An Oracle White Paper
[February 2012]

Oracle Exadata Technical Case Study

Extending a Half-Rack Exadata Database Machine to a Full-Rack with Zero Downtime

Executive Overview

A leading biotech firm in California has successfully deployed Oracle Exadata Database Machine (Exadata) for their sales and marketing data warehouse. By using Exadata, this firm has been able to rapidly develop their data warehouse and reporting applications for their end-users at unprecedented speed. They have also realized substantial performance improvements leading to lower total cost of ownership due to out-of-the-box performance improvements minimizing the time and effort needed to tune many of their critical queries. For example, one typical query reduced their execution time from 641 seconds to 2 seconds. An ETL job was reduced from 90 hours to 2 hours and another decreased from 36 hours to 30 minutes. Many jobs that could not complete at all were able to complete in a few minutes with Exadata.

The customer has also benefited by leveraging their DBA's existing knowledge of Oracle Real Application Clusters (RAC) and Automatic Storage Management (ASM) to administer Exadata. Migrating to Exadata was a relatively simple effort since most of the systems feeding the data warehouse on Exadata were also Oracle databases and standard data migration utilities could be used.

With the success brought by the implementation of Exadata came additional growth as existing projects added more users, and other projects were initiated on the Exadata environment. Their Exadata environment needed to grow from the initial two quarter racks to full racks; however, the aggressive development timelines and round-the-clock usage of the production databases required the Exadata extensions to occur without downtime.

This technical case study describes the latter phase of the biotech firm's Exadata extension of their development/user acceptance testing system from a half to a full rack that occurred without downtime. All phases of the extension were performed without downtime.

Intended Audience

Readers of this paper are assumed to have experience with Oracle Database 11g technologies, familiarity with Oracle's MAA (Maximum Availability Architecture) framework, and a general technical understanding of the Oracle Exadata Database Machine. If referenced in this paper, these topics will not be explained, as they are covered in other papers. See the Appendix for a list of recommended technology white papers and acronyms used in this paper.

Background

The biotech firm implemented their sales and marketing data warehouse on Exadata to accomplish the following objectives:

- Improve "time to market" for new reports by reducing the need for aggregations or materialized views
- Significantly improve query, reporting, and ETL performance
- Permit ad-hoc queries with good performance (ad-hoc queries were not possible before Exadata)
- Meet or exceed mobile device performance SLAs
- Significantly lower the Total Cost of Ownership (TCO)
- Minimize additional support and administrative overhead
- Maintain compatibility with existing data integration, reporting and EII tools

The achievement of the objectives were made possible through Exadata by:

- Delivering excellent performance as witnessed during the Proof of Concepts / competition phase
- Providing excellent scalability for parallel execution
- Avoiding development of most materialized views (MVs) for new applications due to Exadata's speed. Each MV avoided saves one person-week of development time.
- Lowering TCO by minimizing tuning efforts and leveraging existing Oracle licenses and skills
- Minimizing the implementation and conversion efforts since some parts of the application had data that was already managed in an Oracle RDBMS.
- Protecting the investment in Exadata by enabling the initial Exadata V2 system to be extended with the more current X2-2 technology (M2 series cell and compute servers).

The Sales and Marketing Data Warehouse

The customer acquired the Oracle Exadata Database Machine to implement their next-generation sales and marketing data warehouse. The data warehouse manages:

- Customer master data
- Customer alignment with physician's geographical data
- Customer alignment with sales representative's geographical data

This data warehouse is used to analyze and report on:

- Sales data for all products
- Case management and customer payments
- Market/competitive data and analysis
- Sales force automation data

The overall data flow through Exadata is:

Source data => Landing + Staging => DW => Data Mart => End users

Source data: Multiple external and internal sources with various formats.
Landing: Straightforward 1-to-1 mapping of the source data to staging tables using Microstrategy to load the data.
Staging: Operates on data in staging tables to cleanse and standardize data.
DW: In-house designed and customized, highly normalized data warehouse.
Data Mart: The Data Mart is the final data in a form for end users/application to access.
End users: Business Objects, SAS, and internally developed mobile applications.

The Exadata Configuration

The customer initially ordered two quarter-rack Exadata V2 systems in Q1CY2010: one for development and testing, the other for QA and production. This was enough to get started with their development and initial production deployments, but would require additional capacity as more applications were rolled out to more of their user community. Oracle's best practice is to have separate development, QA, and production environments for isolation; however, this customer decided that they could use the somewhat mixed environment for the time being.

A key driver for the customer's choice of Exadata was the reduction in development time that resulted when they realized that most, if not all, materialized views would not be needed with Exadata. This was due to Exadata's extremely fast performance in processing queries in real-time that formerly could only be satisfied with materialized views. The firm estimates that each materialized view required one person-week to develop, test, and deploy; when multiplied across their complex applications, the savings in development time and overall project development time was substantial and allowed the customer to bring their Data Warehouse project on-line faster.

This customer found during the development process that they could achieve substantial performance gains by implementing various ETL processes in the database as PL/SQL procedures, rather than as external batch jobs. This was possible with Exadata because sufficient CPU capacity was available in the compute nodes due to much more efficient storage cell offloading that occurred in many of their

queries. Formerly, without Exadata offloading capabilities, a huge amount of data was processed at the database nodes rather than at the storage cells. Placing the ETL processes in the database resulted in an 8X improvement in performance overall for ETL jobs. The table below shows examples of other jobs along with their improved times compared with the previous system performance based on Hewlett Packard blade servers and SAN storage systems.

Activity	Timing Before Exadata (minutes)	Timing After Exadata (minutes)	Improvement
Two Long-running ETL Jobs	6480	90	72X
Situational Analysis Detail	641	2	321 X
Account Trend Report	257	1	257 X
Competitive Activity Report	265	9	29 X
Geography Details Report	45	8	6 X
Field Performance Report	65	31	2 X
Market Share Report	Could not run	30	Infinite
Brand Performance Report	Could not run	258	Infinite
Performance Ind. Report	Could not run	70	Infinite

Further gains were made by query tuning (typically removing indexes and adding indexes where appropriate), implementing parallel execution (previously most queries ran serially), and utilizing Hybrid Columnar Compression (HCC) extensively on partitioned tables. The HCC compression ratios averaged about 10X for tables compressed with "for query" and about 15X for tables compressed with "for archive", while fact tables averaged a 31X compression ratio.

Extending the Exadata Quarter-Rack to a Full-Rack

The customer was pleased by the application performance improvements s that were achieved using Exadata, and now wanted to ensure that their production and test systems would be ready to meet future growth. This meant that the original production quarter-rack (two compute nodes and three storage cell nodes) that were purchased would need to be extended to a full-rack (eight compute nodes and 14 storage cell nodes). The original quarter-rack development/user acceptance testing system would be extended to a half-rack (four compute nodes and seven storage cell nodes), and later to a full-rack. The extensions were performed in various stages, with the final stage occurring in October 2011, when the original V2 half-rack development system containing X4170 and X4275 nodes was extended to a full rack using X4170 M2 and X4270 M2 nodes. This paper focuses on this later effort.

The customer set very aggressive development and production application rollout goals which required the extension activity to be done while the systems were in use; there were only very small downtime windows that were available. This was as true for their development/test system as it was for production, because developers were busy around the clock to meet their application development schedules.

Planning

Oracle software and hardware engineers worked closely together and with customer's support staff to plan the Exadata extension and ensure it would go smoothly and without interrupting existing workload. There were three main areas involved in the planning:

1. Patch planning
2. Network planning
3. Task planning

Patch Planning

The new Exadata nodes were delivered with Exadata Storage Server Software version 11.2.2.3.5 which was the most recent version at that time. One key consideration is that the X4170 M2 and X4270 M2 DB nodes have always shipped with OL 5.5; OL 5.3 is not supported on the M2 hardware. Furthermore, if OL 5.5 is used on the DB nodes, then the minimum Exadata Storage Software version that is recommended is 11.2.2.3.5 (in order to use OFED version 1.5.1-4.0.50 or greater). This meant that the existing X4170 and X4275 nodes would need to be patched to 11.2.2.3.5 and the DB nodes would need to be upgraded from OL 5.3 to OL 5.5. Finally, it was decided to install 11.2.0.2 BP12 to obtain the latest fixes at the time, and test them out on this environment before applying BP12 to the production system. In summary, these were the patches that were applied to the existing nodes:

Component	Original Release	Target Release	Reason	MOS Notes
Oracle Enterprise Linux	5.3	5.5	New X2 DB nodes have only been tested with OL 5.5; Must upgrade existing OL to match new nodes	888828.1 1284070.1
Exadata Storage Server Software	11.2.2.2.0	11.2.2.3.5	OL 5.5 requires the use of OFED version 1.5.1-4.0.50 or higher available with 11.2.2.3.5 or higher	888828.1 1334254.1
Grid and Oracle Homes	11.2.0.2 BP5	11.2.0.2 BP12	Latest release desired	888828.1

All of these patches could be applied in a rolling fashion with no application downtime; however, to finish the patching effort in a smaller window, the customer chose to take a few hours of downtime to apply the patches on the weekend before the extension activities were scheduled. It is a best practice to review the following notes before planning and undertaking Exadata patching:

- MOS note 888828.1: Database Machine and Exadata Storage Server 11g Release 2 (11.2) Supported Versions
- MOS note 1270094.1: Exadata Critical Issues
- MOS note 1262380.1: Exadata Patching Overview and Patch Testing Guidelines

Oracle recommends applying Exadata patches on a quarterly basis to ensure that critical fixes are applied proactively.

Network Planning

Network planning is needed simply to obtain IP addresses for the new nodes. A worksheet showing the existing node names and addresses with spaces for the additional node information was given to the customer to fill out with new node names. The network information for just the new nodes was entered into a new configuration spreadsheet from which configuration files were generated that were used to verify the proposed new IP addresses. The generated files were also used for the initial configuration of the new nodes along with other data collected early in the planning process that was used to size the grid disks and ASM diskgroups. The grid disks on the new cells should match the size of the grid disks on the existing cells.

Task Planning

A key aspect of task planning was the use of a detailed plan to account for every activity needed from the delivery of the equipment through the turn-over of the system to users. This plan listed each activity, the responsible party, amount of time needed to do it, and whether or not downtime was required or optional.

The *Oracle Exadata Database Machine Owner's Guide* (DBMOG), Chapter 8 "Extending Oracle Exadata Database Machine" was used to guide the technical steps for implementing the work. The customer wanted to have an overall plan of the hardware delivery, patching, and Exadata extension for project planning purposes so the following planning spreadsheet was created (only an excerpt shown for brevity):

ID	Date	Type	Activity	Person	Duration (HRs)	Downtime?
1		Other	Customer receives equipment	Customer		N
2		Other	Gtech tags and moves equipment to RWC	Customer		N
3	Wednesday, September 28, 2011	Prereq	**Fill out configurator spreadsheet for the new nodes coming in** - Customer to verify new IPs do not conflict with existing (checkip.sh)	Oracle ACS / Xteam Customer		N
4	Wednesday, September 28, 2011	Prereq	Send CheckIP and dbm.out to Customer	Oracle ACS		N
5	Thursday, September 29, 2011	Prereq	Review Exacheck results	Oracle ACS / X Team		N
6	Thursday, September 29, 2011	Prereq	Customer Actions: - Run commands sent by HPUJOL on 9/27 and email results to Oracle - Confirm cell software to stay at 11.2.2.3.2 or go to 11.2.2.3.5 - Run checkIP and send results to Oracle	Customer		N
7	Thursday, September 29, 2011	Prereq	Customer Prerequistes - Permission to work on live system - Ethernet cables connected to customer switches and routed - Permit floor space next to Exadata for efficient installation; if this isn't possible, then extra time may be needed.	Customer		N
			Hardware Installation Phase			
8	Wednesday, October 05, 2011	HW	Sun arrives to unpack equipment, organize, label, lay out equipment …	Oracle HW Team		N
9		HW	Installing nodes, cable management arms and assorted pieces (half to full) - Can be done live, care will be taken	Oracle HW Team		N
10		HW	Install Spine IB Switch, if not already present	Oracle HW Team		N
11		HW	Ensure all IB switches are at latest patch (1.1.3.3-2)	Oracle HW Team		
NA	Sunday, October 09, 2011	HW	New nodes are left in an uplugged state but with network drops attached	Oracle HW Team		
			Patch Existing Nodes			
12	Sunday, October 23, 2011	SW	Existing Cell nodes are patched to 11.2.2.3.5	Oracle ACS / X-Team	3	Y
13		SW	Existing DB nodes are patched to 11.2.2.3.5	Oracle ACS / X-Team	3	Y
14		SW	Existing DB nodes are patched to OEL 5.5	Oracle ACS / X-Team	2	Y
	Monday, October 24, 2011		TRAVEL DAY FOR ACS - NO WORK SCHEDULED			
			Handoff to ACS			
15	Tuesday, October 25, 2011	HW	- Firstboot; unplug network and plug in power one at a time, then plug network back in to the node, etc - change bondib0 to bond0 and verify cell.conf matches - Run Verifcation and Configuration Tests - IB checks - No checks that would disrupt anything…no downtime needed	Oracle HW Team Oracle ACS / Xteam	5	N
16		HW	**Final EIS Checkout** **- plug in ethernet cables to eth1 - eth3** **- verify toplogy**	Oracle/Sun Oracle/ACS / Xteam	1	
17	4PM	HW/SW	Handoff to ACS Complete	Oracle Teams		
			Configure New Nodes			
18		SW	**Run Partial OneCommand against the new nodes** - Requires *_group files for NEW nodes (generated from spreadsheet) - Up to and including "Create Oracle Homes", STOP BEFORE CREATE GRIDDISKS - Recommend not running ValidateIB Step, unless performance hit is approved - Create griddisks manually to match existing layout	Oracle ACS / Xteam	3	N; Optional

To assist in collecting all of the information needed during the planning stages, the following commands were executed by the customer and resulting output files sent to Oracle:

- Current IP addresses defined for all Exadata Storage Servers and database servers using the ifconfig command on all servers.
- The following network files. The files are available from any database server in the rack.
  - /etc/resolv.conf
  - /etc/ntp.conf
  - /etc/network
  - /etc/sysconfig/network-scripts/ifcfg-*
- Current name, offset, and sizes of the grid disks using the following command:

```
list griddisk attributes name,offset,size
```

- Output from the `imagehistory` command from all servers.
- The Exachk report for the current rack. Refer to My Oracle Support note 1070954.1 for information about the Exachk utility.

- Output from the `opatch lsinventory` commend for the grid infrastructure and database home directories.

- Number of Sun Datacenter InfiniBand Switch 36 switches currently installed in the rack using the `ibswitches` command from any database server.

- Firmware version of the Sun Datacenter InfiniBand Switch 36 switches using the `nm2version` command on each switch.

- Output from the CellCLI `list cell detail` command.

- Output from each database server using the following command:

```
cat /proc/meminfo | grep 'HugePages_Total'
```

Implementation

As mentioned previously, the details of the process we used is found in the *Oracle Exadata Database Machine Owner's Guide*, chapter 8. The overall procedure is summarized as:

1. Prepare to extend the Exadata Database Machine
   E.g.: Preparing the nodes, cable bundles, and performing preliminary checks

2. Extend the hardware
   E.g.: Adding the new IB spine switch, adding servers, cabling servers

3. Configure the new hardware
   E.g.: Set up network and configuration files, add griddisks to ASM, add nodes to the cluster, extend monitoring, add DB homes, and add instances to the cluster

4. Validate the extended Exadata Database Machine
   E.g.: Verify IB and cluster configurations, conduct power-off-on test, review parallelism, backup, standby, services, DBFS, EM agent, and group file configurations. Run Exachk.

The new hardware was installed, cabled, and carefully validated by Oracle hardware engineers while the existing system continued operating normally inside the rack. When the hardware engineers finished, they handed the system over to the Oracle Advanced Customer Support (ACS) software engineer who configured and patched the new nodes (again, while the existing nodes operated). The cluster was then expanded onto the new nodes, and the configuration effort progressed to the point where the griddisks introduced in the additional cells were added into ASM diskgroups in the background while the existing databases were online. Meanwhile, the additional instances were added to the existing databases in the cluster while the diskgroups were rebalancing the new griddisks.

Some key points regarding the above steps were:

1. The planning stage is very important, especially developing a solid patching strategy to synchronize the patch levels between the existing and new nodes.
2. The new database nodes will inherit the patch levels installed on the existing nodes (for the grid infrastructure and database homes) so it's best to ensure that existing nodes are patched to the recommended levels *before* expanding the cluster. This can be done in a rolling fashion and ahead of time to reduce the amount of work needed during the extension effort.
3. The existing InfiniBand switches may need to be patched to match the new spine switch that will be added; this too can be done without downtime, and ideally before the extension to reduce the total amount of work.
4. Running a healthcheck report using Exachk (see MOS note 1070954.1) one or two weeks before the extension is a best practice to ensure the system is ready (any identified problems should addressed). Another Exachk after the extension is complete should also be done.
5. Collecting the configuration data listed above was very helpful to validate the existing configuration and plan the new configuration. An important part of this was to verify the InfiniBand IP addresses that are already in use and compare them to the new ones needed for the additional nodes. An IP conflict can result in an outage, so care must be taken.
6. The use of OneCommand to do the initial configuration of the new nodes ensured a quick and thorough process was followed compared to a completely manual approach.
7. It is very important to clean out as much as possible any trace files in the GRID_HOME/rdbms/audit and GRID_HOME/log/diag directories to allow the addNode.sh script to finish quickly.
8. The new database nodes were used to add the new griddisks into the ASM diskgroups and perform the rebalance. This avoided impacting the existing database nodes.
9. Ensure DB_RECOVERY_FILE_DEST_SIZE is large enough to handle the new redo log files and other files that will be added there when new instances are added.

In the end, the customer's Exadata extension activities proceeded successfully as planned, with no downtime to their existing system.

Summary

The customerr was able to successfully expand its Exadata capacity without downtime to its mission-critical applications by utilizing the high availability features built into Exadata's architecture and Oracle's ASM and Clusterware infrastructure. The additional Exadata nodes provided them with the additional capacity needed to grow their sales applications, such as:

- Development teams needed more environments due to multiple projects that couldn't share environments across the board in Development, Test, and Production. Also, interfaces with other applications increased and those environments needed testing for these interfaces. All of this meant that additional storage capacity was needed in the short-term and possibly more nodes for the long-term.
- The production Exadata rack also needed additional testing environments due to project complexity and various schedules; since the QA instances were hosted on the production Exadata system, additional environments were needed there too (mostly for performance testing). New nodes on the production rack were dedicated to QA while existing nodes were left in production. I/O bandwidth from the cells was sufficient to enable both environments to run simultaneously with the same set of cells without starving the production environment for I/O.
- After the Exadata extension, we saw a 20% improvement in performance from the existing production nodes without any effort, most likely due to the I/O bandwidth introduced by the additional cells. Additional performance gains were obtained by changing parallel execution configuration settings.

In the end, the customer was able to take their two quarter rack systems all the way to full racks without downtime. In the future, additional growth can easily come from adding half or full racks to their existing racks.

Appendix

An understanding of the following technology white papers and acronyms will provide the reader of this paper with a basic technical foundation.

Technical White Papers

Oracle Real Application Clusters (RAC) 11g Release 2:
http://www.oracle.com/technetwork/database/clustering/overview/twp-rac11gr2-134105.pdf

Deploying Oracle MAA with Exadata Database Machine:
http://www.oracle.com/technetwork/database/features/availability/exadata-maa-131903.pdf

Oracle Exadata Database Machine:
http://www.oracle.com/technetwork/database/exadata/exadata-technical-whitepaper-134575.pdf

Acronyms

ACS = Advanced Customer Services

ASM = Automatic Storage Management

CRS = Cluster Ready Services

DBFS = Database File System

DBM = Database Machine

HCC = Hybrid Columnar Compression

IB = Infiniband

MAA = Maximum Availability Architecture

RAC = Real Application Clusters

QA = Quality assurance

Oracle Exadata Database Machine MAA Case Study –Leading Biotech Firm
July 2011
Author: Oracle High Availability Product Management and MAA X-Team Members

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com

Oracle is committed to developing practices and products that help protect the environment

Copyright © 2011, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd. 1010

Hardware and Software, Engineered to Work Together