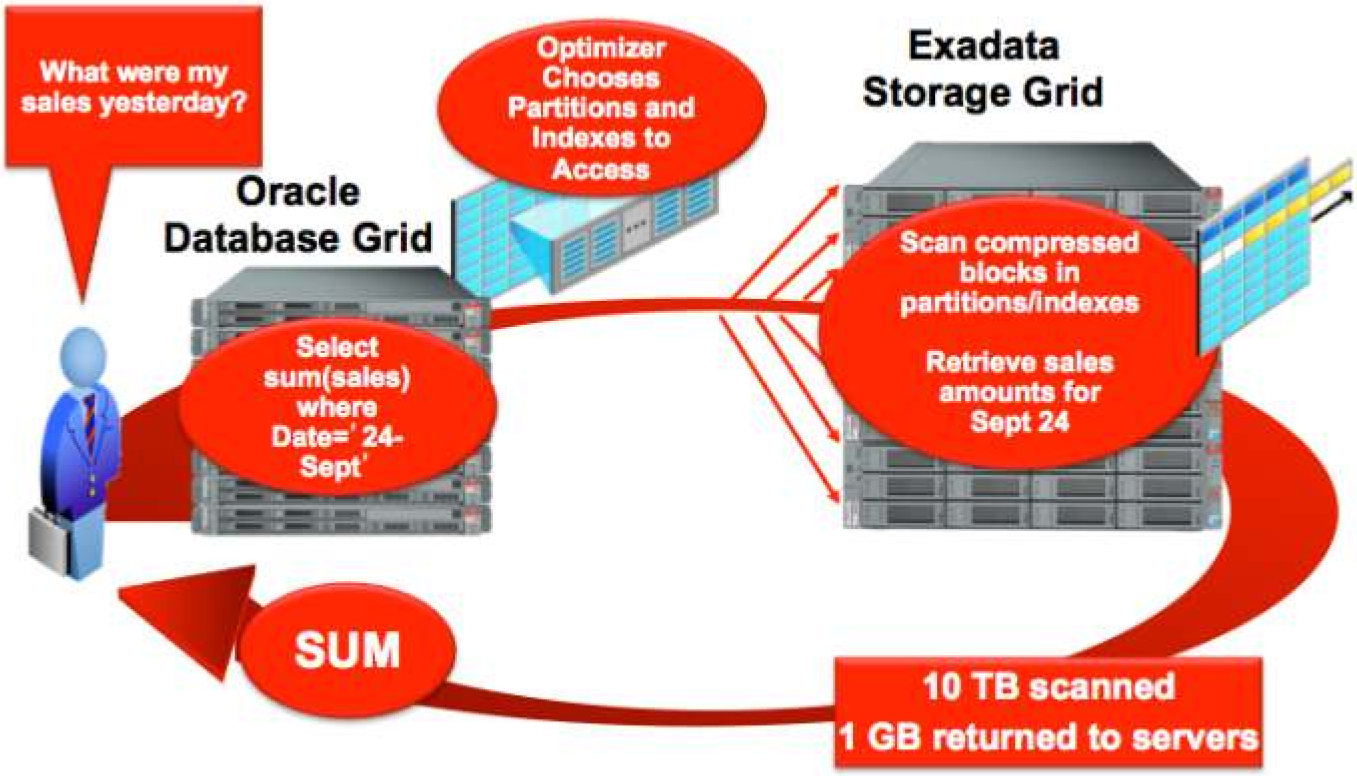
What Makes Exadata Special

After more than 10 years of continuous development, Exadata now has features big and small that make it the most effective platform possible for running Oracle databases. These features include:

- SQL Offload
- Active Storage – Cell Offload
- Massively Parallel Processing (MPP) Design
- Bloom Filters
- Storage Indexes
- High Bandwidth, Low Latency Storage Network
- High Bandwidth, Low Latency Cluster Interconnect
- Fast Node Death Detection
- Smart Flash Cache
- Write-Back Flash Cache
- Large Write Caching & Temp Performance
- Smart Flash Logging
- Smart Fusion Block Transfer
- NVMe Flash Hardware
- Columnar Flash Cache
- In-Memory Fault Tolerance
- Adaptive SQL Optimization
- Exadata Aware Optimizer Statistics
- DRAM Cache in Storage

SQL Offload – The 1st Exadata Innovation

The first innovation Exadata brought to the market when it was released in 2008 was the SQL Offload feature. All other platforms aside from Exadata work by moving data from storage into database servers. By contrast, Exadata works by moving SQL processing and other logic from the database server(s) into the storage layer. The following is a simple example that shows how Exadata handles SQL query processing.


What were my sales yesterday?
Optimizer Chooses Partitions and Indexes to Access
Exadata Storage Grid
Oracle Database Grid
Scan compressed blocks in partitions/indexes
Retrieve sales amounts for Sept 24
Select sum(sales) where Date=' 24-Sept'
SUM
10 TB scanned
1 GB returned to servers

It's important to note that this is ***ONE SIMPLE EXAMPLE*** that shows how query offloading works. Complex queries that involve querying and filtering data will have operations offloaded into the "active storage" layer of Exadata for processing. Query logic is moved to the data rather than moving data to the database server to be operated on by the query.

Exadata storage processes SQL fragments (component pieces of SQL statements) that are derived from the SQL execution plan. The Exadata storage then returns data to the database server in what is known as "row source format", which is a subset of both rows and columns from the underlying table. As discussed previously, Exadata has an extremely high bandwidth, low latency storage network. SQL Offloading means than Exadata also places ***less demand*** on this storage network than other architectures.

Active Storage – Cell Offload

Exadata has what can be described as an "Active Storage" layer, which means that database processing of all sorts happens within the storage layer. All other storage products in the market are what Oracle calls "block servers" that simply accept and process requests for I/O on database blocks or ranges of blocks.

The "Active Storage" layer of Exadata will act as a "block server" in some cases, but it is also able to process higher level commands from the database engine. Some examples of the "active storage" capabilities of Exadata include the following:

- SQL Offload
- XML & JSON Offload
- RMAN Backup (BCT) Filtering
- Data file vs. REDO I/O Segregation
- Encryption/Decryption Offload
- Fast Data File Creation

Oracle has pushed numerous database functions into the storage tier on Exadata, leveraging the "active storage" capabilities of the platform to provide higher performance than can be achieved on any other platform.

All-Flash disk arrays cannot achieve the performance that Exadata delivers, simply because that fast Flash sits behind a slow network. Even though Exadata has the highest bandwidth, lowest latency network in the industry (exceeding the speeds of Fiber Channel SAN networks), it's simply not enough to overcome the laws of physics and "speed of light" limitations. Cell Offloading **moves LOGIC to DATA** in order to take full advantage of the I/O capabilities of Flash.

To give an example of the horsepower available in the Exadata storage layer, the standard fractional configurations of Exadata (Quarter, Half, and Full Racks) deliver an additional 7/10 (70% more) processing power in the storage tier for each core in the database tier. For example, a full rack of Exadata X7-2 has 384 processor cores in the database tier, and 280 processor cores within storage.

Massively Parallel Processing (MPP)

Some of Oracle's competitors still highlight their "shared nothing" architecture and the fact that Oracle database (on non-Exadata platforms) is a "shared everything" architecture. This has often risen to the level of a religious war, with some people claiming that Exadata isn't really (truly) a Massively Parallel Processing (MPP) platform.

For some background, consider this…

In a **shared nothing** design, each node of an MPP cluster owns a set of data and handles all processing of data that resides on that node.

In a **shared everything** design, each node of the cluster can access all data, and data can be processed on any node.

While the Oracle database on non-Exadata hardware is a "shared everything" design, Exadata is really a hybrid of both approaches. Each Exadata storage cell owns a set of data and handles all processing on that data. Database functions that are more data intensive (process large volumes of data) are offloaded into the storage layer. Other functions of the database are processed in the central core or processing cluster of Exadata.

The argument that Exadata isn't truly a "Massively Parallel Processing" (MPP) platform is quite silly based on that fact a Full Rack of Exadata has 674 processor cores. If that amount of processing power isn't "massive", consider the fact that Exadata can scale to 18 racks wide, or 12,132 processor cores. One piece of so-called "evidence" that competitors cite is how JOINS are processed in Exadata, but customers need to understand Bloom Filters to understand the implications.

Bloom Filters

Exadata does not directly process JOIN conditions inside the Exadata storage layer, but this does not tell the full story. Exadata makes use of "Bloom Filters" to facilitate JOIN processing. Filter criteria from the JOIN is pushed into the Exadata storage tier to filter the data. According to the processing methodology outlined by Burton Howard Bloom (the father of Bloom Filters), this technique is a probabilistic approach that will return a small percentage of false-positives, but will filter-out the vast majority of non-qualifying data.

Bloom Filters allow Exadata to offload the vast majority of data filtration related to SQL JOIN processing. While the final process of merging data into a single result set is done on the central RAC cluster of Exadata, the vast majority of processing (data filtration) is performed on the cells using Bloom Filters.

Storage Indexes

Exadata internally generates indexes of data value ranges on Oracle database blocks stored within each Exadata Storage Cell. These are referred to as Storage Indexes, and allow Exadata to bypass database blocks that don't contain values specified in the SQL where clause or join criteria.

Storage indexes are often used for secondary index access on the Exadata platform, and often eliminate the need for secondary b-tree indexes. Oracle recommends that customers create PRIMARY KEY and UNIQUE KEY indexes where required to enforce uniqueness. Oracle also recommends that customers create Foreign Key Covering Indexes because those will represent the most likely access paths. However, beyond those simple indexing rules of thumb, Exadata typically will not require additional indexes.

It's also important to note that PARTITIONS also represent what is essentially a coarse-grained index, and SUB-PARTITONS serve essentially the same function as well. Beyond basic indexing and data partitioning, Exadata's Storage Index feature generally provides all that is needed for final data indexing to deliver needed performance.

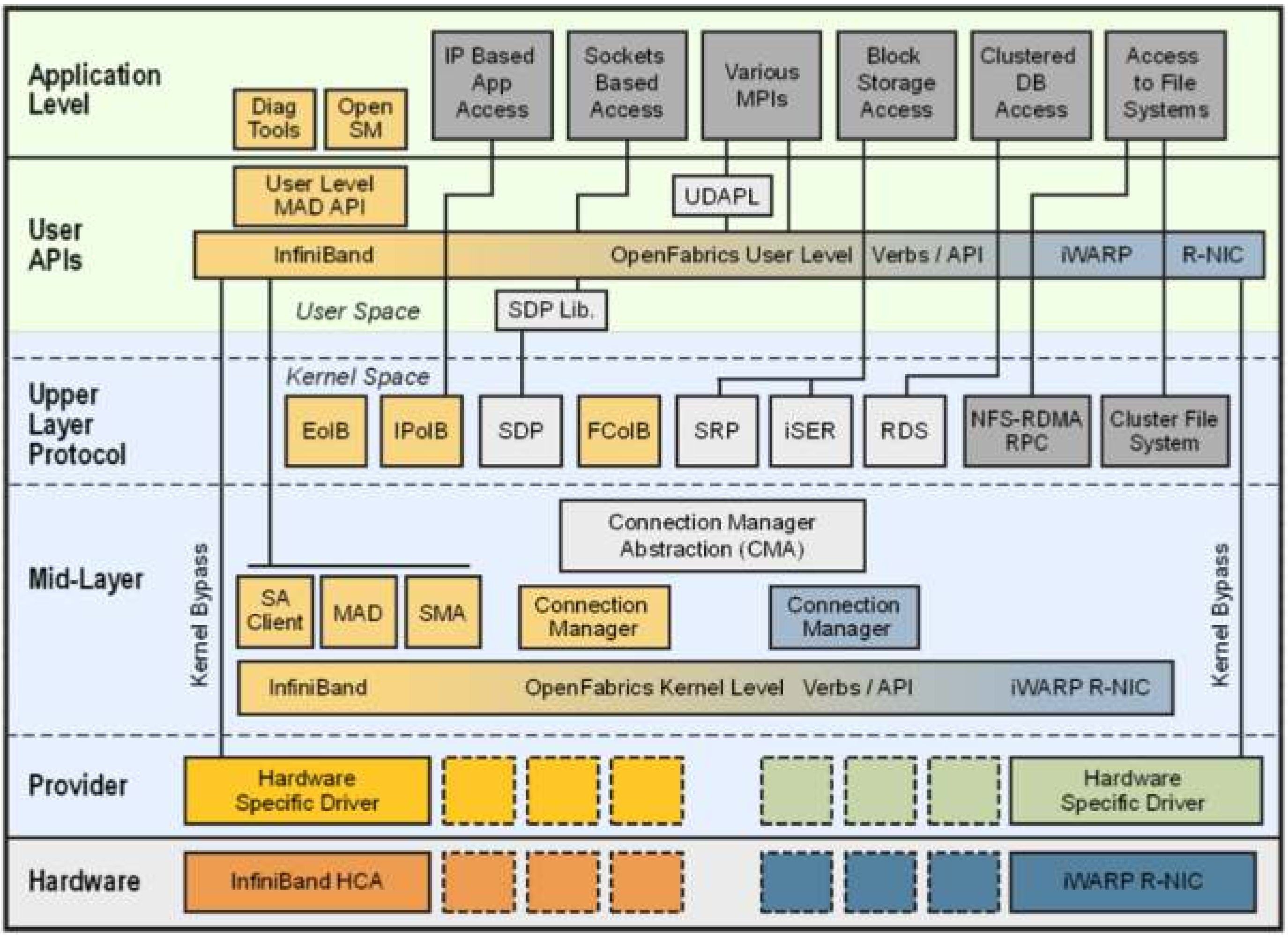
High Bandwidth, Low Latency Storage Network

Exadata has an internal storage network that provides both high bandwidth and low latency. This storage network is currently based on Infiniband, but the specific physical technology used inside of Exadata isn't nearly as important as what it delivers. The storage network of Exadata is the "blackplane" of the system, and is isolated from the customer's data center network(s). Customers should retain this isolation to make future upgrades of Exadata easier without disrupting the large data center network, and without requiring upgrades to the data center network. Put another way, customers should avoid extending the Infiniband network of Exadata unless absolutely necessary.

The current Infiniband network inside of Exadata is Quad Data Rate (QDR), or 40 Gigabits per second, and is bi-directional for overall 80 Gbps. Bandwidth to and from storage is critical for database operations that involve large data movements, such as data loading (usually for Data Warehouse) and large data extracts from the database (ETL outbound to other systems). Notice that I didn't highlight "table scans" in this case, because Exadata uses SQL Offloading as discussed in the previous section.

While I/O bandwidth to/from storage is certainly important for certain database operations, I/O latency is also critical for many other operations. Single block or single row lookups are examples of cases where I/O latency can be critical. Even more critical is I/O latency for REDO LOG operations. Redo log writes are especially sensitive to I/O latency, and even more sensitive when transaction COMMIT is involved. The DBMS must externalize COMMIT markers to storage before transactions can proceed, so low latency is critical on these types of writes.

Oracle has been a big contributor to the advancement of Infiniband, and Oracle will continue to advance networking technology in the industry. One example of this is the advent of RDMA (Remote Direct Memory Access), which highly benefits Exadata in terms of I/O latency.



High bandwidth, low latency access to storage is critical to Oracle database performance, and

Exadata delivers these capabilities better than any other platform. The integration of hardware and software in Exadata are what make this possible. The following white paper provides an in-depth look at this topic.

http://www.oracle.com/technetwork/server-storage/networking/documentation/o12-020-1653901.pdf (http://www.oracle.com/technetwork/server-storage/networking/documentation/o12-020-1653901.pdf)

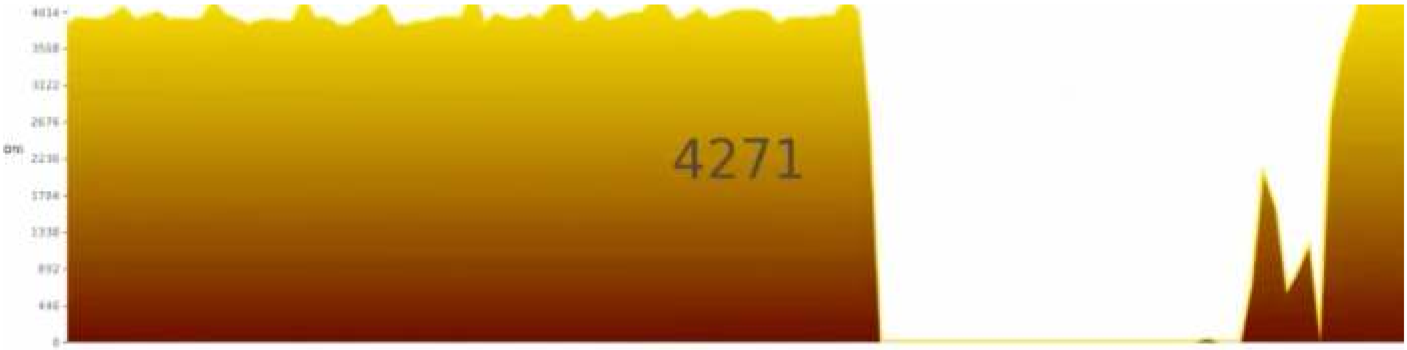
High Bandwidth, Low Latency Cluster Interconnect

Exadata uses a UNIFIED network for all internal communication, including access to storage as well as for the cluster interconnect. The previous section on storage networking within Exadata also applies to the cluster interconnect. However, most other platforms use a SEPARATE network for storage and cluster interconnect. All systems that are based on Fiber Channel SAN technology (Storage Area Networks) will have a separate IP (Ethernet) network for the cluster interconnect.

We have seen that the cluster interconnect often provides critical performance advantages in a active/active clustered system. In fact, the cluster interconnect can be vastly more important for some workloads than performance of the storage network. We have seen Exadata achieve more than 2X higher OLTP transaction throughput (measured at the application layer) compared to other systems, purely because of the cluster interconnect performance differences.

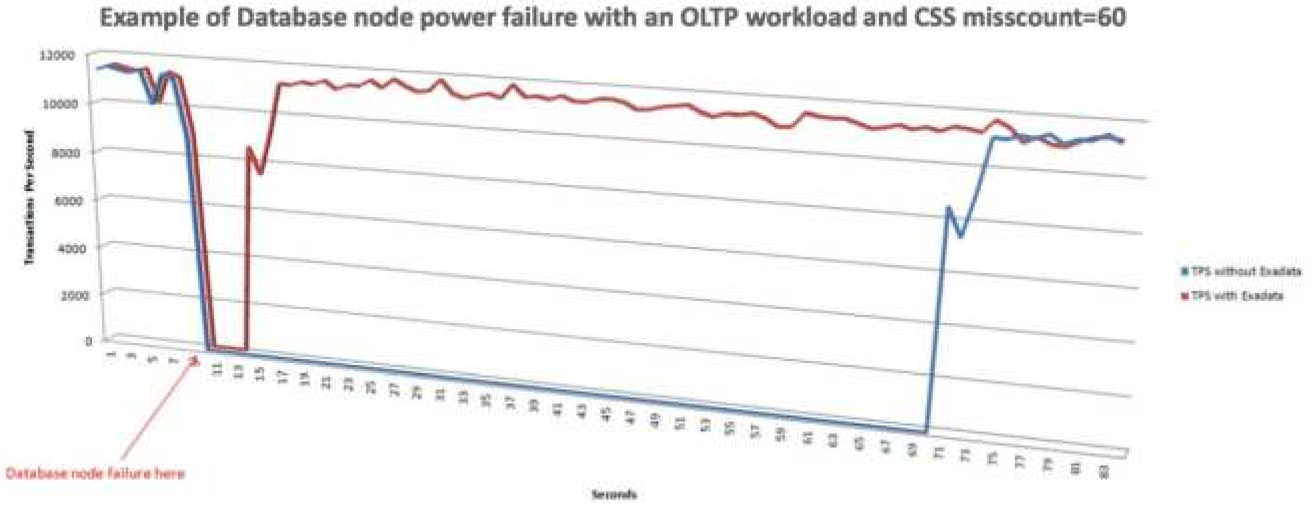
Fast Node & Cell Death Detection

The typical cluster interconnect network uses standard IP (Internet Protocol) over ethernet, which is high performance and supported across virtually all platforms, but brings some disadvantages. Even in earlier versions of Exadata, the ability of the system to quickly detect failure of nodes within a cluster was hampered by network timeout waits. The following graph from testing by Oracle's MAA (Maximum Availability Architecture) team in 2013 shows the issue with node failure detection.



This graph shows a 30-second delay in transaction processing, while the system waits for network timeouts before it can determine that a node has died. The full video by Oracle's MAA team can be found on Vimeo at this link (HERE (https://vimeo.com/62754145)). Other platforms might seen network timeouts lasting as long as 120 seconds, which can be a significant business impact on large scale systems. While Exadata was certainly better than other systems back in 2013 when this video was produced, Oracle has since improved upon this capability with the Fast Node Death Detection feature.

As we said earlier, Exadata is an integrated hardware/software solution, and Fast Node Death Detection is a stunning reminder of this fact. Rather than relying on network timeouts to determine if a node has failed, Exadata moves the cluster member oversight function into the cluster interconnect network. The cluster interconnect network (Infiniband) is now responsible for detecting the death of nodes in each cluster. The following graph shows how node death detection is vastly improved.



This feature also applies to detection of cell death in an Exadata environment, resulting in much faster system reconfiguration and the lowest possible application/business impact possible.

Smart Flash Cache

The best way to understand Exadata's Smart Flash Cache capability is to first compare it to the "tiered storage" concept used by Storage vendors. Tiered storage is an attempt to deliver a better combination of price and performance as compared to a single type of storage. Tiered storage moves "hot" data to faster media, such as moving from spinning disk to Flash storage. Storage tiering can be accomplished through manual data placement, but some vendors have attempted to automate movement of data between slower and faster tiers of storage, typically with corresponding higher and lower storage density, such as spinning disk versus Flash storage.

Instead of using a "tiered storage" concept, Exadata automatically CACHES data into Flash instead of moving the data.

Write-Back Flash Cache

In the early days of Exadata, the Flash Cache was used to provide improved response for READ operations. However, Oracle later enhanced this feature to also cover write operations. Most database writes are done "asynchronously" from end-user transactions, so improving write performance doesn't always improve application performance. As long as the Oracle database writer process (DBWR) stays ahead of transactions and clears dirty blocks from the buffer cache fast enough, write operations won't slow the database.

Although it's true that most writes are not time sensitive in the Oracle database, write operations do become critical in some cases. Redo writes in particular are quite sensitive to latency. See the section of Smart Flash Logging for more information.

TEMP writes occur when SQL statements sort data using an ORDER BY clause, and when the amount of data being sorted is too large to fit into memory. In these cases, each "sort run" is written from memory onto storage, and those writes should be cached. See the section on Large Write Caching for more information on this topic.

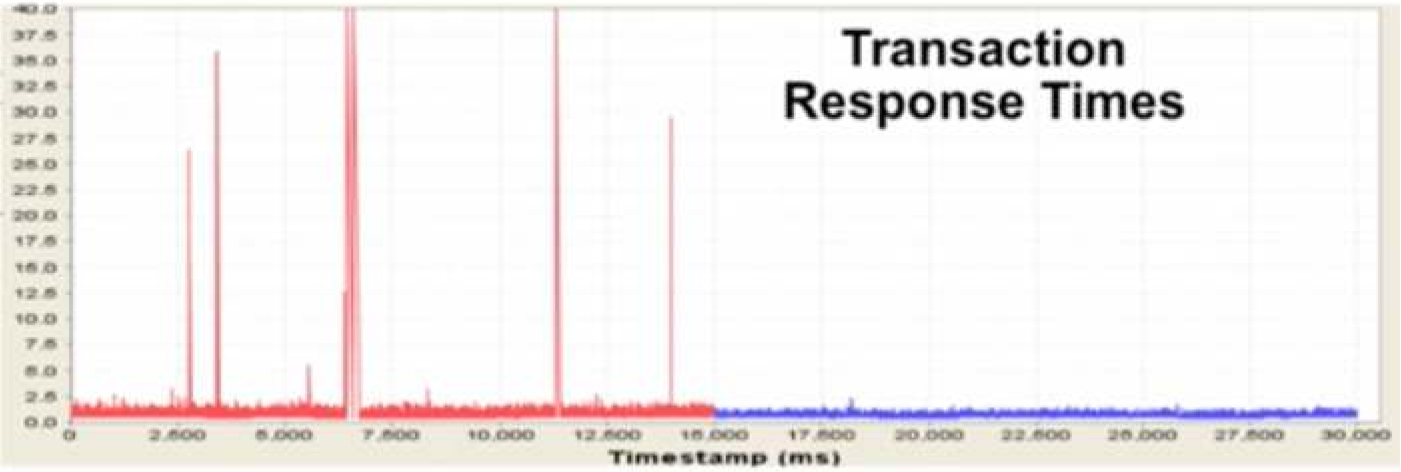
Beyond redo writes and large writes (especially for TEMP processing), the Oracle database on Exadata can still gain performance through use of write-back Flash Cache. Use of Write-back Flash Cache requires that ALL mirror copies are cached, meaning that TWO copies of each write are stored in Flash Cache with NORMAL redundancy, or THREE copies are stored in Flash Cache with HIGH redundancy. Earlier versions of Exadata had smaller capacity Flash hardware. As "Moore's Law" has brought larger and larger Flash modules, Exadata now offers an even larger Flash Cache than with earlier versions.

Large Write Caching & TEMP Performance

Exadata even caches large writes (128KB and larger) into Flash Cache, taking full advantage of Flash capabilities. As the hardware has evolved and larger Flash drives have come on the market, Oracle has also evolved the Exadata software to match these capabilities. For more information on this topic, please see Section A.2.6 of the Exadata manual details (HERE (https://docs.oracle.com/cd/E80920_01/DBMSO/exadata-whatsnew.htm#DBMSO21849)).

Smart Flash Logging – Fast OLTP

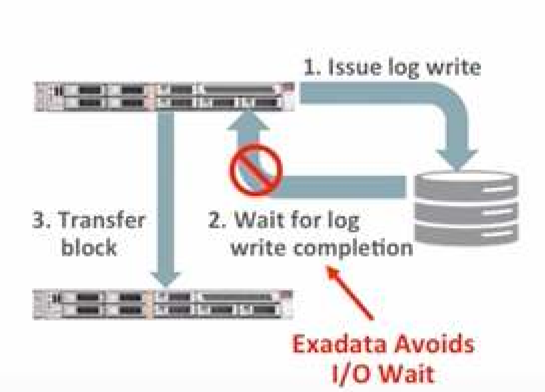
OLTP performance often suffers from spikes in latency of redo log writes. While average redo write performance might be good, these periodic spikes in latency often have a detrimental impact on overall application performance. This can be easily seen on systems by looking at AWR historical data. It's critical to look beyond the averages and look for spikes, which are often seen in the standard deviation of redo log write latency. The following graph shows the existence of spikes in redo write response times that impact transaction response, and how these spikes are corrected by this important Exadata feature:



Using unique hardware/software integration, Exadata is able to treat I/O activity for redo logs in a distinctly different manner than I/O for other database structures. Exadata knows which I/O's are highly critical to database performance and treat those I/O's differently from less critical I/O. Generic disk or Flash storage technologies lack this tight integration with the Oracle database. The result is faster OLTP application performance. The following video provides a deeper explanation of this critical feature.

Smart Fusion Block Transfer – Fast Clustered OLTP

Smart Fusion Block Transfer is an Exadata-specific feature that improves performance of OLTP applications running across clustered database nodes within Exadata. This feature requires close synchronization of processing within the Oracle database, including how log writes are tracked across nodes within Exadata. Exadata is able to transfer database blocks across nodes without waiting for completion of redo log writes.



While this is a highly technical feature that requires deep understand of the inner workings of the Oracle database, customers simply need to understand this feature results in higher performance of OLTP applications on Exadata. The following YouTube video provides some more insight into this feature.

Non-Blocking Transfers of OLTP Data on Clusters

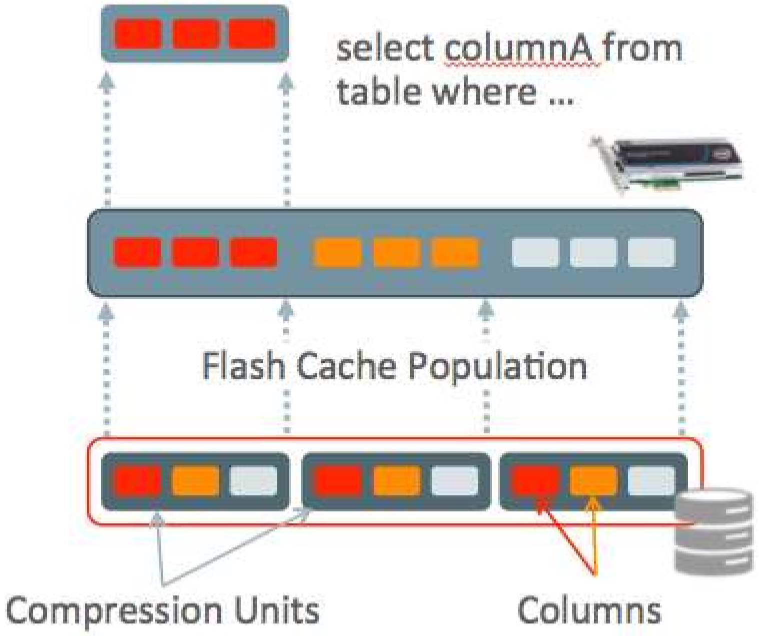
NVMe Flash Hardware

The fundamental architecture and design of the Exadata hardwre allows Oracle to quickly adopt the latest technology innovations as they are introduced into the market. Exadata internally is a very modular design that can accept hardware revisions more quickly than large, monolithic server designs that come from some Oracle competitors.

The introduction of NVMe Flash in Exadata is only the most recent example of how Oracle has very quickly adopted new hardware technologies to keep Exadata ahead of the competition. Non Volatile Memory Express (NVMe) is the latest advancement in Flash disk technology. Rather than essentially “emulating” the behavior of a spinning disk (as can be found in Flash SSD), NVMe Flash uses a protocol that is designed from the ground-up to support Flash.

Columnar Flash Cache

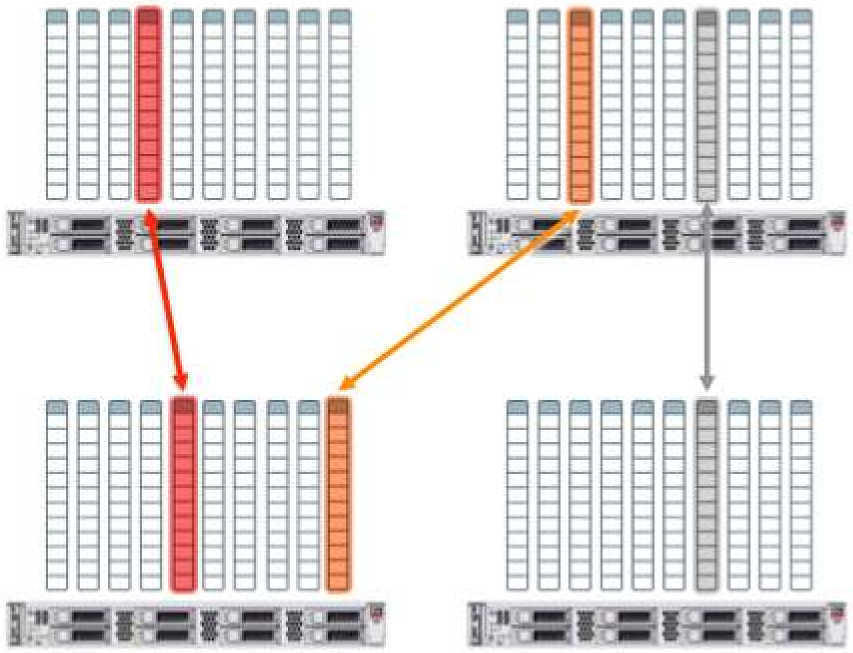
Exadata automatically transforms HCC data into pure columnar format when it places that data into Flash Cache. Data on disk in HCC format is stored in "Compression Units", which represent an unnecessary fragmentation of data once it's brought into the Flash Cache. The following diagram shows how data is represented differently in Flash Cache without being organized into Compression Units:



Organizing HCC data into a pure columnar format will increase performance by 5X on queries that access data in the Flash Cache. This feature also facilitates bring data into the DRAM cache using the In-Memory capabilities of Exadata.

In-Memory Fault Tolerance

The Oracle In-Memory database option is not specific to Exadata, but there are significant advantages to running In-Memory on the Exadata platform. Chief among these advantages is the Exadata specific In-Memory Fault Tolerance feature. The following diagram shows this feature graphically:



In-Memory Fault Tolerance means that data can be effectively mirrored across nodes of the database cluster inside of Exadata. This feature gives customers more options for using the DRAM buffer cache on the nodes of the cluster. In the event of a node failure, the contents of another node can be used to satisfy the query.

Adaptive SQL Optimization

Although not exclusively an Exadata feature, Adaptive SQL Execution is extremely critical for Exadata, since the majority of Oracle Data Warehouse deployments have moved to the Exadata platform. Version 1 of Exadata was exclusively targeted at Data Warehouse deployments, and many Exadata features remain firmly targeted at Data Warehouse workloads. Adaptive SQL Optimization applies specifically to the optimization of highly complex SQL statements.

The Oracle database engine was originally designed for OLTP (prior to Oracle 7.1.4), and the OLTP heritage was apparent in how it optimized SQL statements. The Oracle SQL optimizer originally operated in a single-pass manner. As the term implies, the optimizer would execute once at the start of each SQL statement execution, evaluating a limited number of SQL execution plans. We could play with things like the maximum number of execution plan permutations, but the SQL optimization would only occur once. SQL optimization needs to happen quickly, and generally in less than around 250msec, or it will appear to users as a performance problem.

Adaptive SQL Optimization means that it’s no longer a single-pass operation. What generally happens is that the SQL plan is accurate to a certain number of levels, but intermediate result set cardinality estimates can be inaccurate with highly complex queries. The cardinality (number of rows) in an intermediate result set will often determine the access path of the next level in the query plan. If Oracle detects a significant variation in the actual vs. estimated intermediate result set cardinality, the SQL statement will be re-optimized on the fly.

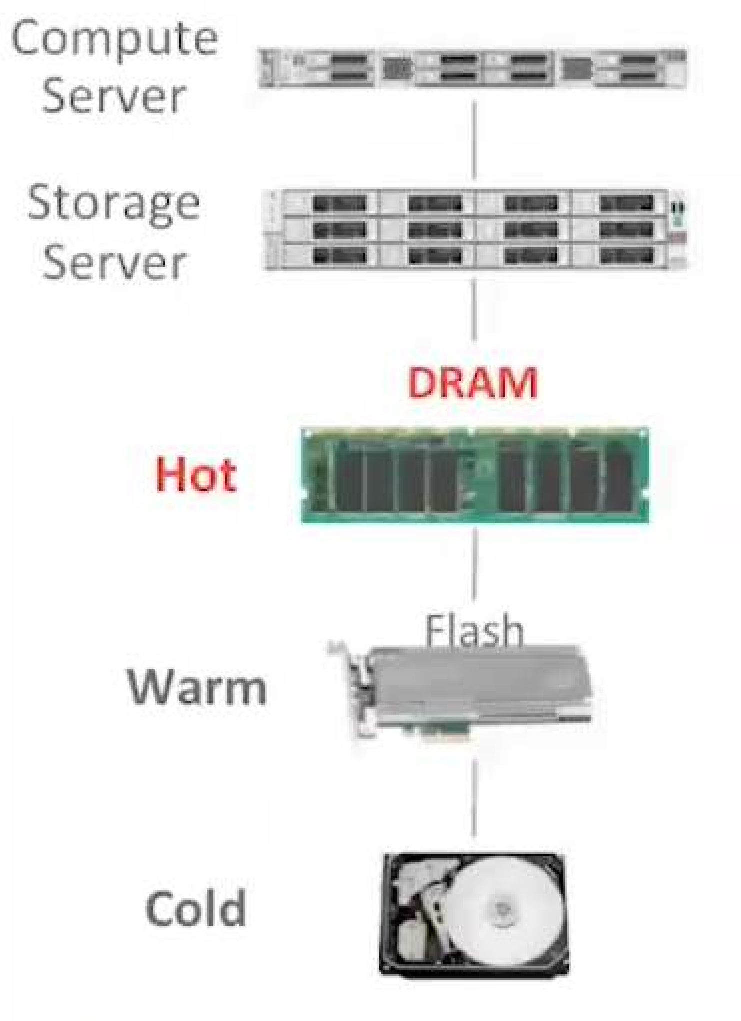
Of course the fundamental accuracy of SQL execution plan selection is highly dependent on optimizer statistics. Adaptive SQL Optimization should help to overcome issues with stale optimizer statistics, but a better solution is to gather proper statistics in the first place. Fortunately, Exadata helps with statistics gathering as well.

Exadata Aware Optimizer Statistics

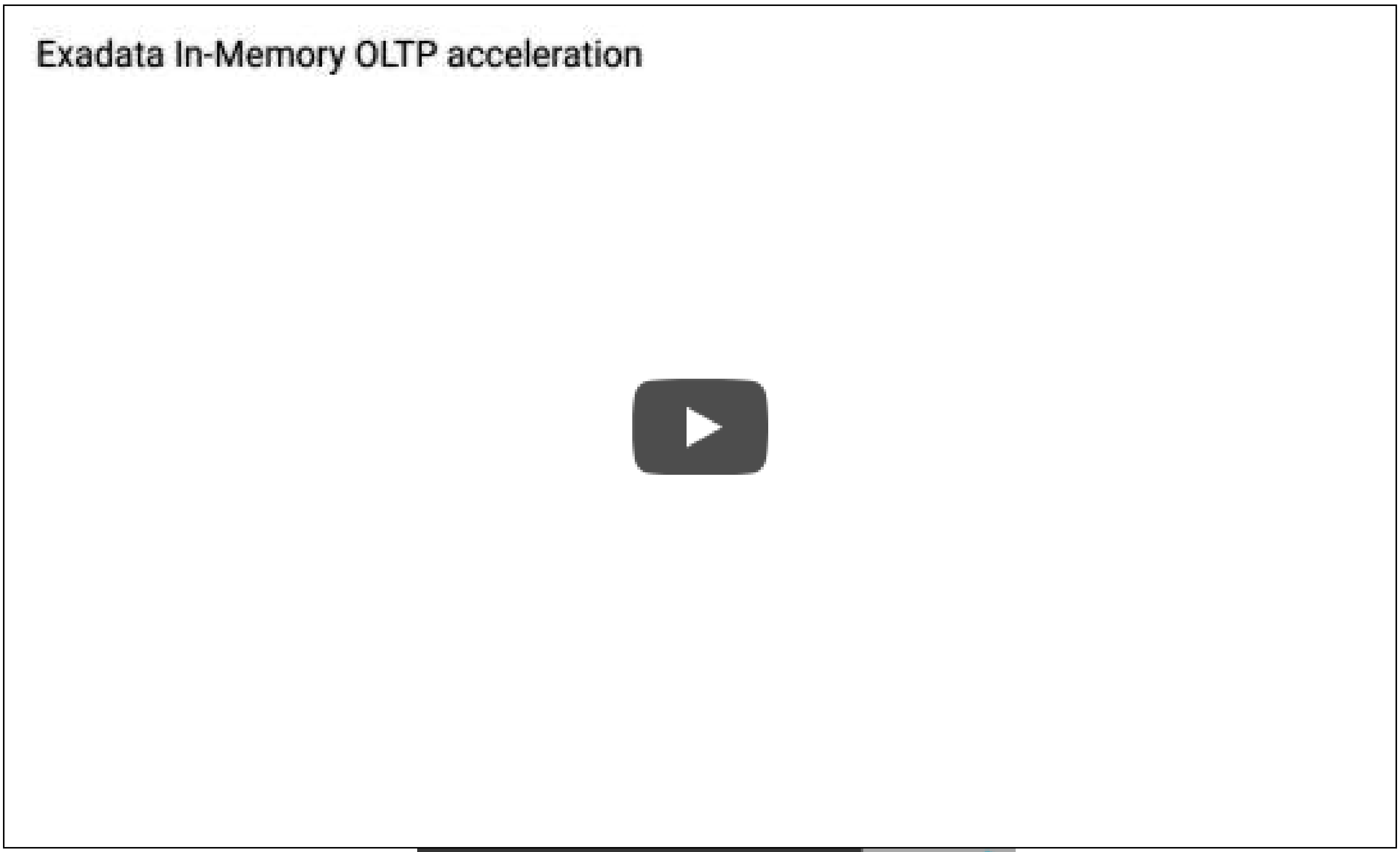
The Oracle SQL Optimizer is well aware of the capabilities of the Exadata platform. Optimizer statistics are gathered using the DBMS_STATS package, and those statistics consider the capabilities of Exadata.

DRAM Cache in Storage

While Exadata Smart Flash Cache has been around since Version 2, the latest versions of Exadata also use DRAM for caching data at the storage tier. While the Exadata Smart Flash Cache was already fast, the new DRAM caching at the storage layer provides an additional 2.5X faster data access.



As shown above, the DRAM cache in Exadata storage provides another layer of automatic data caching to speed execution of database I/O, especially for OLTP applications. The following video provides more detailed explanation of the DRAM cache in the Exadata storage layer.



Exadata In-Memory OLTP acceleration

AdChoices
ORACLE + Dyn
Dynamic DNS
BUY NOW
$40/yr
Report this ad

2 thoughts on "**What Makes Exadata Faster?**"

1. **ARNIE DASGUPTA** says:
   February 10, 2018 at 1:36 am

   Good Article Chris. Nice and Concise. Would be interesting to see how Exadata adopts 3D Xpoint and Optane. Will wait for your article on that.

   Reply (https://chriscraftoracle.wordpress.com/2018/02/09/what-makes-exadata-faster/?replytocom=7#respond)

   ○ **chriscraftus** says:
   February 10, 2018 at 6:02 pm

   Arnie:

   Even the current generation of Flash is much too fast for systems to effectively use that capability in the database tier. Offloading SQL operations and other database tasks is the only way to use the full I/O capability of Flash.

   Regarding the next generation of Flash, I can tell you that Oracle is constantly evaluating new generations of each new component. Oracle has developed software models of the overall architecture to aid in the evaluation of those new generations of each component. This includes spinning disk, solid state storage, DRAM, networking, system bus, new processors, etc. We've seen over the past 10 years that the Exadata system architecture has allowed Oracle to adopt new hardware components very quickly as those components hit the market, are proven to be stable, with sufficient supplies, and at the right price point to keep Exadata competitive.

   Reply (https://chriscraftoracle.wordpress.com/2018/02/09/what-makes-exadata-faster/?replytocom=8#respond)