Datrium Technical Report
DVX Solution for Oracle RAC on VMware vSphere

Abstract

This technical report presents an effective solution to deploy a virtualized Oracle RAC solution on Datrium DVX and VMware vSphere. New features in DVX 4.0 provide support for clustering Oracle VMs with RAC technology.

Date: April 1, 2018

Report: TR-2018-04-01

Table of Contents

Table of Contents (continued)

List of Figures

1 Executive Summary

The Datrium DVX system provides an excellent platform for data center virtualization with the right combination of primary storage, scale-out backup and cloud DR all in one solution. Customers looking to virtualize Oracle and provide additional application resiliency with Real Application Cluster (RAC) configurations can now build a solution effectively on the Datrium DVX platform on VMware vSphere.

When considering simplicity of management, performance of the applications and infrastructure and protection of the data produced, Datrium DVX provides an ideal choice for today's data center modernization efforts. New capabilities in the DVX 4.0 release not only address the Oracle RAC solutions but also provide a cost-effective path to cloud as a data protection / backup method.

2 Recommendations Overview

This Technical Report covers several aspects of deploying Oracle RAC on Datrium DVX. This section highlights the key points and practices covered in more detail in this document.

2.1 Practice 1

- Ensure that there are appropriate cluster resources to service the planned workloads. Over commitment of CPU resources is acceptable under most circumstances, however never overcommit memory resources for vSphere clusters running Oracle Database workloads. Any capacity issues should be easily resolvable with the Datrium solution in that the compute and storage nodes are separated. Simply add the resources that are needed.

2.2 Practice 2

- Follow established virtualization best practices. Verify the VMware configuration for all cluster settings, hardware redundancy, VM settings, and Business Continuity/Disaster Recovery requirements.

2.3 Practice 3

- Follow established Oracle RAC best practices when configuring the solution. Ensure appropriate hardware redundancy, N+1 nodes, Linux configuration, and Oracle software configuration. Running Oracle RAC on the VMware virtualization platform with Datrium DVX allows for simple transfer of existing Oracle knowledge and skills.

3 Introduction

The Datrium DVX system provides an ideal virtualization platform for building a modern, state of the practice data center solution for database application services. Starting as small as a single virtualization host with a single database and scaling and clustering the instances as needed to meet the demands of business growth and availability is easily accomplished with the Open Convergence capabilities of the Datrium solution.

In this paper, we examine one such configuration for an Oracle RAC solution that delivers:

- Simpler administration
- Data center modernization technologies
- Secure and protected data

Whenever considering new technologies, systems or approaches, connecting the right resources is the ideal way for vendors and customers to quickly assess the value of a particular solution to meet their needs. This section helps put this report in context with the appropriate staff and business objectives.

3.1 Audience

This Technical Report is intended for solution designers, system architects, database administrators or systems administrators that are looking to cluster virtualized Oracle databases on VMware vSphere with Oracle RAC. We assume a working knowledge of Datrium DVX, VMware vSphere, Linux and Oracle RAC. For additional information about administering, configuring and managing Oracle RAC on DVX please refer to the appropriate manufacturer's documentation.

3.2 Purpose and Assumptions

The purpose of this Technical Report is to cover enough details about the feature, functionality and configuration to better understand deploying Oracle RAC virtual machines on a Datrium DVX system. It is neither a beginner's guide nor is it intended as a how-to guide for setting up the solution. It is also not intended to cover all possible configuration solutions.

In particular, considerations around cost, licensing and performance are not addressed in this report and can be reviewed with your local providers of the particular components of the solution outlined in this Technical Report.

We also assume that you are at least familiar with running Oracle RAC and more likely already have Oracle RAC workloads in a physical environment that you are looking to virtualize.

4 Solution Overview

This Technical Report covers the new capabilities of Datrium DVX Open Converged Architecture for supporting virtualized Oracle RAC solutions on VMware vSphere and Datrium DVX. In this particular configuration we are supporting up to four Oracle RAC instances, ideally with each instance on a dedicated separate host (Compute Node) on a single DVX Data Node. The Data Node required to support Oracle RAC on DVX is the F12X12 with Flash E2E model. The system explored in this Technical Report looks like the following figure.

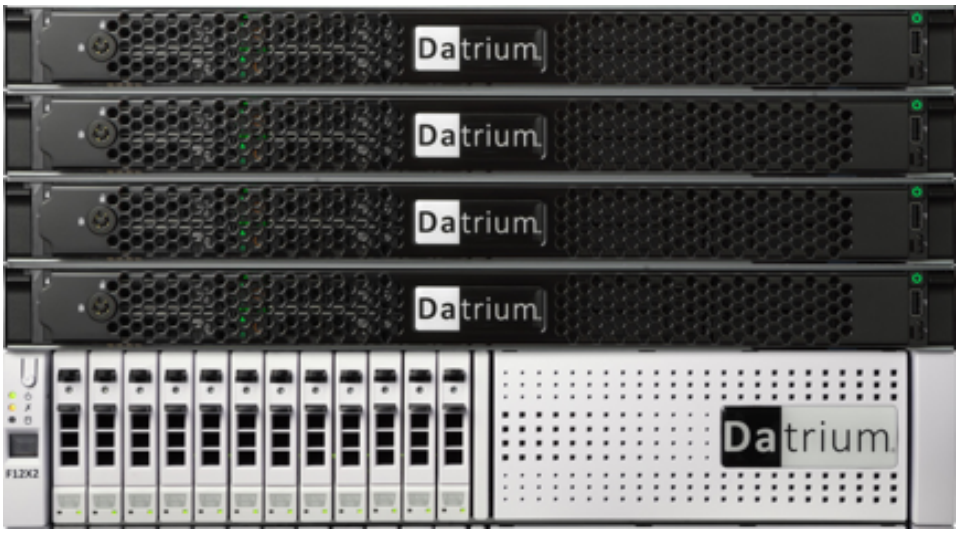

Figure 1. Datrium DVX Solution for Oracle RAC

4.1 Terminology

The solution discussed in this paper covers an Oracle RAC application framework running on VMware vSphere virtualization environment built upon a Datrium DVX Open Converged Infrastructure platform. There are a couple of terms common across these layers worth calling out that, depending on context, can mean different things to the audience. For this paper we will be using the following terminology to discuss the solution in more detail:

- Oracle RAC Instance – this is an instance of the virtualized Oracle database host running in a virtual machine (VM) on the physical ESX host which is also referred to as a Datrium Compute Node
- Host – this refers to the physical server running the vSphere hypervisor and the Datrium hyperdriver software supporting the virtualization solution for the Oracle RAC Instances
- Disk – this will refer to the logical device presented to the Oracle RAC Instance which in turn will be a vDisk from the VMware virtualization layer residing on the Datrium DVX Datastore
- vDisk – this is the virtual disk (VMDK) presented from the DVX Datastore through the

VMware vSphere storage layer to the guest Oracle RAC Instance VM

- NIC – this is a hardware network device connecting the physical server (Compute Node) to other physical servers or to the Data Node(s) – NICs are connected to physical switches and are either 1GbE (typical for management) or 10G (recommended for data, cluster and application)
- vNIC – this is a virtualized network device that is assigned to and possibly shares a physical NIC with other vNICs – it will be presented to the VM as a separate network interface for use as needed

4.2 Datrium DVX

With Split Provisioning and Open Convergence capabilities, customers can start with the right size of systems and then grow to a scalable solution that meets needs as the business requirements increase. Independent scaling for either compute or capacity gives you the freedom to start with just the right amount of infrastructure when beginning a project and adding performance or capacity to support growth. DVX allows for seamless addition of additional compute nodes for performance or data nodes for capacity to the solution. You can begin from the minimum supported two nodes (one Data Node and one Compute Node) and expand from there to Petabyte Scale configurations as needed. Compute nodes – where the database virtual machines run – can come in a variety of sizes and capabilities that address different organizational needs and budgets. These servers run the DVX software. They can be Datrium supplied Compute Nodes or qualified 3rd Party Compute Nodes providing a wide range of host options for deployment.

The potential scalability of a Datrium DVX system with Split Provisioning is shown in the following figure and covered in more detail here.

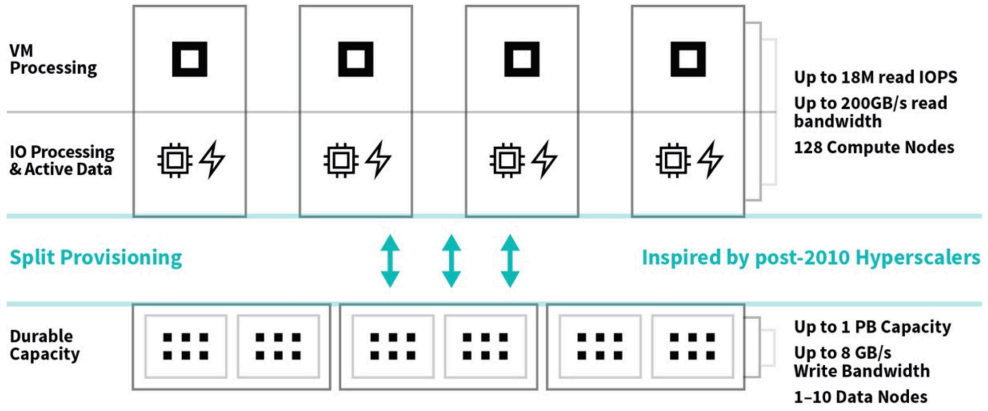

Figure 2. DVX Split Provisioning

4.2.1 Compute Node

The Datrium Compute Node is a physical server running VMware vSphere hypervisor software and installed with Datrium hyperdriver software. Compute Nodes can be Datrium supplied models or compatible 3rd party servers. With a DVX solution, the Compute Nodes have local flash devices (SATA, SAS or NVMe) that are used to hold the active primary data copy for the Oracle database instances running on those physical resources. More information about the Compute Nodes can be found here.

The DVX software runs on the Compute Node and leverages local CPU resources to provide the data services, including erasure coding, data reduction, encryption, and data copy management for the storage system presented to the virtual machines. More information about the software that runs on the Compute Nodes can be found here.

With Open Convergence, the flexibility of which particular server is used for the Oracle solution can be selected to meet the needs of performance, capacity and licensing objectives of the organization.

4.2.2 Data Node

The Datrium Data Node comes preconfigured with fully redundant, hot swappable components. This includes mirrored NVRAM for fast writes, and redundant 10Gb or 25Gb network ports with load balancing and path failover for high speed data traffic. The Data Node for the Oracle RAC solution contains 12 x 1.92TB SSD devices supporting always-on erasure coding, compression, and global dedupe across data received from all connected compute nodes. The capacity of the F12X2 Data Node is 16 TB usable before reduction.

For this configuration, a single Data Node in the solution provides the central shared durable data needs of the Oracle RAC hosts and instances as well as the rest of the VMware environment. For many configurations, this mixed-use approach for the Data Node works well. In a typical DVX environment, the host isolation of active data and IO processing on the Compute Node is central to the DVX architecture. This approach provides scalability through adding hosts when more IO performance is needed. A single Data Node can support up to 32 connected Compute Nodes. Each Oracle RAC configuration is currently defined for up to 4 Compute Nodes.

The Datrium DVX solution is simple to deploy and even simpler to manage. Typical storage management tasks of dealing with LUNs, RAID, or pools are a thing of the past. With Datrium, all storage monitoring and administration is VM-centric and end-to-end – vCenter to vDisk. The DVX Data Node presents a single vSphere datastore to hold all components from the database environment.

More information on the DVX Data Node can be found here.

4.2.3 Data Cloud Foundation

Protecting the critical parts of the solution is achieved by leveraging the Data Cloud Foundation capabilities of the DVX system. The core environment which includes the VMware infrastructure. VMs as well as the database instances and any other components are protected through regular Protection Group policy driven snapshot mechanisms within the Datrium solution. Protection Group policies could be leveraged to protect individual instances or database disks. The policy driven approach can use the native dynamic name matching methods to pick up the evolving set of VMs or vDisks within the DVX system.

We also enabled Datrium's unique Blanket Encryption to protect all database and infrastructure data active on the hosts and at rest on the Data Node. This capability uses AES-XTS-256 end-to-end so ware encryption without any performance hit to your application.

More information on the Data Cloud Foundation details can be found here.

4.3 VMware vSphere

This solution leverages VMware vSphere as the virtualization platform for supporting Oracle RAC on Datrium DVX. Each physical host is installed with the desired version of ESXi, minimum is 5.5U2. It is also useful to have VMware vCenter installed into the environment to help manage the physical hosts – up to 4 – in the RAC cluster. vCenter will also provide a single point of manageability and monitoring for Datrium DVX using the available vCenter plugin.

For supporting the Datrium DVX Compute Node to Data Node connectivity, it is recommended to have a separate 10G network for the data traffic. This would be in addition to networks to support the Oracle application and RAC environment as discussed below.

4.4 Oracle RAC

Oracle Real Application Clusters (RAC) is a database clustering option provided by Oracle that utilizes multiple database instances to provide HA and other services for the application layer.

Two important definitions concerning RAC are:

1. A database is a collection of files.
2. An instance is a memory structure and a set of processes that access those files.

In a RAC database, there is only one copy of the database files themselves, and they must reside on shared storage. There are multiple instances running on different nodes within the defined cluster. Each Oracle RAC cluster will have two layers of software. The first is the Grid Infrastructure (GI) software; this consists of the cluster management software, Clusterware Ready Services (CRS), and the volume manager Automatic Storage Manager (ASM). The second software component is the database (DB) software itself.

The installation and management of Oracle RAC databases can be quite complex and any decision to implement it should not be taken lightly. Once implemented, an Oracle RAC system will allow the database to remain available to connected applications during a variety of hardware and operating system failures as well as for most patching. These benefits can be tremendous for applications that have very high availability requirements.

Note that the increased complexity of RAC can, and often does, lead to more complex administration and day to day management considerations when compared to a single-instance database.

4.4.1 Virtualized RAC

There is nothing within a RAC configuration that prevents it from running in VMware virtualized environments. When running in a virtualized environment all of the HA benefits of a RAC database are available, in addition to the virtualization benefits provided by vSphere. These benefits allow for a multitude of hardware failures, operating system patching and database patching, all while maintaining database availability. In addition, major version software upgrades are possible with minimal downtime.

Another major component to a RAC configuration is the required networking. Every Oracle RAC Instance in the RAC cluster must have a minimum of two vNICs. For virtual environments, two physical NICs are usually more than sufficient as all failover is handled at the hypervisor layer. One vNIC will be placed on the "private" network and the other will connect to the "public" network. The private network is typically mapped to an un-routed, high-speed network that provides communication between all nodes in the cluster to help maintain the cluster and is also called the interconnect. The interconnect is also used to share memory blocks between the nodes of the cluster. Whenever possible the interconnect should utilize jumbo frames. The public network is for all other network traffic that includes all database connections and any management traffic for the operating system (ssh, ntp, etc.).

Mapping the vNIC to physical NIC configurations is outside the scope of this paper. For application, cluster and data networks it is recommended to use 10G networking and at least 2 physical NICs for increased availability and throughput.

4.5 Combined Solution

The overall Oracle RAC solution will look similar to the high-level design diagram shown here. Note that redundant network connections are not shown – only logical connection points.

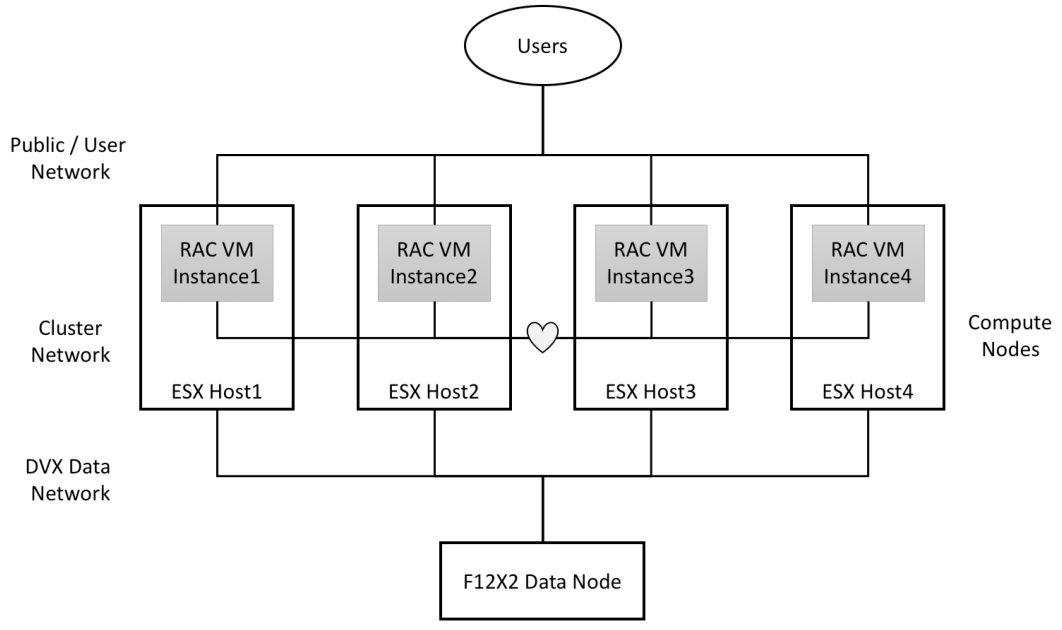

Figure 3. High Level Oracle RAC Design

5 Recommendations and Guidelines

At its base the combination of Datrium DVX and VMware vSphere is an optimal platform for virtualization. As such, all of the best practices for running Oracle RAC on a virtualized platform still apply. The following is a summary of these best practices at each of the layers that have been developed over the years. These are a combination of recommendations from Oracle, Red Hat, and VMware.

The most important of all of the guidelines is to adhere to the first commandment of IT and to follow the Keep It Simple Stupid (KISS) principle. Never make the configuration any more complicated than you absolutely have to. The following recommendations are based on over a decade of House of Brick experience virtualizing Oracle RAC on VMware.

5.1 vSphere

There are many layers at the vSphere level that all have configuration items that will need to be addressed. These configuration items are for both manageability and for performance.

5.1.1 VM Settings

The settings for the VM itself can have a significant impact on the performance of the VM. The primary concerns here are to ensure that the vSphere scheduler can work as efficiently as possible and to help the manageability of the environment.

CPU:

- Configure one core per socket.
- No Oracle Database VM should have fewer than two vCPUs.

Memory:

- Whenever possible configure 100% memory reservations.
- Do not overcommit the physical host's memory.

Network:

- Configure the vNICs to utilize the VMXNET drivers.
- There is no need to use multiple vNICs for the same network, let vSphere handle the underlying physical redundancy.

Storage:

- Set the SCSI controllers to the type Paravirtual.
- Utilize all four SCSI controllers available to the VM. Allocate the VMDKs across the controllers as discussed later in section on ASM and VMDK Configuration.
- Use multiple VMDKs for each ASM diskgroup. Exception: OCR diskgroup only requires one VMDK.
- All VMDKs that are for shared storage are required to be thick provisioned eager zeroed.
- All VMDKs used for the Oracle ASM diskgroups should be set to independent persistent.
- All VMDKs used for the Oracle ASM diskgroups should be set to multi-writer.

General:

- Assign only the resources required to service the workload, do not over provision.
- Remove any virtual hardware that is not needed, for example most VMs should have absolutely no need for a floppy drive or an optical drive.
- Keep the VMware hardware version to the newest possible.
- Maintain VMware Tools at the newest version possible. For Linux VMs open-vm-tools can be utilized.

5.1.2 Cluster Settings

The main point of the cluster settings is to make sure that everything functions as desired. The primary things of note here are:

- Admission control, while this is not always configured, it can be implemented to ensure that resources are not overcommitted beyond a set limit if at all.

- Over commitment, while over committing resources is a fundamental principle for vSphere, it is important to ensure that the cluster has an appropriate level of resources available to it. For clusters hosting Oracle Databases it is of utmost importance to never overcommit on memory.
- Enhanced vMotion Control, if different CPUs are being used between the nodes of the vSphere cluster an appropriate EVC setting will make sure that vMotion will still function.
- Distributed Resource Scheduler, ensure that if two or more VMs that are on the same host become busy that they will be separated. Do use caution that this is not set too "twitchy."
- Resource pools can be used to help manage the resources VMs are allowed to use.
- Affinity and Anti-Affinity rules can be utilized to ensure that two nodes of the RAC cluster do not wind up running on the same vSphere physical host.

5.1.3 Storage

There are many ways to present shared storage to the members of a RAC cluster when using vSphere. The three primary ways are:

- VMDK – this is the preferred method and the one used in this solution.
- IP-based storage – IP-based storage either from an NFS share or from iSCSI can be presented directly to the guest operating system.
- Raw Device Mapping (RDM) – The use of RDM is not recommended for the shared storage requirement of RAC clusters.

For the Datrium DVX solution the use of VMDKs is the best practice. With this approach, there are two things to keep in mind when designing and deploying any application. First, the DVX datastore provides a single, sharable storage target for all of the VMware storage selection options. In other words, place all VM storage on the DVX datastore. Secondly, the host workload is run primarily from local flash on the ESX host and secured on the storage provided by the Data Node. Working from these two assumptions, we have the following notes:

- Make sure there is sufficient local flash (SSD) capacity on the host to hold the entire database instance(s) assigned to that physical host.
- Make sure there is sufficient backing capacity on the Data Nodes to hold all databases in use.
- Use actual DVX data reduction measures from actual production sample sets to aid in any sizing efforts that will leverage the always-on data services. For initial considerations, ~2x capacity savings through compression and deduplication have been observed within our database customers using DVX today.
- For Oracle RAC deployments, the Data Node needs to be the F12X2 – all flash – model to ensure optimal recovery conditions in the event of instance/host failover within the RAC cluster.
- The DVX VAAI VIB should be installed on all physical vSphere hosts in the RAC cluster. For more information about the specific Datrium DVX settings, refer to the DVX 4.0 Systems Management Guides (or later) available on the Datrium Support Site.

5.2 Database Backups

The primary native utility for backing up an Oracle database is Recovery Manager (RMAN). There are many things to consider when protecting the Oracle database and environment using native Oracle tools as well as storage provider alternatives.

- The RMAN utility is a part of the database software installation. There is no special installation process.
- Most DBAs are very familiar with RMAN and will prefer to use it for all backup and restore functionality. Most DBAs will be reluctant to give up control of this functionality. They will want to be comfortable with all backup and restore processes.
- If for some reason Oracle Support is contacted during a restore process, they are most familiar with RMAN. Any potential resolution will be much faster if the use of RMAN is maintained. Any other backup and restore procedure will get "best efforts" support from Oracle.
- RMAN was written to be very flexible and to provide for quite a few different scenarios. It is possible to write backups (knows as sets and pieces) to disk, ASM diskgroup and/or to tape. Datrium DVX can provide a backup destination that is also replicated offsite for even more robust protection.
- There are several different file types used in an Oracle database; these include: database data files, control files, initialization parameter file, online redo log files, and archive log files. RMAN can be used to backup all of these file types, except for online redo log files, which are technically copied to archive log during the backup process.
- RMAN has all of the backup functionality that you would expect; full, incremental and differential backups are all available.
- Technically, full backups cannot have any incremental backups applied to them during the restore process. Incremental backups are based on levels and a level 0 is effectively a full backup.
- It is common practice to take weekly full backups (level 0) and then daily incremental (level 1).
- It is a common practice to write all backups to disk. This disk can be an NFS mount. Once the backups are on disk, the backup files can be copied off to another media or location like cloud or tape.
- All information concerning the details of the backup is kept in the control files. Optionally, an RMAN catalog, a database that maintains backup information, can be utilized.
- Cloning activity can utilize RMAN backups as a foundation.

The Datrium DVX system also provides built-in data protection capabilities through the Protection Group and Replication features of the product. These methods can also be effectively deployed with no impact to the running applications and provide an alternate and sometimes more readily available aspect to protecting the Oracle RAC environment. More details about the Datrium Data Cloud Foundation capabilities can be found here or through your local Datrium representative.

- The use of Datrium DVX Protection Group based storage level snapshots could simplify the backup and restore process. They are also incredibly useful with any cloning to non-production databases. We recommend the DBA and site admins become familiar and comfortable with the Datrium DVX built in data protection functionality provided.

5.3 Linux

The first three things to decide on when installing Linux are distribution, version, and installation type. For the first two make sure that you check the Oracle certification for the version of Oracle Database that you will be running. For the distribution simply choose whichever certified distribution that you prefer. Once the distribution is decided upon you will likely have a few different versions to choose from, take the newest one that you are comfortable with. As for the installation type it is recommended to choose a minimal installation and then to manually install any additional packages.

The next step is to work through the Oracle installation prerequisites. If the Oracle Linux distribution was selected there is an optional RPM that can be installed that will configure most of the prerequisites automatically. For other distributions you will need to work through the configuration steps manually. For the most part the prerequisites will be sufficient for most Oracle Database installations and should be followed unless from experience you know that the database requires something different.

5.3.1 Kernel Parameters

There are a few kernel parameters that should be adjusted after the software installation is completed. These parameters are summarized in the following table.

Parameter	Description	Value
kernel.shmmax	The maximum size of any shared memory chunk.	Dependent on the aggregate of the SGA sizes on the server. Typically, 50% of the size of the largest SGA on the VM.
kernel.shmall	This value determines the maximum amount of memory that all shared memory can take. The actual setting is derived Shareable memory = shmall * pagesize. Pagesize = getconf PAGE_SIZE	Dependent on memory of the VM. This is in pages not bytes. Typically, between 70-80% of the available memory, and should never be greater than the memory assigned to the VM.

vm.nr_hugepages	Defines the amount of memory to reserve for huge pages. Value is derived based on the huge page size. Value can be obtained from /proc/meminfo.	Dependent on the aggregate of the SGA sizes on the server.
vm.swappiness	Defines how aggressively memory pages are swapped to disk.	1
vm.dirty_background_ratio	Defines the percentage of memory that can become dirty before a background flushing of the pages to disk.	3
vm.dirty_ratio	Defines the percentage of memory that can be occupied by dirty pages before a forced flush. If you set this to a low value, the kernel will flush small writes to the disk more often. Higher values allow the small writes to stack up in memory. They'll go to the disk in bigger chunks.	80
vm.dirty_expire_centisecs	How old dirty data should be before the kernel considers it old enough to be written to disk.	500
vm.dirty_writeback_centisecs	How often the kernel should check if there is "dirty" (changed) data to write out to disk (in centiseconds).	100

5.3.2 IO Scheduler

The IO scheduler can affect the IO performance of the VM. It is best to set all disks to use the deadline scheduler. Depending on OS distribution and version the default IO scheduler may already be set to deadline. To check which scheduler is being used run the following command:

```
# cat /sys/block/<disk>/queue/scheduler
```

For example, for SDA:

```
# cat /sys/block/sda/queue/scheduler

noop anticipatory [deadline] cfq
```

The scheduler that is framed by the brackets is the scheduler in effect for the disk that has been queried. It is possible to set different schedulers for different disks. The following command will change the scheduler for the selected disk immediately.

```
# echo deadline > /sys/block/sda/queue/scheduler
```

The setting will not survive a reboot. To permanently change the scheduler for a disk either set the *elevator* parameter in the grub.conf file or use the Linux utility *tuned*.

5.3.3 Huge Pages

Using large memory pages is a technique that allows the operating system to allocate and manage memory using a larger-than-standard page size for shared memory segments. The default page size for 64-bit Linux systems is 4KB (4096 bytes). By contrast, 64-bit Linux huge pages are 2MB. Huge pages are set with the kernel parameter vm.nr_hugepages. Utilizing huge pages can significantly improve performance for Oracle Database VMs and the time it takes to manage configuring it is well worth it.

Oracle Database uses two different memory management methods that attempt to simplify the configuration of memory for the database. These are Automatic Shared Memory Management (ASMM) introduced in 10g, and Automatic Memory Management (AMM) introduced in 11g. Both memory management methods are available in 11g and 12c versions. When using AMM the non-shared memory for the PGA is included, due to this AMM is not compatible with huge pages. It is recommended to utilize ASMM.

5.3.4 Transparent Huge Pages

Transparent huge pages (THP) are an attempt to have the operating system manage huge pages automatically for the shared memory segments. The use of THP with Oracle Database and can cause significant performance issues, therefore it is recommended to disable THP. This can be done with either a grub.conf configuration or through the Linux utility tuned.

5.4 Oracle

At the Oracle software layer there really is no difference between a physical and a virtual environment. You will still follow the software prerequisites, adhere to optimum flexible architecture principals, and manage the databases as you do today. Due to the hardware redundancy being handled at the vSphere layer managing Oracle RAC in virtualized environments in many ways is easier than managing it in a physical world.

5.4.1 ASM and VMDK Configuration

Of utmost importance is the shared storage requirement. For the storage layer it is recommended to use Oracle Automatic Storage Management (ASM) as the volume manager. There is nothing within the virtualization layer that will prevent the use of ASM, nor will using ASM unnecessarily add to

the complexity of the configuration. As the VMDKs are attached to the VMs the operating system will "see" the disks as locally attached storage. This configuration is very basic and works well with ASM.

As mentioned it is best to ensure that multiple PVSCSI adapters are configured for the VM for the VMDKs. The exact storage layout that is used will be dependent on the number of databases hosted on the RAC cluster, the size of the databases, and other management concerns.

The following is an example layout of the OS volumes and ASM diskgroups by controller. Ultimately the needs of the database will determine the actual number of VMDKs, their size, and their layout.

- Controller 0 – OS disk(s), Oracle software disk(s), OCR DG. Optionally REDO, AL DGs.
- Controller 1 – DATA DG
- Controller 2 – DATA DG
- Controller 3 – FRA DG. Optionally REDO, AL DGs.

A frequent question is how many disks to use for any specific diskgroup. The official recommendation from Oracle is to use 2-4 disks per path to storage, however this recommendation is for a physical environment and it does not make much sense in a virtual world. When virtualizing Oracle RAC, we typically look at the size of the diskgroup first. It is a best practice and as of 12c a requirement that all disks in a diskgroup be of the same size. With this requirement it makes more sense to decide how much you will want to grow a diskgroup by, then determine the initial size of the diskgroup, and then to use as many disks as necessary for the grow by size to get to the initial size. For example, for a DATA diskgroup with a grow by size of 256 GB and an initial size of 1 TB then 4 VMDKs for the diskgroup would be necessary.

One of the few tunables for an ASM diskgroup is the allocation unit (AU) size. By default, ASM will stripe across all disks within a diskgroup; and the AU is simply the size of that stripe. The general recommendations from Oracle concerning AU should be followed unless there is specific performance data from the database to suggest otherwise. The summary of Oracle recommendations is as follows:

- OCR diskgroup – 1 MB. This is a very small diskgroup and utilizes fairly small I/O patterns.
- DATA diskgroup – 4 MB. For any diskgroup that host data files, the default value of 4 MB should be used. Some data warehouses with very large sequential reads and writes may benefit from a larger AU. Test with the specific database to verify.
- REDO – 1 MB. If redo is placed into its own separate diskgroup, a smaller AU should be used.
- FRA – 4-32 MB. If the FRA is utilized for very large backup files with little to no other smaller I/O access, larger AU sizes can be utilized. If the FRA is a multi-purpose diskgroup (e.g., backups, redo, archivelog dest, etc.) then the default value of 4 MB should be maintained.

When creating the diskgroups the setting for external redundancy should be used for all diskgroups. The Datrium DVX System is providing all of the necessary data protection; there is no need to waste storage at the ASM layer by providing additional mirroring.

5.4.2 Multi-Writer

In order to share a single VMDK amongst multiple VMs the multi-writer flag must be set. Technically this disables simultaneous write protection so that multiple VMs can write to the VMDK at the same time. For more information refer to VMware Knowledge Base article 1034165.

In addition to providing information on the setting the article also provides instruction on how to set the multi-writer flag. This is most easily done with vSphere version 6 or newer. In older versions of vSphere editing advanced VM settings through the GUI is not possible while the VM is running. Starting with vSphere 6 the option for setting the multi-writer flag for a running VM is allowed.

When setting the multi-writer flag the VMDK must also be configured as thick provisioned eager zeroed. Additionally, it is recommended that all VMDKs that will be a part of any ASM diskgroup be set as independent persistent so that snapshots do not affect the VMDK.

There are a few restrictions with VMDKs that have the multi-writer flag set. The most common tasks that cannot be performed are VM Snapshots and Storage vMotion. Most backup software that takes a backup of the VM as a whole will take a VM Snapshot as a part of the process. Storage level snapshots will likely still function as expected. Please refer to the Knowledge Base article as noted above for full details.

Starting in vSphere 6, the configuration of shared disks can now be performed with the GUI while the VM is running; in previous releases the only way to add the VMDKs to a running VM was from the vSphere CLI.

For more information, please refer to the following:

- http://houseofbrick.com/vsphere-6-and-oracle-rac/
- https://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=1034165

5.4.3 Device Persistence

Regardless of the storage presentation method when using ASM it is necessary to label the disks so that they are properly recognized at boot time. This is done with either the Oracle utility ASMLib or the Linux utility UDEV. Both methods are supported and will function. It is truly preference as to which to implement.

To use a disk with ASM the disk must first be partitioned. The disk should be partitioned before either ASMLib or UDEV is used to label it. See Appendix A for more detail on ASMLIB and UDEV setup and use.

5.5 Testing

As part of the Datrium specific validation and testing of the solution we used combinations of the SLOB (Silly Little Oracle Benchmark) tool for basic IO loads as well as Swingbench for more of the failure mode tests. In both situations, performance of the Oracle database environment was not impacted by the clustering approach using RAC and both host (vSphere) and instance (VM) failure scenarios operated as expected.

For this configuration support a 4-node (physical host) and 4-VM (Oracle RAC Instance) environment was constructed and tested and used as the reference solution for this document.

6 Conclusion

The latest 4.0 version of Datrium DVX now supports the features needed to allow organizations to fully virtualize their Oracle database solutions including RAC implementations. IT and DB administrators can take full advantage of the simplicity of management and performance gains of the DVX solution and deploy production Oracle databases – single instance or clustered – and leverage the advances brought by improving server horsepower, local flash storage access and data protection and management to meet the needs of today's modern data centers.

Appendix A - ASMLIB and UDEV Setup

ASMLib

The utility, ASMLib, can be used to label and set permissions of the disk(s) at boot time. For Oracle Linux systems, ASMLib is included as a part of the operating system and will be easier to use than UDEV.

Download

For Red Hat systems, ASMLib will have to be installed manually. First, download the appropriate packages from Oracle and/or Red Hat. The link below offers additional information.

Reference: http://www.oracle.com/technetwork/server-storage/linux/asmlib/index-101839.html

Install

Use the utility RPM to install the downloaded packages. The package names in this example were for a RHEL 7.1 installation. Package names will differ based on ASMLib and Linux version.

```
# rpm -Uvh kmod-oracleasm-2.0.8-8.el7.x86_64.rpm
# rpm -Uvh oracleasm-support-2.1.8-3.el7.x86_64.rpm
# rpm -Uvh oracleasmlib-2.0.8-2.el7.x86_64.rpm
```

Configure

For both Red Hat and Oracle Linux, configure ASMLib for use with the command oracleasm.

```
# oracleasm configure -i
Configuring the Oracle ASM library driver.

This will configure the on-boot properties of the Oracle ASM
library

driver. The following questions will determine whether the
driver is

loaded on boot and what permissions it will have. The current
values

will be shown in brackets ('[]'). Hitting without typing an

answer will keep that current value. Ctrl-C will abort.
```

```
Default user to own the driver interface []: oracle
Default group to own the driver interface []: dba
Start Oracle ASM library driver on boot (y/n) [n]: y
Scan for Oracle ASM disks on boot (y/n) [y]: y
Writing Oracle ASM library driver configuration: done
```

```
# oracleasm init
Creating /dev/oracleasm mount point: /dev/oracleasm
Loading module "oracleasm": oracleasm
Mounting ASMlib driver filesystem: /dev/oracleasm
```

Add Disks

Label the disks for use.

```
# oracleasm createdisk DATA01 /dev/sdb1
Writing disk header: done
Instantiating disk: done
```

Oracle ASMLib Alternative - ASMFD

Starting with Oracle Grid Infrastructure 12.1.0.2, ASMLib is deprecated in favor of a new utility for labeling the disks for device persistence. The new utility is called ASM Filter Driver (ASMFD). ASMFD is very similar to ASMLib and is installed with the Grid Infrastructure home. In version 12.1.0.2, there is a chicken and egg issue when utilizing ASMFD. The installation and configuration process for the GI home requires a diskgroup to be created; however, ASMFD cannot be configured until after the installation. For this reason, a software-only installation will have to be performed. Follow the relevant Oracle installation documentation for installing a Grid Infrastructure home and to configure ASMFD. Under most circumstances ASMLib should be more than sufficient.

UDEV

The utility UDEV is a dynamic device manager for Linux systems. It allows for devices to be identified and labeled according to a set of rules. For Oracle Database systems it can be utilized to identify, label, and set ownership and permissions for the disk devices. UDEV can be used as an alternative to ASMLib.

This example does assume that the disks have been partitioned. All nodes in the cluster will need the UDEV rules file and will need to have UDEV started. Once the file is created on the first node, it can be copied to the other nodes in the cluster. First, obtain the ID of the disk(s).

There are minor UDEV syntax changes between RHEL 5, 6, and 7. Syntax for all versions is noted.

RHEL/OL 5.x

```
# scsi_id -g -u -s /block/sdb
36000c2969b01f5516f63233ac569b91e
```

RHEL/OL 6.x/7.x

```
# scsi_id -g -u /dev/sdb
36000c2969b01f5516f63233ac569b91e
```

For RHEL 7, the scsi_id command is not in the root user's $PATH by default. It is located in /usr/lib/udev. The switches used are the same as RHEL 6.

Next, edit (or add) the file /etc/udev/rules.d/99-oracle-asm.rules, and add a line for each of the ASM disks using the uuid from the previous command for the "RESULT=" key. Also, make sure that the NAME, OWNER, GROUP, and MODE parameters are set properly.

RHEL/OL 5.x

```
# cat /etc/udev/rules.d/99-oracle-asm.rules
KERNEL=="sd?1", BUS=="scsi", PROGRAM=="/sbin/scsi_id -g -u
-s /block/$parent", RESULT=="36000c2969b01f5516f63233ac
569b91e", NAME="oracle/ASM_DATA01", OWNER="oracle", GROUP="d
ba", MODE="0660"
```

RHEL/OL 6.x/7.x

```
# cat /etc/udev/rules.d/99-oracle-asm.rules
KERNEL=="sd?1", BUS=="scsi", PROGRAM=="/sbin/scsi_id -g -u /
dev/$parent", RESULT=="36000c2969b01f5516f63233ac569b91e",
NAME="oracle/ASM_DATA01", OWNER="oracle", GROUP="dba",
MODE="0660"
```

Start udev and verify that the new devices are listed in /dev/oracle and are owned by oracle:dba. In this example the dba group is the primary group for the Oracle user. Your implementation may use a different group name.

The device directory does not exist.

```
# ls -l /dev/oracle
ls: /dev/oracle: No such file or directory
```

Start udev.

```
# start_udev
Starting udev:                                [ OK ]
```

Verify that the devices were created and that they have proper permissions.

```
# ls -l /dev/oracle
total 0
brw-rw---- 1 oracle dba 8, 17 Jan 21 12:54 ASM_DATA01
```

About the Authors

Mike McLaughlin is the Director, Technical Marketing at Datrium. Prior to Datrium Mike was the Sr. Manager of Technical Marketing at Nimble Storage (HPE). Mike has been involved in VMware based solutions for the past several years working with customer and partners like VMware and Oracle in helping define, test and deploy virtualized database solutions.

Joseph Grant is a Principal Architect for House of Brick Technologies and is involved with many projects for customers around the world at any given time. Joe primarily works as an Oracle DBA with an emphasis on performance, virtualization, operations, and automation. In addition to Oracle software, Joe works extensively with VMware vSphere primarily in the areas of architecture, operations, and performance. Joe is an Oracle Certified Professional, and formerly a VMware Certified Professional.