Oracle VMware Hybrid Cloud High Availability Guide

REFERENCE ARCHITECTURE

Table of contents

Executive Summary

Business Case

Customers deploying Oracle Real Application Clusters (RAC) face a multitude of requirements including stringent SLAs, continued high performance, and application availability.

Deploying Oracle RAC on physical architecture is subjected to challenges similar to those running Oracle non-RAC on physical architecture. These challenges include but are not exclusive to hardware failure due to a failed component, power outage, and complete hardware meltdown.

Providing high availability in these environments presents a significant challenge for business organizations. Hardware issues negate the inherent value proposition of Oracle RAC, which is to provide application-level high availability with sustained infrastructure high availability.

On-Premises with VMware vSphere

With VMware vSphere®, customers have successfully run business-critical, high performance demanding Oracle workloads for many years. VMware vSphere provides high availability natively at the infrastructure level and is completely complementary to the application level high availability that Oracle RAC provides.

With greater numbers of production servers now virtualized, demand for providing highly converged server-based storage is surging. VMware vSAN™ aims to provide highly scalable, available, reliable, and high-performance storage using cost-effective hardware (i.e., direct-attached disks in VMware ESXi™ hosts).

VMware vSphere® Virtual Volumes™ (vVols) is an integration and management framework that virtualizes SAN/NAS arrays, enabling a more efficient operational model that is optimized for virtualized environments and is centered on the application instead of the infrastructure. vVols simplifies the delivery of storage service levels to individual applications by providing finer control of hardware resources and native array-based data services that can be instantiated with virtual machine (VM) granularity.

Extended Oracle RAC provides greater availability than local Oracle RAC. It provides extremely fast recovery from a site failure and enables all servers, in all sites, to actively process transactions as part of a single database cluster. Extended Oracle RAC enables transparent workload sharing, workload balancing, site maintenance without service disruption, and high availability across sites.

VMware vSAN Stretched Cluster enables active/active data centers that are separated by metro distance. Extended Oracle RAC with VMware vSAN Stretched Cluster enables transparent workload sharing between two sites accessing a single database, while providing flexibility to migrate or balance workloads between sites in anticipation of planned events (e.g., hardware maintenance).

VMware vSphere Metro Storage Cluster (vMSC) is a specific storage configuration commonly referred to as a stretched storage cluster or metro storage cluster. These configurations are usually implemented in environments where disaster and downtime avoidance are a key requirement.

Migrating to Public Cloud

Enterprise IT infrastructure and operations organizations are looking for ways to migrate on-premises vSphere-based workloads to the public cloud, consolidate and extend data center capacities, and optimize, simplify and modernize their disaster recovery solutions.

VMware Cloud™ on AWS is an on-demand service that enables customers to run applications across vSphere-based cloud environments with access to a broad range of AWS services. Powered by VMware Cloud Foundation™, this service integrates vSphere, vSAN and VMware NSX® along with VMware vCenter® management, and is optimized to run on dedicated, elastic, bare-metal AWS infrastructure. ESXi hosts in VMware Cloud on AWS reside in an AWS availability zone (AZ) and are protected by vSphere High Availability (HA).

Stretched Clusters for VMware Cloud on AWS is designed to protect against an AWS AZ failure. Applications can span multiple AWS AZs within a VMware Cloud on AWS cluster.

With Stretched clusters for VMware Cloud on AWS, exceptionally demanding applications (i.e., those whose requirements include high SLAs, continued high performance, and application availability) can now be deployed to the cloud while simultaneously delivering high availability across multiple AZs.

Solution Overview

This paper describes the configuration and deployment of traditional Oracle RAC on VMware vSphere and VMware vSAN. The paper discusses configuration and deployment of Extended Oracle RAC on VMware vSAN Stretched Cluster and VMware vSphere Metro Storage Cluster (vMSC).

Key Results

The following highlights validate the capability of the VMware vSphere platform to provide high availability and performance to business-critical Oracle RAC workloads:

- Ability to provide infrastructure-level high availability to traditional Oracle RAC deployments using VMware vSphere, VMware vSAN, and VMware vVols
- Ability to provide site-level high availability along with infrastructure-level high availability to Extended Oracle RAC deployments using VMware vSphere Metro Storage Cluster (vMSC) and VMware vSAN Stretched Cluster

Introduction

Purpose

This reference architecture is focused on the deployment of traditional Oracle RAC on VMware vSphere, vSAN and vVols, as well as the deployment of Extended Oracle RAC using VMware vSphere Metro Storage Cluster (vMSC) and, vSAN Stretched Clusters.

Scope

This reference architecture validates the capability of the VMware vSphere platform to provide high availability and performance to business-critical Oracle RAC workloads

- Ability to provide infrastructure level high availability to traditional Oracle RAC deployments using VMware vSphere, VMware vSAN and VMware vVols
- Ability to provide site level high availability along with infrastructure level high availability to Extended Oracle RAC deployments using VMware vSphere Metro Storage Cluster (vMSC) and VMware vSAN Stretched Cluster

Audience

This reference architecture is intended for Oracle Database administrators, virtualization and storage architects involved in planning, architecting, and administering a traditional or Extended Oracle RAC environment on a VMware software-defined data center (SDDC) platform.

Terminology

This paper includes the following terminology:

Term	Definition
Oracle Single Instance	Oracle Single-Instance database consists of a set of memory structures, background processes, and physical database files, which serves the database users.
Oracle Clusterware	Oracle Clusterware is a portable cluster software that allows clustering of independent servers so that they cooperate as a single system.
Oracle Automatic Storage Management (Oracle ASM)	Oracle ASM is a volume manager and a file system for Oracle database files that support Single-Instance Oracle Database and Oracle RAC configurations.

Oracle ASMLIB and Oracle ASMFD	Oracle ASMLIB maintains permissions and disk labels that are persistent on the storage device, so that the label is available even after an operating system upgrade. Oracle ASMFD helps prevent corruption in Oracle ASM disks and files within the disk group

TABLE 1. KEY TERMS

Technology Overview

Overview

This section provides an overview of the technologies used in this solution:

- VMware vSphere
- VMware vSAN
- VMware vSphere Virtual Volumes (vVols)
- VMware vSphere Metro Storage Cluster (vMSC)
- VMware vSAN Stretched Cluster
- VMware SDDC
- Hybrid Cloud
- VMware Cloud on AWS
- Stretched Clusters for VMware Cloud on AWS
- Raw Device Mapping (RDM) and Virtual Disk (VMDK)
- VMware Virtual Disk Provisioning Policies
- VMware Multi-Writer Attribute for Shared VMDKs
- Shared disks using Raw Device Mapping (RDM)
- Shared disks using VMware vSphere Virtual Volumes (vVols)
- Storage Policy Based Management (SPBM)
- VMware vSAN Storage Policy
- Mware vSphere Virtual Volumes (vVols) Storage Policy
- VMware Distributed Switch and Distributed Port Group
- VMware DRS and Affinity Rules
- Oracle Database 19c
- Oracle Database Architecture
- Oracle Multitenant Architecture
- Oracle Automatic Storage Management
- Oracle ASMLIB and ASMFD
- Linux Device Persistence and udev rules
- Oracle Clusterware
- Oracle Real Application Cluster
- Oracle RAC One Node
- Extended Oracle RAC
- Oracle RAC on VMware vSphere Platform
- Extended Oracle RAC on VMware vSphere Platform

VMware vSphere

VMware vSphere, the industry-leading virtualization and cloud platform, is the efficient and secure platform for hybrid clouds, accelerating digital transformation by delivering simple and efficient management at scale, comprehensive built-in security, a universal application platform, and seamless hybrid cloud experience. The result is a scalable, secure infrastructure that provides enhanced application performance and can be the foundation of any cloud.

vSphere 7.0 is the next-generation infrastructure for next-generation applications. vSphere 7.0 simplifies management at scale, secures both infrastructure and workloads, delivers a universal platform for applications, and provides a seamless hybrid cloud experience. It powers the computing environment for modern applications, artificial intelligence (AI) and machine learning (ML), and business-critical applications.

Notable features of vSphere 7.0 include:

- Simplified lifecycle management, including new tools for simplified upgrades, patching and configurations.
- Intrinsic security, which includes vSphere trust authority.
- Identity federation and application acceleration, including enhancements to DRS and vMotion for large and mission-critical workloads.

Learn more about the *new features of VMware vSphere*.

VMware vSAN

VMware vSAN powers industry-leading hyper-converged infrastructure (HCI) solutions with a vSphere-native, high-performance architecture. vSAN helps organizations evolve their data center without risk, control IT costs, and scale to address tomorrow's business needs.

Responding to industry trends and requirements, vSAN 7.0 simplifies infrastructure management by reducing the number of tools required to manage the server lifecycle. Organizations can now unify block and file storage with vSAN, thus reducing the need for third-party solutions and accelerating file share provisioning. vSAN 7.0 also includes new capabilities, including enhancement of cloud-native applications and support for file-services and vSphere add-on for Kubernetes (formerly known as Project Pacific) through VMware Cloud Foundation.

Seamless integration with the complete VMware SDDC stack, as well as leading hybrid cloud offerings, makes it the most complete platform for VMs—whether running business critical databases, virtual desktops, or next-generation applications.

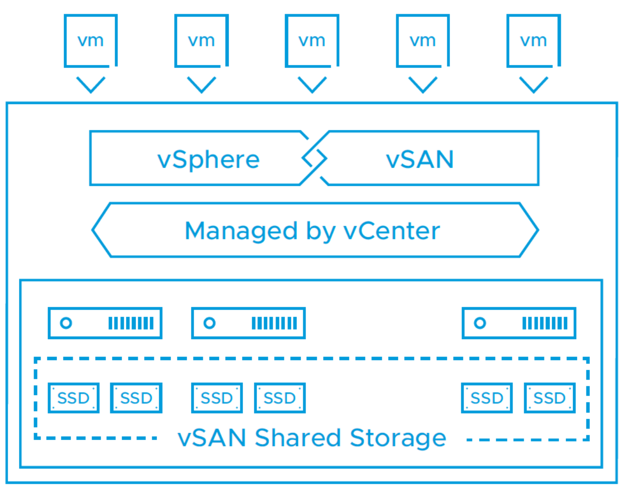


FIGURE 1. VMWARE VSAN HCI

Learn more about *VMware vSAN*.

VMware vSphere Virtual Volumes

vVols is an integration and management framework that virtualizes SAN/NAS arrays, enabling a more efficient operational model that is optimized for virtualized environments and is centered on the application instead of the infrastructure.

vVols simplifies operations through policy-driven automation that enables more agile storage consumption for VMs and dynamic adjustments in real time, when they are needed. It simplifies the delivery of storage service levels to individual applications by providing finer control of hardware resources and native array-based data services that can be instantiated with VM granularity.

With vVols, VMware offers a paradigm in which an individual VM and its disks, rather than a LUN, become a unit of storage management for a storage system. vVols encapsulate virtual disks and other VM files, then natively store the files on the storage system.

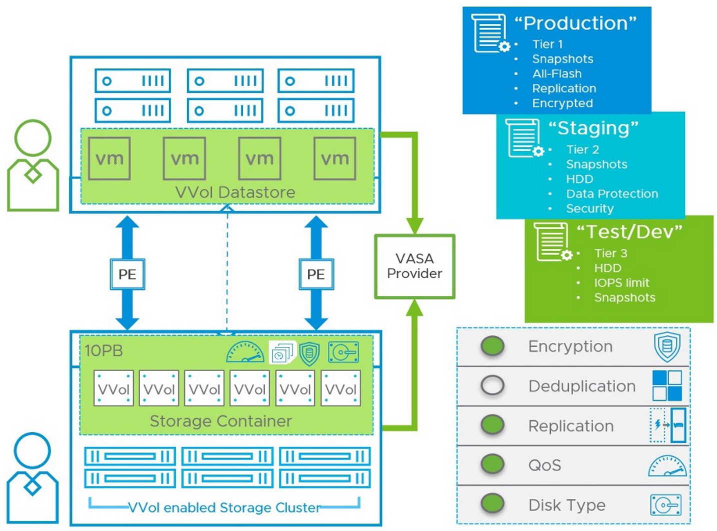


FIGURE 2. VMWARE VSPHERE VIRTUAL VOLUMES

Learn more about *VMware vSphere Virtual Volumes (vVols)*.

Recommendation is to work with the Storage specific vendor about getting correct guidance for implementing VMware vVols as every storage vendor implementation of vVols is implemented differently.

VMware vSphere Metro Storage Cluster

A VMware vSphere Metro Storage Cluster is a specific storage configuration which is commonly referred to as *stretched storage clusters or metro storage clusters*. These configurations are usually implemented in environments where disaster and downtime avoidance are a key requirement.

A vMSC configuration is a specific storage configuration that combines replication with array-based clustering. These solutions are typically deployed in environments where the distance between data centers is limited, often metropolitan or campus environments.

vMSC infrastructures are implemented with a goal of reaping the same benefits that vSphere HA clusters provide to a local site, in a geographically dispersed model with two data centers in different locations. A vMSC infrastructure is essentially a stretched cluster. The architecture is built on the premise of extending what is defined as "local" in terms of network, storage and compute to enable these subsystems to span geographies, presenting a single and common base infrastructure set of resources to the vSphere cluster at both sites. It in essence stretches storage, network and compute between sites.

The primary benefit of a stretched cluster model is that it enables fully active and workload-balanced data centers to be used to their full potential and it allows for an extremely fast recovery in the event of a host or even full site failure. The capability of a stretched cluster to provide this active balancing of resources should always be the primary design and implementation goal. Although often associated with disaster recovery, vMSC infrastructures are not recommended as primary solutions for pure disaster recovery.

vMSC solutions are classified into two distinct categories. These categories are based on a fundamental difference in how hosts access storage. It is important to understand the different types of stretched storage solutions because this influences design considerations.

The following two main categories are as described on the VMware Hardware Compatibility List:

- Uniform host access configuration: ESXi hosts from both sites are all connected to a storage node in the storage cluster across all sites. Paths presented to ESXi hosts are stretched across a distance.
- Nonuniform host access configuration: ESXi hosts at each site are connected only to storage node(s) at the same site. Paths presented to ESXi hosts from storage nodes are limited to the local site.

Learn more about *VMware vSphere Metro Storage Cluster*.

VMware vSAN Stretched Cluster

Stretched clusters extend the VMware vSAN cluster from a single data site to two sites for a higher level of availability and inter-site load balancing. Stretched clusters are typically deployed in environments where the distance between data centers is limited, such as metropolitan or campus environments.

Stretched clusters can be used to manage planned maintenance and avoid disaster scenarios, as maintenance or loss of one site does not affect the overall operation of the cluster. In a stretched cluster configuration, both data sites are active sites. If either site fails, vSAN uses the storage on the other site. vSphere HA restarts any VM that must be restarted on the remaining active site.

One site must be designated as the preferred site. The other site becomes a secondary or nonpreferred site. The system uses the preferred site only in cases where there is a loss of network connection between the two active sites. The site designated as preferred is typically the one that remains in operation, unless the preferred site is resyncing or has another issue. The site that leads to maximum data availability is the one that remains in operation.

Each stretched cluster consists of two data sites and one witness host. The witness host resides at a third site and contains the witness components of VM objects. It contains only metadata and does not participate in storage operations.

The witness host serves as a tiebreaker when a decision must be made regarding availability of datastore components, occurring when the network connection between the two sites is lost. In this case, the witness host typically forms a vSAN cluster with the preferred site. However, if the preferred site becomes isolated from the secondary site and the witness, the witness host forms a cluster using the secondary site. When the preferred site is online again, data is resynchronized to ensure both sites have the latest copies of all data.

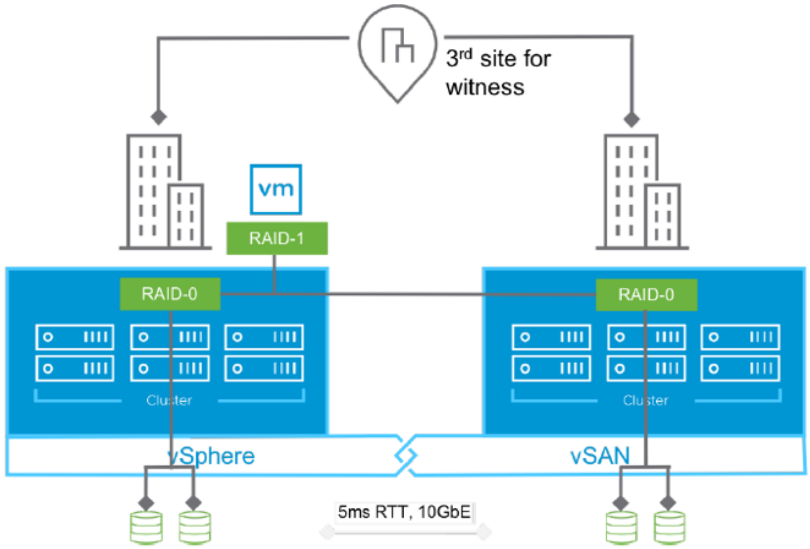


FIGURE 3. VMWARE VSAN STRETCHED CLUSTER

Learn more about *VMware vSAN Stretched Cluster*.

VMware SDDC

The mobile cloud era is changing line-of-business (LOB) expectations of IT. For IT organizations to securely deliver the anticipated improvements in service quality and speed, an SDDC approach is required.

The VMware approach to the SDDC delivers a unified platform that supports any application and provides flexible control. The VMware architecture for the SDDC empowers companies to run hybrid clouds and leverage unique capabilities to deliver key outcomes that enable efficiency, agility, and security. Enterprises using VMware technology have three ways to establish an SDDC and transition at their own pace: build their own using reference architectures, use a converged infrastructure, or use a hyper-converged infrastructure (through which the full SDDC is delivered already implemented on the customer's hardware of choice).

Learn more about the *VMware SDDC*.

Hybrid and Multi-Cloud as the VMware Cloud

The term hybrid cloud describes the use of both private and public cloud platforms, working in conjunction. It can refer to any combination of cloud solutions that work together on-premises and off-site to provide cloud computing services to a company. A hybrid cloud environment allows organizations to benefit from the advantages of both types of cloud platforms and choose which cloud to use based on specific data needs.

A multi-cloud environment is as its name suggests, reflecting multiple and disparate cloud offerings and forms, all of which are part of the ubiquitous VMware Cloud.

VMware's *hybrid cloud* portfolio offers a combination of solutions that enable organizations to easily extend, protect, or replace on-premises infrastructure. These hybrid cloud offerings are built on an SDDC architecture, leveraging VMware's industry-leading compute, networking, and storage virtualization technologies.

Any combination of clouds powered by VMware creates a common operating environment across VMware-based on-premises private clouds and VMware-based public clouds. Cloud solutions from VMware Cloud Provider Partners (VCPP) including IBM, Oracle, Microsoft, Google, Amazon Web Services (AWS) and others. Native public clouds such as AWS, Azure, Oracle and Google Cloud Platform using VMware technologies including VMware Cloud Foundation, VMware vRealize® and VMware Cloud Services, along with on-premises managed cloud services such as VMware Cloud on DellEMC, form the core of VMware Cloud offerings.

This approach enables a diverse set of use cases, including regional capacity expansion, disaster recovery, application migration, data center consolidation, new application development and burst capacity.

Learn more about *VMware Hybrid Cloud*.

VMware Cloud on AWS

VMware Cloud on AWS is an on-demand service that enables customers to run applications across vSphere-based cloud environments with access to a broad range of AWS services. Powered by VMware Cloud Foundation, this service integrates vSphere, vSAN and NSX along with VMware vCenter management, and is optimized to run on dedicated, elastic, bare-metal AWS infrastructure.

With VMware Hybrid Cloud Extension (HCX), customers can easily and rapidly perform large-scale bi-directional migrations between on-premises and VMware Cloud on AWS environments.

With the same architecture and operational experience on-premises and in the cloud, IT teams can now quickly derive instant business value from use of the AWS and VMware hybrid cloud experience. VMware Cloud on AWS is ideal for enterprise IT infrastructure and operations organizations looking to migrate on-premises vSphere-based workloads to the public cloud, consolidate and extend data center capacities, and optimize, simplify and modernize their disaster recovery solutions.

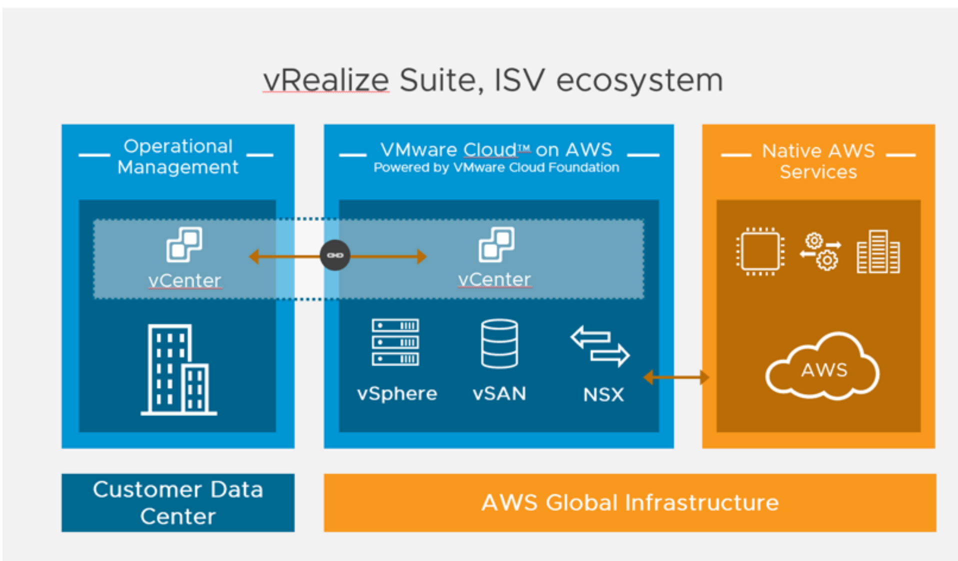


FIGURE 4. VMWARE CLOUD ON AWS

Learn more about *VMware Cloud on AWS*.

Stretched Clusters for VMware Cloud on AWS

Amazon's global infrastructure is broken up into regions. Each region supports the services for a given geography. Within each region, Amazon builds isolated and redundant islands of infrastructure called Availability Zones (AZ).

With VMware Cloud on AWS, ESXi hosts traditionally reside in an AWS AZ and are protected by vSphere HA. A new feature called Stretched Clusters for VMware Cloud on AWS is designed to protect against an AZ failure.

Now applications can span multiple AZs within a VMware Cloud on AWS compute cluster. vSAN fault domains are configured to inform vSphere and vCenter as to which hosts reside in which AZs. Each fault domain is named after the AZ it resides within to increase clarity.

Some of the advantages are:

- Zero RPO high availability for enterprise applications virtualized on vSphere across AZs, leveraging multi-AZ stretched clustering. Stretching an SDDC cluster across two AWS AZs within a region means that if an AZ goes down, it is simply treated as a vSphere HA event and the VM is restarted in the other AZ.
- Stretched clusters enable developers to focus on core application requirements and capabilities instead of infrastructure availability.
- Significant improvement of application availability without needing to architect it into the application.

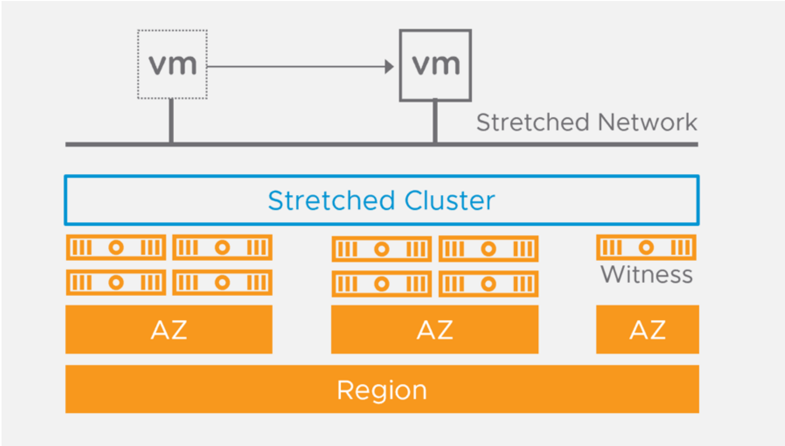


FIGURE 5. STRETCHED CLUSTER FOR VMWARE CLOUD ON AWS

Learn more about *Stretched Clusters for VMware Cloud on AWS*.

VMware Datastore Types

Datastores are logical containers, analogous to file systems, that hide specifics of physical storage and provide a uniform model for storing VM files. Datastores can also be used for storing ISO images and VM templates.

Depending on the storage utilized, datastores can be of different types. vCenter Server and ESXi support the following types of datastores:

Datastore Type	Description
VMFS (version 5 and 6)	Datastores that you deploy on block storage devices use the vSphere Virtual Machine File System (VMFS) format. VMFS is a special high-performance file system format that is optimized for storing VMs. See *Understanding VMFS Datastores*.
NFS (version 3 and 4.1)	An NFS client built into ESXi uses the Network File System (NFS) protocol over TCP/IP to access a designated NFS volume. The volume is located on a NAS server. The ESXi host mounts the volume as an NFS datastore and uses it for storage needs. ESXi supports versions 3 and 4.1 of the NFS protocol. See *Understanding Network File System Datastores.*
vSAN	vSAN aggregates all local capacity devices available on the hosts into a single datastore shared by all hosts in the vSAN cluster. See the *Administering VMware vSAN*.
Virtual Volumes (vVols)	vVols datastore represents a storage container in vCenter server and vSphere client. See *Working with Virtual Volumes*.

TABLE 2. TYPES OF VMWARE DATASTORES

The following table compares networked storage technologies that ESXi supports:

Technology	Protocols	Transfers	Interface
Fibre Channel	FC/SCSI	Block access of data/LUN	FC HBA
Fibre Channel over Ethernet	FCoE/SCSI	Block access of data/LUN	• Converged Network Adapter (hardware FCoE) • NIC with FCoE support (software FCoE)
iSCSI	IP/SCSI	Block access of data/LUN	• iSCSI HBA or iSCSI enabled NIC (hardware iSCSI) • Network adapter (software iSCSI)
NAS	IP/NFS	File (no direct LUN access)	Network adapter

TABLE 3. NETWORKED STORAGE THAT ESXI SUPPORTS

The following table compares the vSphere features that different types of storage support:

Storage Type	Boot VM	vMotion	Datastore	RDM	VM Cluster	VMware HA and DRS	Storage APIs - Data Protection
Local Storage	Yes	No	VMFS	No	Yes	No	Yes
Fibre Channel	Yes	Yes	VMFS	Yes	Yes	Yes	Yes
iSCSI	Yes	Yes	VMFS	Yes	Yes	Yes	Yes
NAS over NFS	Yes	Yes	NFS 3 and NFS 4.1	No	No	Yes	Yes

TABLE 4. VSPHERE FEATURES SUPPORTED BY STORAGE

Learn more about *VMware Datastore storage types*.

Raw Device Mapping and Virtual Disk

VMFS is a high-performance cluster file system (CFS) that provides storage virtualization that is optimized for VMs. Each VM is encapsulated in a small set of files. VMFS is the default storage-management interface for these files on physical SCSI disks and partitions.

Virtual disks (VMDK) are virtual disks that are attached to a VM which in turn detects the VMDK as a local SCSI target. The virtual disks are simply files on the VMFS volume.

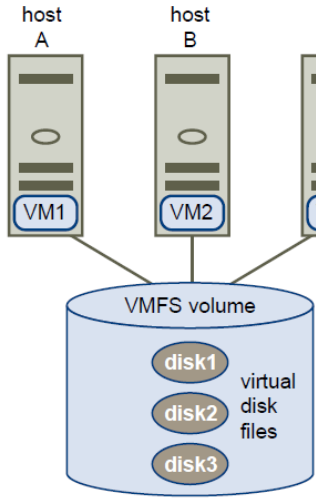


FIGURE 6. VIRTUAL MACHINE FILE SYSTEM

Learn more about *VMware VMDKs*.

A raw device mapping (RDM) file is a special file in a VMFS volume that manages metadata for its mapped device. To the VM, the storage virtualization layer presents the mapped device as a virtual SCSI device. RDM files contain metadata used to manage and redirect disk accesses to the physical device.

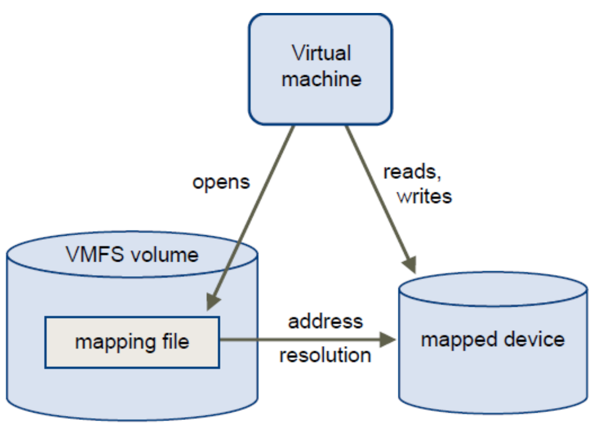


FIGURE 7. RAW DEVICE MAPPING FILE

Learn more about *VMware RDMs*.

The following technical features of VMFS are among those that make it suitable for use in a virtual environment:

- Automated file system with hierarchical directory structure
- Optimization for VMs in a clustered environment

- Lock management and distributed logical volume management
- Dynamic datastore expansion by spanning multiple storage extents
- CFS with journal logging for fast recovery
- Thin-provisioned virtual disk format for space optimization
- VM–level point-in-time snapshot copy management
- Encapsulation of the entire VM state in a single directory
- Support for VMware vSphere Storage APIs – Array Integration (VAAI)

In some cases, RDM storage access remains a better solution. The following scenarios call for raw disk mapping:

- Using Microsoft Server Failover Cluster (WSFC) for clustering in a virtual environment (changing with the release of vSphere 7)
- Heavy dependence on storage snapshot and replication
- Implementing N-Port ID virtualization (NPIV)

Two compatibility modes are available for RDMs:

- **Virtual compatibility mode**: the RDM acts like a virtual disk file. The RDM can use snapshots. The RDMs in the case are known as virtual RDMs.
- **Physical compatibility mode**: the RDM offers direct access to the SCSI device for those applications that require lower-level control. The RDMs in the case are known as physical RDMs.

Learn more about *VMware RDM modes*.

VMware recommends using VMDKs for Oracle deployments to take advantage of all of the features VMFS has to offer. RDM can be used for specific use cases as outlined above.

Learn more about the advantages of *VMware VMDK and RDM*.

VMware Virtual Disk Provisioning Policies

When performing certain VM management operations, it's possible to specify a provisioning policy for the virtual disk file. The operations include creating a virtual disk, cloning a VM to a template, or migrating a VM with VMware vSphere® Storage vMotion®.

You can also use vSphere Storage vMotion or cross-host vSphere Storage vMotion to transform virtual disks from one format to another.

Option	Description
Thick Provision Lazy Zeroed	Creates a virtual disk in a default thick format. Space required for the virtual disk is allocated when the disk is created. Data remaining on the physical device is not erased during creation but is zeroed out on demand later on first write from the VM. VMs do not read stale data from the physical device.
Thick Provision Eager Zeroed (EZT)	A type of thick virtual disk that supports clustering features such as the multi-writer attribute for Oracle RAC. Space required for the virtual disk is allocated at creation time. In contrast to the thick provision lazy zeroed format, the data remaining on the physical device is zeroed out when the virtual disk is created. Creating virtual disks in this format may take longer than creation of other types of disks. Increasing the size of an eager zeroed thick virtual disk causes a significant stun time for the VM.

Thin Provision	Use this format to save storage space. For the thin disk, provision as much datastore space as the disk would require based on the value entered for the virtual disk size. The thin disk starts small and, at first, uses only as much datastore space as the disk needs for its initial operations. If the thin disk needs more space later, it can grow to its maximum capacity and occupy the entire datastore space provisioned to it. Thin provisioning is the fastest method to create a virtual disk because it creates a disk with only the header information. It does not allocate or zero out storage blocks. Storage blocks are allocated and zeroed out when they are first accessed.

TABLE 5. VIRTUAL DISK FORMATS AVAILABLE IN VSPHERE STORAGE VMOTION

The below table shows the various VMDK modes:

Option	Description
Dependent	Dependent disks are included in snapshots.
Independent-persistent	Disks in persistent mode behave like conventional disks on a physical computer. All data written to a disk in persistent mode is written permanently to the disk.
Independent– non-persistent	Changes to disks in non-persistent mode are discarded when the VM is turned off or reset. Non-persistent mode enables restarting of the VM with a virtual disk in the same state every time. Changes to the disk are written to and read from a redo log file that is deleted when the VM is turned off or reset.

TABLE 6. VMDK MODES

Learn more about *VMware virtual disk provisioning policies*.

VMware Multi-Writer Attribute for Shared VMDKs

VMFS is a clustered file system that disables (by default) multiple VMs from opening and writing to the same virtual disk (.vmdk file). This prevents more than one VM from inadvertently accessing the same .vmdk file. The multi-writer option allows VMFS-backed disks to be shared by multiple VMs. An Oracle RAC cluster using shared storage is a typical use case.

As occurs with VMFS, vVol (beginning with ESXi 6.5), and NFS datastores, VMware vSAN also prevents multiple VMs from opening the same VMDK in read-write mode.

Some of the current restrictions of the multi-writer attribute, which are documented in *KB 1034165*, are:

- Storage vMotion is disallowed
- Snapshots not supported (snapshots of VMs with independent-persistent disks are supported, however)
- Changed block tracking (CBT) not supported
- Cloning, hot extend virtual disk not supported

Independent persistent mode is **not** required for enabling the multi-writer attribute.

In the case of VMware vSphere (non vSAN Storage) on VMFS, vVol (beginning with ESXi 6.5) and NFS datastores, using the multi-writer attribute to share the VMDKs for Oracle RAC requires:

- SCSI bus sharing needs to be set to **none**
- VMDKs must be EZT; thick provision lazy zeroed or thin-provisioned formats are not allowed

In the case of NFS datastores that do not support vSphere APIs for array integration (VAAI), and hence allow creation of EZT VMDDs via the web client, refer to *KB 2147691* for steps to create EZT VMDKs.

The above procedure for VMDKs should also be followed for virtual RDMs.

In the case of physical RDM(s), sharing physical RDM(s) for Oracle RAC requires:

- SCSI bus sharing is set to **physical**
- Compatibility mode for the shared RDM is set to **physical** for physical compatibility mode

In the case of VMware vSAN, the following is required:

- SCSI bus sharing is set to **none**
- Prior to vSAN 6.7 Patch P01, the virtual disk must be EZT to enable multi-writer mode
- Beginning with VMware vSAN 6.7 Patch P01 (ESXi 6.7 Patch Release ESXi670-201912001), Oracle RAC on vSAN does **not** require shared VMDKs to be EZT (OSR=100) for multi-writer mode to be enabled

Find extensive documentation in *KB 1034165* for VMware non-vSAN and *KB 2121181* for VMware vSAN.

Further explanation can also be found in the following VMware blog posts:

- *Oracle RAC on vSAN 6.7 P01 – No more Eager Zero Thick requirement for shared VMDKs*
- *Oracle RAC storage migration from non-vSAN to vSAN 6.7 P01 – Through Thick to Thin*

RAC shared storage provisioning along with changes for VMware platforms on-premise are summarized in the table below:

VMware Platform	Datastore	Version	RAC shared VMDK requirement for multi-writer attribute	Reference
VMware vSphere	VMFS	ESXi 5.x and above	EZT	*KB 1034165*
VMware vSphere	NFS	ESXi 5.x and above	EZT	*KB 1034165*
VMware vSphere	vVol	ESXi 6.5 and above	Vendor specific	*KB 1034165* and *KB 2113013*
VMware vSAN	vsanDatastore	Prior vSAN 6.7 Patch P01	EZT	*KB 2121181*
VMware vSAN	vsanDatastore	Beginning with VMware vSAN 6.7 Patch P01 (ESXi 6.7 Patch Release ESXi670-201912001)	Thin-Provisioned	*KB 2121181*

TABLE 7. VMWARE PLATFORM REQUIREMENTS FOR RAC SHARED STORAGE PROVISIONING

Shared Disks Using Raw Device Mapping

As mentioned above, there are two compatibility modes available for RDMs:

- Virtual compatibility mode (virtual RDMs)
- Physical compatibility mode (physical RDMs)

Set the type of SCSI bus sharing for a VM and indicate whether to share the SCSI bus. Depending on the type of sharing, VMs can access the same virtual disk simultaneously on the same server or on any other server.

Change the SCSI controller configuration for a VM only if the VM is on an ESXi host.

SCSI bus sharing options are:

Option	Description
None	Virtual disks cannot be shared by other virtual machines.
Virtual	Virtual disks can be shared by virtual machines on the same server.
Physical	Virtual disks can be shared by virtual machines on the any server.

TABLE 8. SCSI BUS SHARING OPTIONS

Learn more about *SCSI bus sharing*.

Important observations to keep in mind:

- SCSI bus sharing ensures a VM can register the keys for SCSI 3 persistent reservation.
- In order to use physical RDMs as shared storage for Oracle RAC, the multi-writer attribute should not be set as physical bus sharing indirectly leads to disk opened in multi-writer mode.

Additionally, note the following from the Oracle MySupport document RAC: *Frequently Asked Questions (RAC FAQ) (Doc ID 220970.1)*

- Oracle Clusterware and Oracle RAC do **not** require nor use SCSI-3 persistent group reservation (PGR) for a Oracle Clusterware-only installations.
- In a native Oracle RAC Stack (no third-party or vendor cluster, nor Oracle Solaris Cluster) SCSI-3 PGR is not required by Oracle and should be disabled on the storage (for disks / LUNs used in the stack).
- When using a third-party or vendor-cluster solution such as Symantec Veritas SFRAC, the third-party cluster solution may require that SCSI-3 PGR be enabled on the storage, as those solutions will use SCSI-3 PGR as part of their I/O fencing procedures.

The following steps should be followed when adding shared RDM(s) in physical compatibility mode to Oracle RAC VMs:

- Set SCSI bus sharing to **physical** for those SCSI controllers where the shared RDM(s) will be added to.
- Set the compatibility mode for the shared RDM to **physical** for physical compatibility mode.

In order to use virtual RDMs as shared storage for Oracle RAC, the multi-writer attribute should be used. Follow the same procedure as outlined for shared VMDKs using the multi-writer attribute, as detailed in *KB 1034165* for VMware vSphere (non vSAN Storage).

Using shared physical and virtual RDM(s) for Oracle RAC is clearly explained using examples and screenshots in *Oracle RAC and VMware Raw Device Mapping (RDM)*.

Learn more about *VMware RDM considerations and limitations*.

Shared disks using VMware vSphere Virtual Volumes

Beginning with VMware vSphere 6.5, vVols 2.0, and VASA 3.0, vVols are now validated to support Oracle RAC workloads delivering policy-based, VM-centric storage for Oracle RAC clusters.

See *Whats New in Virtual Volumes (vVols) 2.0* **for further details.**

Learn more about *shared disks using vVols*.

RAC shared storage provisioning for vVols is summarized in the table below:

VMware Platform	Datastore	Version	RAC shared VMDK requirement for multi-writer attribute	Reference
VMware vSphere	vVol	ESXi 6.5 and later	Vendor specific	*KB 1034165* and *KB 2113013*

TABLE 9. SHARED STORAGE PROVISIONING FOR VSPHERE VIRTUAL VOLUMES

Storage Policy-Based Management

Storage policy-based management (SPBM) is a storage policy framework that helps administrators match VM workload requirements to storage capabilities. SPBM runs as an independent service in the vCenter Server and helps to align storage with the application demands of VMs.

SPBM enables the following mechanisms:

- Advertisement of storage capabilities and data services that storage arrays and other entities such as I/O filters enable
- Bidirectional communications between ESXi and vCenter Server on one side, with storage arrays and entities on the other
- VM provisioning based on VM storage policies

Administrators build policies by selecting the desired capabilities of the underlying storage array. The SPBM engine interprets the storage requirements of individual applications specified in policies associated with individual VMs and dynamically composes the storage service, placing the VM on the right storage tier, allocating capacity, and instantiating the necessary data services (e.g., snapshots, replication).

As an abstraction layer, SPBM abstracts storage services delivered by vVols, vSAN, I/O filters, or other storage entities.

SPBM integrates with vSAN to discover the capabilities the vSAN setup offers and presents the administrators options to author storage policies custom-made for the workload requirement.

Rather than integrating with each individual type of storage and data service, SPBM provides a universal framework for different types of storage entities.

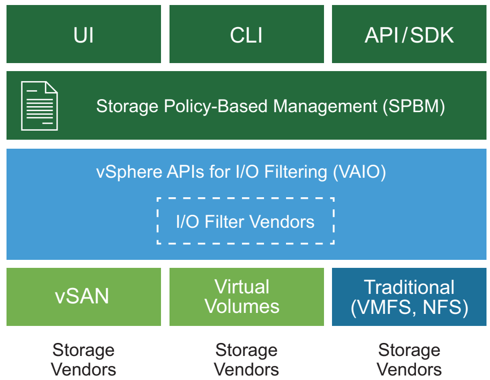


FIGURE 8. SPBM UNIVERSAL FRAMEWORK

Learn more about *VMware SPBM*.

VMware vSAN Storage Policy

VMware vSAN forms the storage backbone of traditional VMware vSAN clusters and VMware Cloud on AWS. VMware vSAN storage polices define storage requirements for VMs. These policies guarantee the required level of service for VMs, as they determine how VM storage is allocated.

The default policy contains vSAN rule sets and a set of basic storage capabilities typically used for the placement of VMs deployed on vSAN datastores.

Storage Policy Specification	Description
Number of Failures to tolerate	Defines the number of hosts, disk, or network failures a VM object can tolerate. For n failures tolerated, n+1 copies of the VM object are created and 2n+1 hosts with storage are required. The settings applied to the VMs on the Virtual SAN datastore determines the datastore's usable capacity. Default is 1
Object space reservation	Percentage of the object logical size that should be reserved during the object creation. The default value is 0 percent and the maximum value is 100 percent
Number of disk stripes per object	This policy defines how many physical disks across each copy of a storage object are striped. The default value is 1 and the maximum value is 12
Flash read cache reservation	Flash capacity reserved as read cache for the VM object. Specified as a percentage of the logical size of the VMDK object. It is set to 0 percent by default and Virtual SAN dynamically allocates read cache to storage objects on demand. Default is 0
Force provisioning	No

TABLE 10. STORAGE POLICY RULES

Learn more about *VMware vSAN storage policy*.

Object space reservation (OSR) determines the percentage of the logical size of the VMDK object that must be reserved or thick provisioned when deploying VMs.

The default value is 0% (thin provisioning) and maximum value is 100% (thick provisioning).

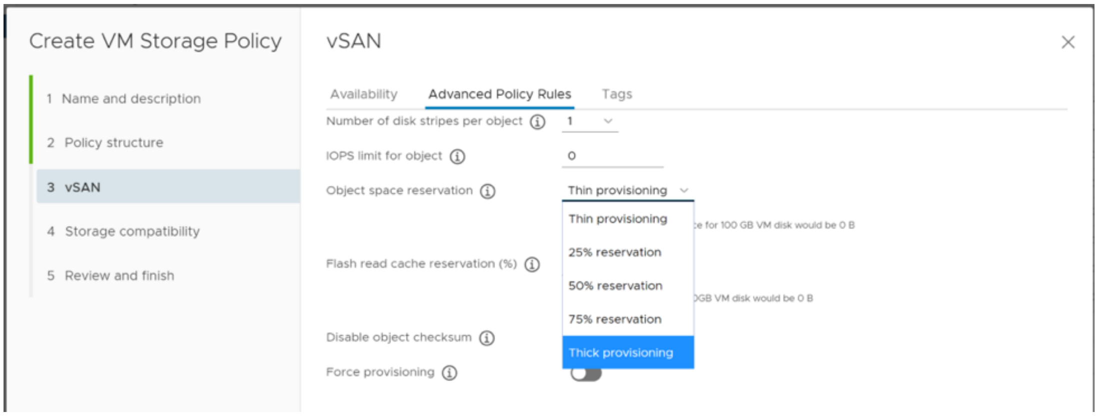


FIGURE 9. CREATING VM STORAGE POLICY

VMware vSAN enables VM-specific setting of availability, capacity, and performance policies.

Learn more about *OSR*.

VMware vSphere Virtual Volume Storage Policy

For vVols, VMware provides a default storage policy that contains no rules or storage requirements (vVol No Requirements Policy). This policy is applied to VM objects when a VM's policy is unspecified on the vVols datastore. With vVol No Requirements Policy, storage arrays can determine the optimum placement for VM objects.

The default vVol No Requirements Policy provided by VMware has the following characteristics:

- The policy cannot be deleted, edited, or cloned.
- The policy is compatible only with vVols datastores.
- Ability to create a VM storage policy for vVols and designate it as the default.

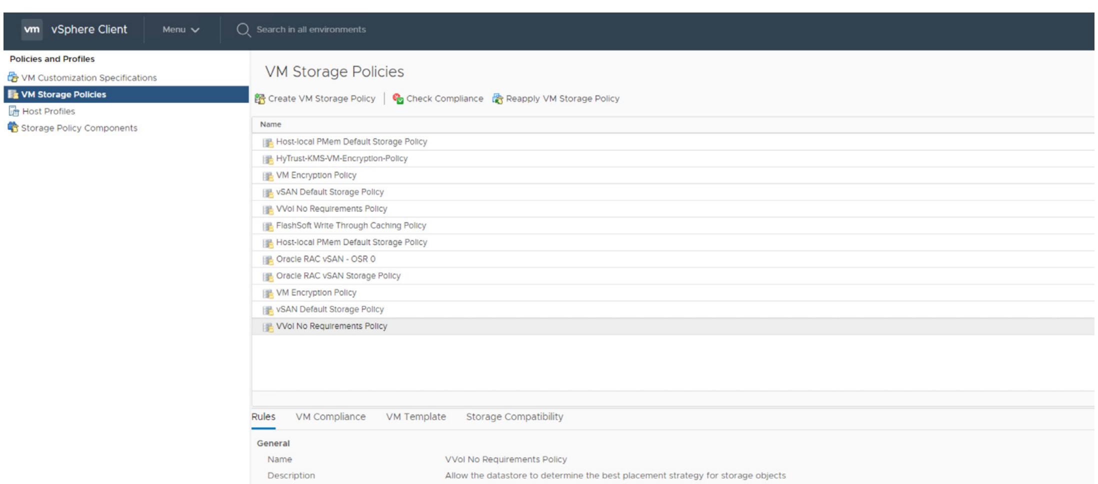


FIGURE 10. DEFAULT VVOLS NO REQUIREMENTS POLICY

Learn more about *vVol storage policy*.

VMware Distributed Switch and Distributed Port Group

A vSphere distributed switch provides centralized management and monitoring of the networking configuration of all hosts that are associated with the switch. A distributed switch requires vCenter Server system and its settings are propagated to all hosts that are associated with the switch.

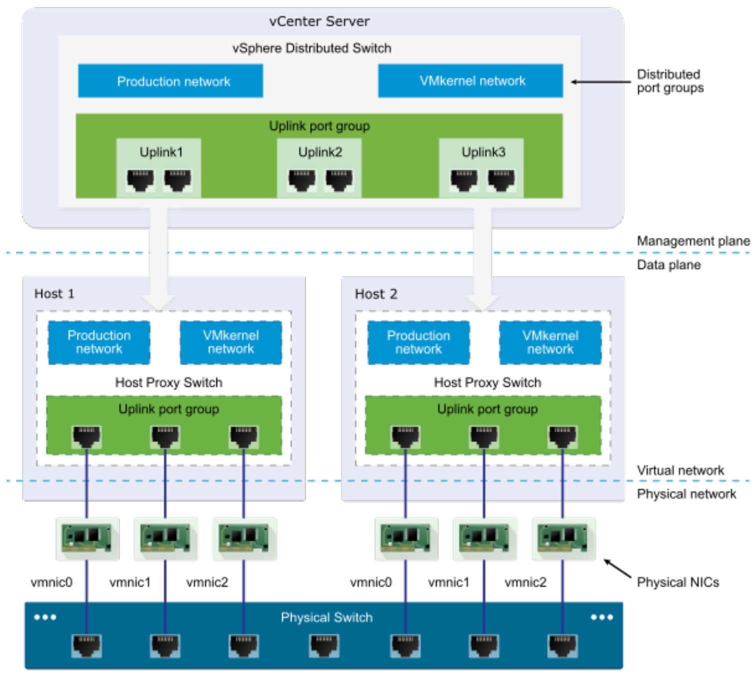


FIGURE 11. VMWARE VSPHERE DISTRIBUTED SWITCH

Distributed port groups provide network connectivity to VMs and accommodate VMkernel traffic. The identity for each distributed port group can be set using a network label, which must be unique to the current data center. NIC teaming, failover, load balancing, VLAN, security, traffic shaping, and other policies on distributed port groups are all configurable. The virtual ports connected to a distributed port group share the same properties that are configured to the distributed port group.

As with uplink port groups, the configuration applied to distributed port groups on vCenter Server (the management plane) is automatically propagated to all hosts on the distributed switch through their host proxy switches (the data plane). In this way, it's possible to configure a group of VMs to share the same networking configuration by associating the VMs to the same distributed port group.

Learn more about *VMware distributed switch and distributed port group*.

Oracle recommends private interconnect adapters be on their own dedicated VLAN with adequate bandwidth for cache fusion traffic. Details can be found in *Oracle Network Interface Hardware Minimum Requirements section*.

On VMware vSphere and VMware vSAN platforms, a distributed port group, **DPortGroup-OraclePrivate**, is created and dedicated for Oracle RAC private interconnect. Both Oracle and VMware recommend using dedicated uplinks for this port group, if possible, to guarantee adequate bandwidth for Oracle RAC private interconnect traffic.

For converged networks architecture, ensure that QoS is set for the Oracle RAC private interconnect traffic.

VMware Cloud on AWS uses network segments, as opposed to distributed port groups. Segments are logical networks for use by workload VMs in the SDDC compute network.

VMware Cloud on AWS supports three types of logical network segments: routed, extended and disconnected.

- A routed network segment (the default type) has connectivity to other logical networks in the SDDC and through the SDDC firewall to external networks.
- An extended network segment extends an existing L2VPN tunnel, providing a single IP address space that spans the SDDC and on-premises network.
- A disconnected network segment has no uplink and provides an isolated network accessible only to VMs connected to it. Disconnected segments are created when needed by HCX (see *Getting started with VMware HCX*). These can also be created and converted to other segment types.

Learn more about *VMware Cloud on AWS network segment*.

The RAC private interconnect networking setup and VMware platform changes on-premises are shown below:

VMware Platform	Distributed Switch	Distributed Port Group	Version	Reference
VMware vSphere on-premises	Regular Distributed Switch	Dedicated Distributed Port Group for RAC	vSphere 5.5 and later	*https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.networking.doc/GUID-D21B3241-0AC9-437C-80B1-0C8043CC1D7D.html*

TABLE 11. RAC PRIVATE INTERCONNECT NETWORKING SETUP

VMware NSX and Micro-Segmentation

VMware NSX is a network virtualization platform that for the first time makes micro-segmentation economically and operationally feasible. NSX provides the networking and security foundation for the SDDC, enabling the three key functions of micro-segmentation: isolation, segmentation, and segmentation with advanced services.

Businesses gain key benefits with micro-segmentation:

- **Network security *inside* the data center**: flexible security policies aligned to virtual network, VM, OS type, dynamic security tag, and more, for granularity of security down to the virtual NIC.
- **Automated deployment for data center agility**: security policies are applied when a VM spins up, are moved when a VM is migrated, and are removed when a VM is deprovisioned – no more stale firewall rules.
- **Integration with leading networking and security infrastructure**: NSX is the platform enabling an ecosystem of partners to integrate – adapting to constantly changing conditions in the data center to provide enhanced security. Best of all, NSX runs on existing data center networking infrastructure.

These micro-segmentation capabilities make NSX ideal for securing intra-data center network traffic, for fully isolating disparate networks (e.g., for highly sensitive workloads or for multi-tenancy), and for simplifying networks that would otherwise require complex access policies.

Learn more about *VMware NSX*.

VMware vSphere High Availability

VMware vSphere HA clusters enable a collection of ESXi hosts to work together so that, as a group, they provide higher levels of availability for VMs than each ESXi host can provide individually.

Hosts in the cluster are monitored and, in the event of a failure, the VMs on a failed host are restarted on alternate hosts.

When a vSphere HA cluster is created, a single host is automatically elected as the master host. The master host communicates with vCenter Server and monitors the state of all protected VMs and the state of the slave hosts. Different types of host failures are possible and the master host must detect and appropriately deal with the failure. The master host must distinguish between a failed host and one that is in a network partition, or one that has become network isolated. The master host uses network and datastore heartbeating to determine the type of failure.

Learn more about *VMware vSphere HA*.

VMware vSphere Distributed Resource Scheduler and Affinity Rules

VMware vSphere Distributed Resource Scheduler™ (DRS) affinity and anti-affinity rules help DRS perform better placements of VMs by understanding the application dependencies and availability.

The placement of VMs on hosts within a cluster can be controlled by using affinity rules. Two types of rules can be created:

- **VM-host affinity rules**: used to specify affinity or anti-affinity between a group of VMs and a group of hosts. An affinity rule specifies that the members of a selected VM DRS group can or must run on the members of a specific host DRS group. An anti-affinity rule specifies that the members of a selected VM DRS group cannot run on the members of a specific host DRS group.
- **VM-VM affinity rules**: used to specify affinity or anti-affinity between individual VMs. A rule specifying affinity causes DRS to try to keep the specified VMs together on the same host, for example, for performance reasons. With an anti-affinity rule, DRS tries to keep the specified VMs apart, for example, so that when a problem occurs with one host, only one of two VMs is lost.

When an affinity rule is added or edited, and the cluster's current state is in violation of the rule, the system continues to operate and tries to correct the violation.

Learn more about *vSphere DRS and affinity rules*.

A VM-host affinity rule specifies whether or not the members of a selected VM DRS group can run on the members of a specific host DRS group.

Unlike a VM-VM affinity rule, which specifies affinity (or anti-affinity) between individual VMs, a VM-host affinity rule specifies an affinity relationship between a group of VMs and a group of hosts. There are required rules (designated by must) and preferential rules (designated by **should**).

As part of creating the VM-host affinity rule, the following specifications for the rule are indicated:

- **Must run on hosts in group** (e.g., VMs in VM group one must run on hosts in host group A)
- **Should run on hosts in group** (e.g., VMs in VM group one should, but are not required, to run on hosts in host group A)
- **Must not run on hosts in group** (e.g., VMs in VM group one must never run on host in host group A)
- **Should not run on hosts in group** (e.g., VMs in VM group one should not, but may, run on hosts in host group A)

A notable new feature of many available on VMware Cloud on AWS is the ability to create compute policies and use vSphere tags and attributes.

Compute policies enable customers to define VM placement constraints as preferential policies in their SDDC by leveraging inventory tags. In a multi-cluster environment, a single policy can be defined to constrain the placement of tagged VMs using the following capabilities:

- **Simple VM-host affinity**: this capability constrains the placement of tagged VMs on specifically tagged hosts in each cluster, thereby circumventing the need to define rules on a per-cluster basis.
- **VM-VM anti-affinity**: this policy allows the user to specify anti-affinity relations between a group of VMs. These groups of VMs are identified using vSphere tags. The policy automatically applies to all the VMs that have the tags specified in the policy. DRS will try to ensure that all the VMs in the vCenter that have the policy's VM-tag, are preferably placed on separate hosts.
- **Disable DRS vMotion**: this policy allows the user to specify that a VM not be migrated away from the host on which it was powered on, unless the host is placed into maintenance mode.

Learn more about *compute policies*.

vSphere tags and attributes allow attachment of metadata to objects in the vSphere inventory to make it easier to sort and search for these objects.

A tag is a label that can be applied to objects in the vSphere inventory. Once a tag is created, it is assigned to a category. Categories allow related tags to be grouped together. Defining a category enables administrators to specify the object types for the category's tags and decide if more than one tag in the category can be applied to an object.

For vSphere tags and attributes, VMware Cloud on AWS supports the same set of tasks as an on-premises SDDC.

Learn more about *vSphere tags and attributes*.

Oracle Database 19c

Oracle Database 19c, the latest generation of the world's most popular database, provides businesses of all sizes with access to the world's fastest, most scalable and reliable database technology for secure and cost-effective deployment of transactional and analytical workloads in the cloud, on-premises and hybrid cloud configurations.

Oracle Database 19c is the final, and therefore *long-term support* release of the Oracle Database 12c family of products (which includes Oracle Database 18c).

Oracle Database 19c builds upon the innovations of previous releases such as multi-tenant, in-memory, JSON support, sharing and many other features that enable Oracle's Autonomous Database Cloud Services. This latest release of the world's most popular database also introduces new functionality, providing customers with a multi-model enterprise-class database for all their typical use cases, including:

Operational database use cases such as traditional transactions, real-time analytics, JSON document stores and Internet of Things (IoT) applications

Analytical database use cases such as; traditional and real-time data warehouses and data marts, big data lakes and graph analytics

Learn more about *Oracle Release 19c*.

Oracle Database Architecture

An Oracle Database server consists of a database and at least one database instance in case of a single instance database. In case of RAC, an Oracle Database will have more than one instance accessing the database.

- A database is a set of files, located on disk, that store data. These files can exist independently of a database instance.
- An instance is a set of memory structures that manage database files. The instance consists of a shared memory area, called the system global area (SGA), and a set of background processes. An instance can exist independently of database files.

The physical database structures that comprise a database are:

- **Data files**: every Oracle Database has one or more physical data files, which contain all the database data. The data of logical database structures, such as tables and indexes, is physically stored in the data files.
- **Control files**: every Oracle Database has a control file. A control file contains metadata specifying the physical structure of the database, including the database name and the names and locations of the database files.

- **Online redo log files**: every Oracle Database has an online redo log, which is a set of two or more online redo log files. An online redo log is made up of redo entries (also called redo log records), which record all changes made to data.
- Many other files including parameter files, archived redo files, backup files and networking files are important to any Oracle Database operations.

Learn more about *Oracle Database Architecture*.

Oracle Multitenant Architecture

The multitenant architecture enables an Oracle Database to function as a multitenant container database (CDB).

A CDB includes zero, one, or many customer-created pluggable databases (PDBs). A PDB is a portable collection of schemas, schema objects, and non-schema objects that appears to an Oracle Net client as a non-CDB.

All Oracle databases before Oracle Database 12c were non-CDBs.

Learn more about *Oracle Multitenant Architecture*.

Oracle Automatic Storage Management

Oracle Automatic Storage Management (ASM) is a volume manager and a file system for Oracle Database files that supports single-instance Oracle Database and Oracle RAC configurations.

Oracle ASM is Oracle's recommended storage management solution that can be used for both Oracle RAC and single instance Oracle Databases and provides an alternative to conventional volume managers, file systems, and raw devices.

Oracle ASM uses disk groups to store data files. An Oracle ASM disk group is a collection of disks that Oracle ASM manages as a unit. Disks can be added or removed from a disk group while a database continues to access files from the disk group.

Oracle ASM simplifies storage management through the principle of stripe-and-mirror-everything (SAME). Intelligent mirroring capabilities allow administrators to define 2-or 3-way mirrors to protect vital data. When a read operation identifies a corrupt block on a disk, Oracle ASM automatically relocates the valid block from the mirrored copy to an uncorrupted portion of the disk.

Learn more about *Oracle Automatic Storage Management*.

Oracle ASMLIB and ASMFD

Oracle ASMLIB maintains permissions and disk labels that are persistent on the storage device, so that the label is available even after an operating system upgrade.

The Oracle ASM library driver simplifies the configuration and management of block disk devices by eliminating the need to rebind block disk devices used with Oracle ASM each time the system is restarted.

With Oracle ASMLIB, administrators can define the range of disks to be made available as Oracle ASM disks. Oracle ASMLIB maintains permissions and disk labels that are persistent on the storage device, so that the label is available even after an operating system upgrade.

Learn more about *Oracle ASMLIB*.

Oracle ASM filter driver (ASMFD) helps prevent corruption in Oracle ASM disks and files within the disk group.

Oracle ASMFD rejects write I/O requests that are not issued by Oracle software. This write filter helps to prevent users with administrative privileges from inadvertently overwriting Oracle ASM disks, thus preventing corruption in Oracle ASM disks and files within the disk group. For disk partitions, the area protected is the area on the disk managed by Oracle ASMFD, assuming the partition table is left untouched by the user.

Oracle ASMFD simplifies the configuration and management of disk devices by eliminating the need to rebind disk devices used with Oracle ASM each time the system is restarted.

Learn more about *Oracle ASMFD*.

Linux Device Persistence and Udev Rules

Device names in Linux are not guaranteed to be persistent across reboots (e.g., a device named /dev/sdb can be renamed as /dev/sdc on next reboot). To guarantee device persistence across reboots, Linux udev rules may be used.

Learn more about *configuring device persistence for Oracle storage*.

Oracle Clusterware

Oracle Clusterware is a portable cluster software that provides comprehensive multi-tiered high availability and resource management for consolidated environments. It supports clustering of independent servers so that they cooperate as a single system.

Oracle Clusterware is the integrated foundation for Oracle RAC, and the high-availability and resource management framework for all applications on any major platform. Oracle Clusterware was first released with Oracle Database 10g release 1 (10.1) as the required cluster technology for the Oracle multi-instance database, Oracle RAC. The intent is to leverage Oracle Clusterware in the cloud to provide enterprise-class resiliency where required, and dynamic, online allocation of compute resources where and when they are needed.

Learn more about *Oracle Clusterware 19c*.

Oracle Clusterware and Oracle RAC do not require nor use SCSI-3 persistent group reservation (PGR). In a native Oracle RAC stack (no third-party or vendor cluster or Oracle Solaris cluster) SCSI-3 PGR is not required by Oracle and should be disabled on the storage (for disks/LUNs used in the stack).

Oracle support positions the reasoning for this as follows: "If you have PR enabled on a device, the device would have a default behavior regarding PR expecting the client that makes use of PR to call the right commands as required. However, the default behavior of a device for which PR is enabled depends on the platform, the device driver, and the PR setting and may not work for Oracle Clusterware and the RAC stack, since we do not call these commands to operate the PR behavior."

When using a third-party or vendor cluster solution such as Symantec Veritas SFRAC, the third-party cluster solution may require that SCSI-3 PGR is enabled on the storage, as those solutions will use SCSI-3 PGR as part of their I/O fencing procedures.

This is extensively documented in Oracle MySupport (formerly Metalink) article *RAC: Frequently Asked Questions (RAC FAQ) (Doc ID 220970.1)*.

Oracle Restart

Oracle Restart enhances the availability of Oracle databases in a single-instance environment. When Oracle Restart is installed, various Oracle components can be automatically restarted after a hardware or software failure or whenever the database host computer restarts.

Component	Notes
Database instance	Oracle Restart can accommodate multiple databases on a single host computer.
Oracle Net listener	-

Database services	Does not include the default service created upon installation because it is automatically managed by Oracle Database. Also does not include any default services created during database creation or global services. For more information about global services, see the *Oracle Database Global Data Services Concepts and Administration Guide*.
Oracle ASM instance	-
Oracle ASM disk groups	Restarting a disk group means mounting it.
Oracle Notification Services (ONS)	In an Oracle Grid Infrastructure for Standalone Servers (Oracle Restart) environment, ONS can be used in Oracle Data Guard installations for automating failover of connections between primary and standby database through fast application notification (FAN). ONS is a service for sending FAN events to integrated clients upon failover.

TABLE 12. ORACLE COMPONENTS AUTOMATICALLY RESTARTED BY ORACLE RESTART

Oracle Restart runs periodic check operations to monitor the health of these components. If a check operation fails for a component, the component is shut down and restarted. Oracle Restart is used in standalone server (non-clustered) environments only. For Oracle RAC environments, the functionality to automatically restart components is provided by Oracle Clusterware.

Oracle Restart runs out of the Oracle Grid Infrastructure home, which is installed separately from Oracle Database homes.

Learn more about *Oracle Restart*.

Oracle RAC One Node

Oracle RAC One Node is a single instance of an Oracle RAC database that runs on one node in a cluster. This option adds to the flexibility that Oracle offers for database consolidation. Many databases can be consolidated into one cluster with minimal overhead while also providing the high availability benefits of failover protection, online rolling patch application, and rolling upgrades for the operating system and Oracle Clusterware.

It requires no application changes, can support any Oracle Database workloads, and is easily upgraded to a full multi-instance Oracle RAC configuration. It eliminates the single database server as a single point of failure and takes further advantage of clustering to apply rolling patches and database service relocation without incurring downtime.

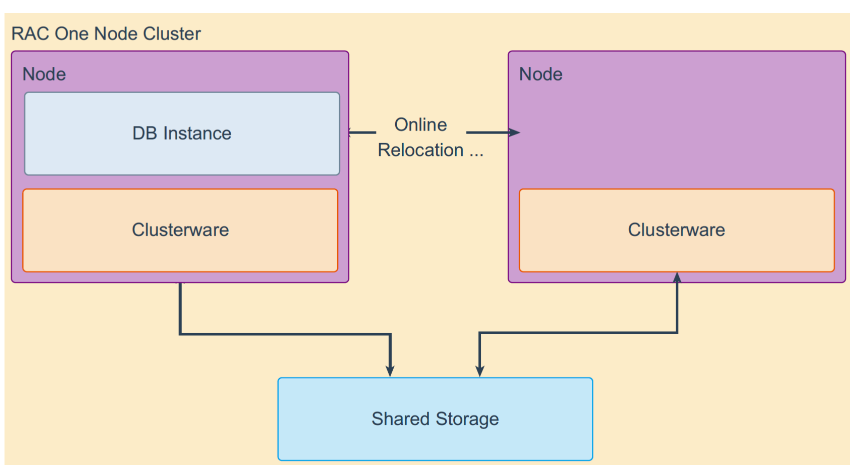


FIGURE 13. RAC ONE NODE (*ORACLE CORPORATION*)

Learn more about *Oracle RAC One Node 19c*.

Oracle Real Application Cluster

Non-cluster Oracle Database instances have a one-to-one relationship between Oracle Database and the instance. Oracle RAC environments, however, have a one-to-many relationship between the database and instances. An Oracle RAC database can have several instances, all of which access one Oracle Database. All database instances must use the same interconnect, which can also be used by Oracle Clusterware.

The combined processing power of the multiple servers can provide greater throughput and Oracle RAC scalability than is available from a single server.

A cluster comprises multiple interconnected computers or servers that appear as if they are one server to end users and applications. The Oracle RAC option with Oracle Database enables clustering of Oracle Database instances. Oracle RAC uses Oracle Clusterware for the infrastructure to bind multiple servers, so they operate as a single system.

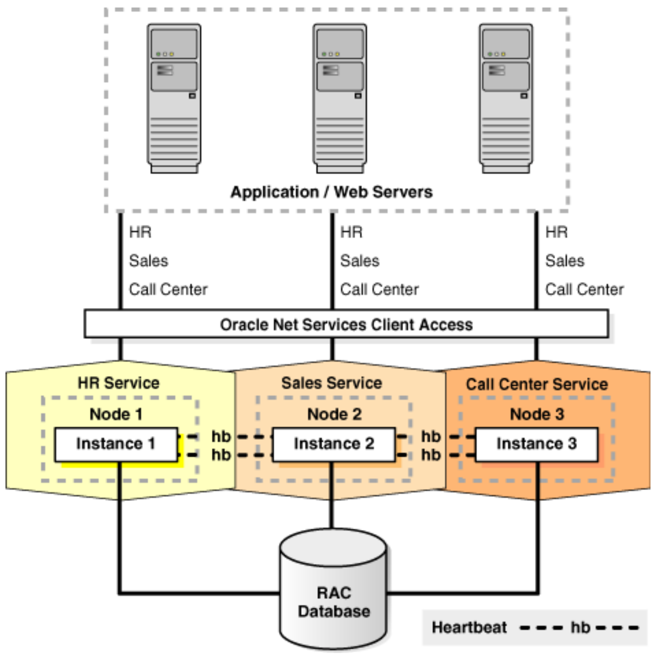


FIGURE 14. ORACLE RAC (*ORACLE CORPORATION*)

Learn more about *Oracle RAC 19c*.

Extended Oracle RAC

An Oracle extended cluster consists of nodes that are in multiple locations called sites.

When you deploy an Oracle standalone cluster, you can also choose to configure the cluster as an Oracle extended cluster. You can extend an Oracle RAC cluster across two or more geographically separate sites, each equipped with its own storage. In the event that one of the sites fails, the other site acts as an active standby.

Oracle RAC on extended distance (stretched) clusters provides extremely fast recovery from a site failure and allows for all nodes, in all sites, to actively process transactions as part of a single database cluster.

Both Oracle ASM and the Oracle Database stack, in general, are designed to use enterprise-class shared storage in a data center. Fibre channel technology, however, enables distribution of compute and storage resources across two or more data centers, and connection of the two hrough ethernet cables and fibre channel, for compute and storage needs respectively.

The high impact of latency, and therefore distance, creates some practical limitations as to where this architecture can be deployed. An active/active Oracle RAC architecture fits best where the two datacenters are located relatively close (<100km) and where the costs of setting up a low latency and dedicated direct connectivity between the sites for Oracle RAC has already taken place, which is why it cannot be used as a replacement for a disaster recovery solution such as Oracle Data Guard or Oracle GoldenGate.

Learn more about *extended Oracle RAC 19c*, with additional details *here*.

Oracle RAC on VMware vSphere Platform

The two key requirements of Oracle RAC are:

- Shared Storage
- Multicast Layer 2 Networking

These requirements are challenging for Oracle RAC on native AWS installations.

Support for shared storage and multicast layer 2 networking are built into the vSphere platform. With Oracle RAC on VMware vSphere and VMware vSANtraditional or extended Cluster, these requirements are natively addressed.

vSphere HA clusters enable a collection of ESXi hosts to work together so that, as a group, they provide higher levels of infrastructure level availability for VMs than each ESXi host can provide individually.

vSphere HA provides high availability for VMs by pooling the VMs and the hosts they reside on into a cluster. Hosts in the cluster are monitored and in the event of a failure, the VMs on a failed host are restarted on alternate hosts.

When a vSphere HA cluster is created, a single host is automatically elected as the master host. The master host communicates with vCenter Server and monitors the state of all protected VMs and of the slave hosts.

Oracle RAC and VMware HA solutions are completely complementary to each other. Running Oracle RAC on the VMware platform provides application-level HA that Oracle RAC enables in addition to the infrastructure-level HA that VMware vSphere platform provides.

Learn more about *VMware vSphere HA*.

Extended Oracle RAC on VMware vSphere Platform

Many business-critical applications require five 9s of availability, or 99.999% availability (less than five minutes of downtime per year). This is where the marriage of vSphere HA and Oracle RAC really shines. This combination has been used to great effect by several very large organizations globally.

Extended Oracle RAC provides greater availability than local Oracle RAC. It provides extremely fast recovery from a site failure and enables all servers, in all sites, to actively process transactions as part of a single database cluster.

Running Extended Oracle RAC on a stretched cluster architecture provides the same advantages as traditional Oracle RAC, across data centers and sites, in addition to the site-level protection enabled by stretched cluster architecture.

Both traditional Oracle RAC and extended Oracle RAC require:

- Layer 2 Network Adjacency
- Shared storage

In addition to the above requirements, the following must also be considered when deploying an extended Oracle RAC:

- Latency requirements of the workload
  - Site distance (0, 25, 50, 100, > 100KM?)
  - Network latency between RAC nodes across sites increases and should be kept at five milliseconds (ms) or less
  - Network connection/bandwidth between Sites? Dark fiber over dense wavelength division multiplexing (DWDM)
- Storage that is synchronously, bi-directionally replicated in 5ms or less is required at each site
- A witness or quorum site is required for any clustered app or storage to avoid a *split brain*

Network Latency

The high impact of latency, and therefore distance, creates some practical limitations as to where this architecture can be deployed. An active/active Oracle RAC architecture fits best where the two datacenters are located relatively close (<100km) and where the costs of setting up low latency and dedicated direct connectivity between the sites for Oracle RAC has already taken place. For these reasons, it cannot be used as a replacement for a disaster recovery solution such as Oracle Data Guard or Oracle GoldenGate.

In general, for this to function properly across the broadest range of clusters and types, there must be at least 5ms or less round-trip latency (RTT) between the cluster nodes (whether they are in the same or different data center or not). This is true of Oracle RAC and vSphere HA clusters (vSphere Enterprise Plus licenses support up to 10 ms).

Storage Replication

For an extended distance cluster, storage should be synchronously and bi-directionally replicated between sites. There are three ways to do this:

- **Host-based replication**: the most common example of this is Oracle ASM. **Learn more about** *host-based replication*.
- **Appliance-based replication**: examples include EMC VPLEX, IBM SVC, HP Peer Persistence, NetApp Metro Cluster, and Pure Storage ActiveCluster.
- **VMware vSAN Stretched Cluster**: inherently provides this storage solution suitable for extended Oracle RAC using the vSAN fault-domain concept.

Host-Based Storage Virtualization (ASM)	Vendor-Based Storage Virtualization
Storage agnostic	Not storage agnostic as specific to a vendor (i.e., tightly integrated with the underlying storage)
Bundled with Oracle and easy to set up	Setup takes time and is relatively expensive
CPU performance hit/extra CPU usage	No CPU penalty as storage virtualization is handled by the underlying storage unit
Third site required for voting disk	Requires third site as witness for arbitration in case of site failure

TABLE 13. COMPARISON OF HOST-BASED REPLICATION AND APPLIANCE-BASED REPLICATION

VMware vSAN Stretched Cluster-Based Native Storage Virtualization	Vendor-Based Storage Virtualization
Storage virtualization is native to the VMware platform	Storage virtualization specific to a vendor (i.e., tightly integrated with the underlying storage)
vSAN Stretched Cluster offers ease of deployment and easy enablement, without additional software or hardware. Integrates well with other VMware features like VMware vSphere vMotion and vSphere HA.	Setup takes time and needs additional software and hardware
Cost-effective server SAN solution for extended distance with easy setup	Setup takes time and is relatively expensive
Scale-out architecture with resources (storage and compute) balanced across both sites.	More vendor equipment needed and is expensive

Reduced consumption of Oracle cluster node CPU cycles associated with host-based mirroring. Instead, vSAN takes care of replicating the data across sites.	No CPU penalty as storage virtualization is handled by the underlying storage unit
Easy to deploy the preconfigured witness appliance provided by VMware and can be used as vSAN Stretched Cluster witness at the third site	Third site required as witness for arbitration in case of site failure; more vendor equipment needed and is expensive
Elimination of Oracle Server and Clusterware at the third site	Elimination of Oracle Server and Clusterware at the third site

TABLE 14. COMPARISON OF VMWARE VSAN STRETCHED CLUSTER REPLICATION AND APPLIANCE-BASED REPLICATION

Witness Site

All clustered technologies require a tiebreaker (i.e., a witness or quorum disk) to prevent a cluster-fencing or split brain situation where every node (due to network or disk heartbeat failure with other nodes in the cluster) assumes that it is the sole surviving member of the cluster, thereby proclaiming itself to be the master. This is referred to as split-brain syndrome.

Split brain syndrome causes the data and application at each site be out of sync and unable to be resynchronized. The occurrence of the witness site prevents creating multiple masters and avoids this scenario.

There are two general implementations of VMware stretched cluster architectures:

- VMware vSphere Metro Storage Cluster
- VMware vSAN Stretched Cluster

Learn more about *Extended Oracle RAC on VMware vSAN Stretched Cluster*.

Solution Configuration

This section introduces the resources and configurations for the solution including:

- Architecture diagram
- Hardware resources
- Software resources
- Network configuration
- Storage Configuration

Architecture Diagram

This reference architecture has been created on VMware vSphere 6.7 platform. The steps to set up a traditional or extended Oracle RAC is exactly the same on VMware vSphere 7.0 and above.

There are two components to the solution architecture:

- On-premises vSphere cluster on site A
- On-premises vSphere cluster on site B

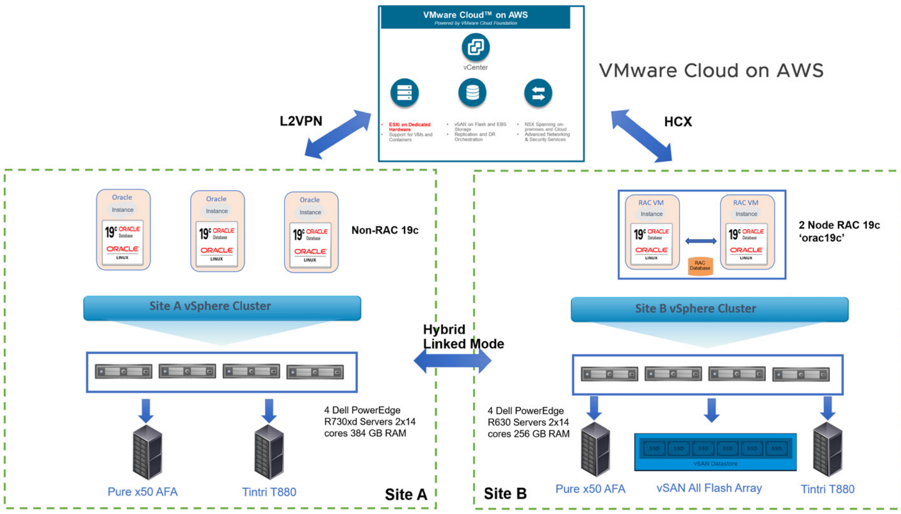


FIGURE 16. ORACLE ON VMWARE CLOUD FOR AWS SOLUTION ARCHITECTURE

The on-premises setup has a two-site separate, dedicated vSphere cluster configuration, Site A and Site B.

- Site A is hosting non-Production RAC and Non-RAC workloads including DR workloads
- Site B is hosting Production RAC and Non-RAC workloads
- Both Site A and Site B are in Hybrid Linked mode.

Site A infrastructure details:

- A four-node vSphere 6.7 cluster, ESXi patch level 14320388
- Each ESXi server is a Dell PowerEdge R730xd rack server with two sockets, 14 cores each with Intel Xeon processor E5-2695 v3 at 2.30GHz with hyperthreading technology and 384GB of RAM
- Each ESXi server features access to two storage units:
  - Pure x50 all-flash storage (Purity/FA 5.3.2) for both block FC storage and vVols
  - Tintri T880 all-flash for NFS
- Each ESXi server features:
  - (2) QLogic 8Gb FC host bus adapters for FC storage
  - (2) 10Gbit connections to Tintri T880 NFS storage

Site B infrastructure details:

- A four-node vSphere 6.7 cluster, ESXi patch level 15160138
- Each ESXi server is a Dell PowerEdge R630 server, Intel Xeon CPU E5-2680 v4 at 2.40GHz, (2) 14 cores, 256GB RAM with hyperthreading
- Each ESXi server features access to three storage units:
  - Pure x50 all-flash storage (Purity/FA 5.3.2) for both block FC storage and vVols
  - Tintri T880 all-flash for NFS
  - vSAN 6.7 all-flash array (AFA) for hyperconverged storage
- Each ESXi server had
  - (2) QLogic 8Gb FC host bus adapters for FC storage
  - (2) 10Gbit connections to Tintri T880 NFS storage and vSAN traffic

- Each ESXi server features two vSAN flash SSD disk groups
- Each vSAN disk group features (1) 800GB SSD for cache and (1) 2TB SSD for capacity

The Stretched Clusters for VMware Cloud on AWS setup has the following configuration:

- A six-node stretched cluster for VMware Cloud on AWS setup across two AZs
  - Three servers in AZ **us-west-2b**
  - Three servers in AZ **us-west-2c**
- Each ESXi server is an Amazon EC2 i3p.16xlarge with two sockets, 18 cores each with Intel Xeon processor E5-2686 v4 at 2.30GHz and 512GB RAM memory
- Storage provided by the HCI vSAN instance

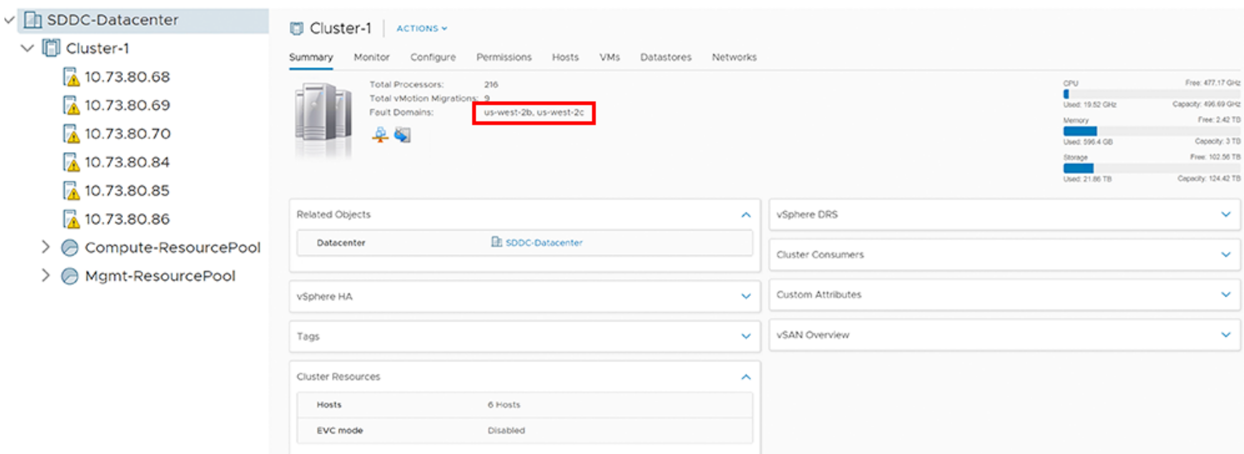


FIGURE 17. STRETCHED CLUSTERS FOR VMWARE CLOUD ON AWS SETUP

An L2VPN is used to extend the on-premises data center on site A to VMware Cloud on AWS to rapidly and easily migrate application workloads from on-premises to the VMware Cloud on AWS and back. This is possible because there are no networking changes with L2VPN capability between on-premises SDDC and VMware Cloud on AWS.

Hardware Resources

The following hardware resources are utilized for the vSphere cluster on site A:

Description	Specification
Server	(4) ESXi server
Server Model	Dell PowerEdge R730xd rack server
CPU	Two sockets, 14 cores each, Intel Xeon processor E5-2695 v3 at 2.30GHz with hyperthreading enabled
RAM	384GB RAM

Storage controller	(2) Emulex LPe12000 8Gb PCIe fibre channel adapter
Storage Array	Pure x50 AFA (Purity/FA 5.3.2) and Tintri T880 AFA
Network	(2) 10Gbit connections to Tintri T880 NFS storage

TABLE 15. SITE A VSPHERE CLUSTER HARDWARE RESOURCES

The following is a summary of one ESXi server in the vSphere cluster on site A:

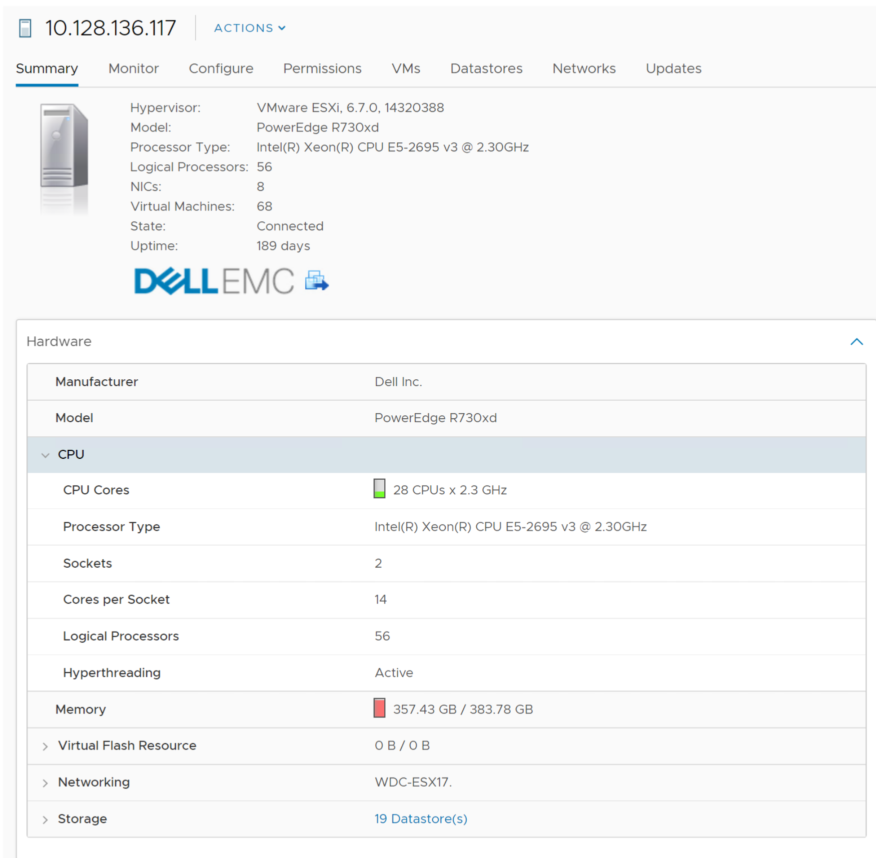


FIGURE 18. VMWARE ESXI SITE A SERVER SUMMARY

The following hardware resources are utilized for the vSphere cluster on site B:

Description	Specification
Server	(4) ESXi server
Server Model	Dell PowerEdge R630

CPU	Two sockets with 14 cores each, Intel Xeon CPU E5-2680 v4 at 2.40GHz, 2x14 cores with hyperthreading enabled
RAM	256GB RAM
Storage controller	(2) 8Gb ISP2532-based fibre channel to PCIe HBA
Storage Array	Pure x50 AFA (Purity/FA 5.3.2), Tintri T880 AFA and vSAN 6.7 AFA
vSAN Storage Controller	Dell HBA330 Mini
vSAN AFF Disks	Each server has two vSAN flash SSD disk groups, each vSAN disk group has (1) 800GB SSD for cache and (1) 2TB SSD for capacity
Storage Network	(2) QLogic 8Gb FC host bus sdapters for FC storage
Network	(2) 10Gbit connections to Tintri T880 NFS storage and vSAN traffic

TABLE 16. SITE B VSPHERE CLUSTER HARDWARE RESOURCES

The following is a summary of one ESXi server in the vSphere cluster on site B:

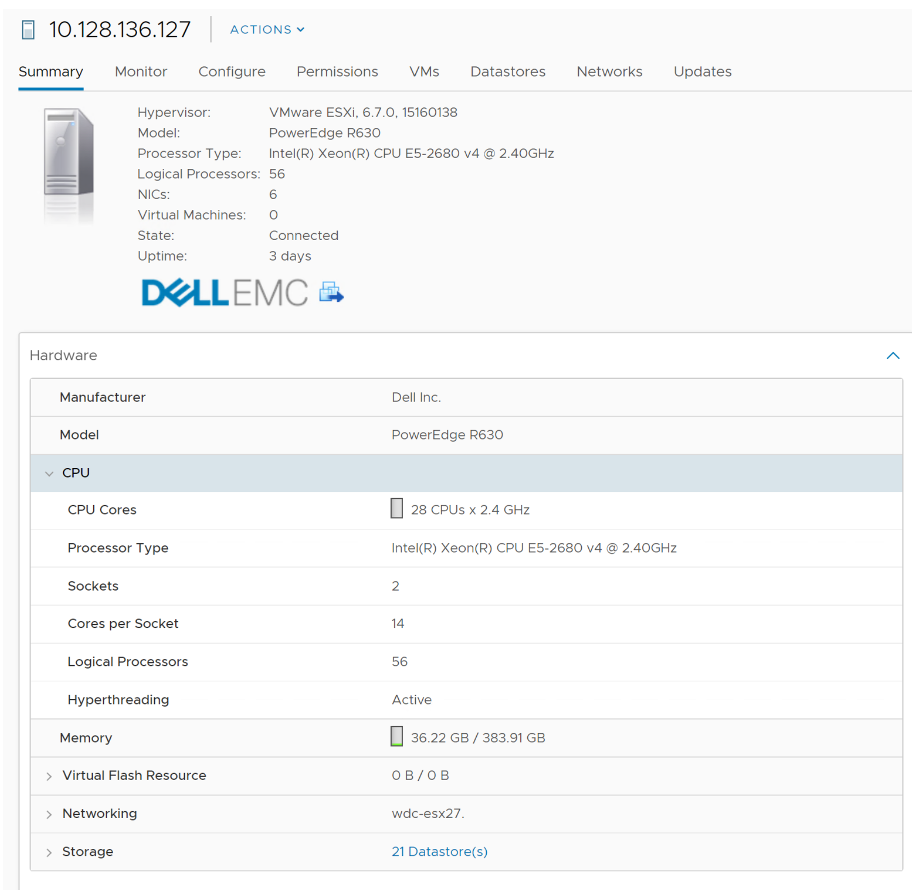


FIGURE 19. VMWARE ESXI SITE B SERVER SUMMARY

The following hardware resources are utilized for Stretched Clusters for VMware Cloud on AWS:

Description	Specification
Server	(6) ESXi servers
Server model	Amazon EC2 i3p.(16) large Server
CPU	Two sockets, 18 cores each, Intel Xeon processor E5-2686 v4 at 2.30GHz
RAM	512GB
Disks	(8) NVMe drives, each drive 1.73TB across two vSAN disk groups
vSAN disk groups	Two disk groups, each disk group with (1) NVMe for cache and (3) NVMe for capacity
Network	25G Amazon Elastic Network Adapter (ENA)

TABLE 17. STRETCHED CLUSTERS FOR VMWARE CLOUD ON AWS HARDWARE RESOURCES

The stretched cluster for VMware Cloud on AWS features six ESXi servers across two fault domains or AZs for site-level HA with three ESXi servers in each AZ.

vSAN fault domains are configured to inform vSphere and vCenter which hosts reside in which AZs. Each fault domain is named after the AZ it resides within to increase clarity.

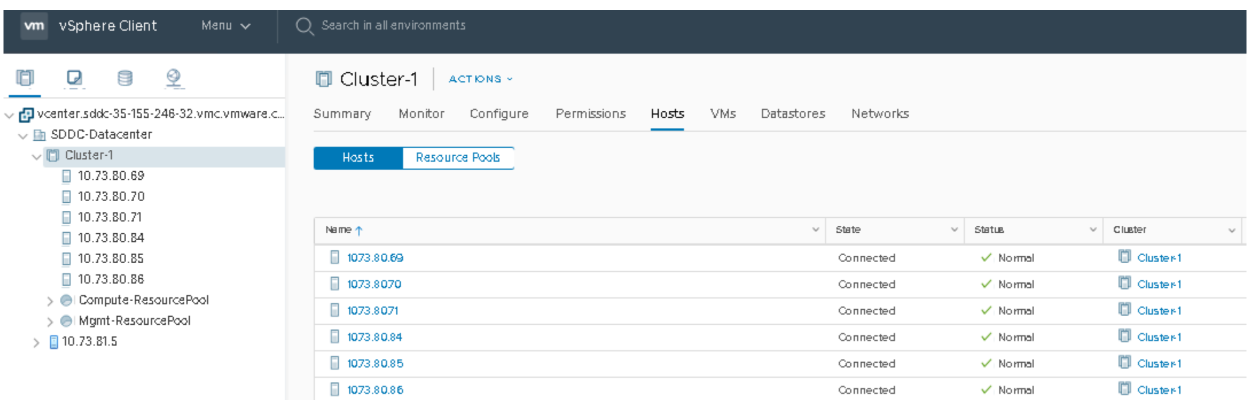


FIGURE 20. FAULT DOMAIN CONFIGURATION

Fault domains/AZs are **us-west-2b** and **us-west-2c** with three ESXi servers in each AZ.

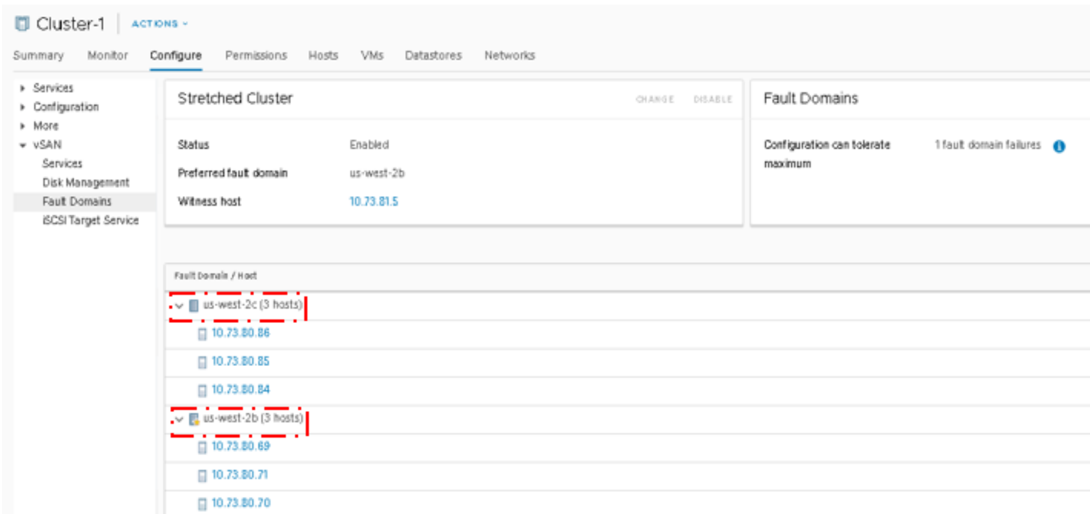


FIGURE 21. FAULT DOMAIN NAMING BY AZ

To protect against split-brain scenarios and help measure site health, a managed vSAN witness is also created in a third AZ. The third AZ is picked at random from the remaining AZs.

The witness has been engineered to run on an EC2 M5.XL AMI to reduce the cost to the customer.

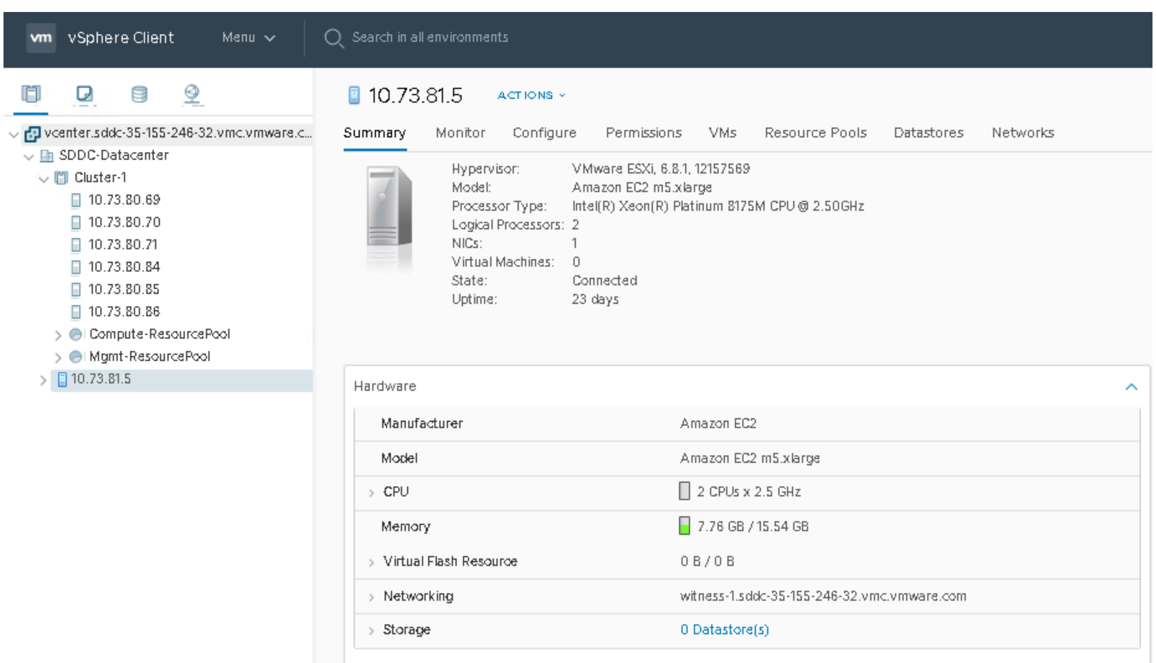


FIGURE 22. MANAGED VSAN WITNESS

The following is a summary of one ESXi server in a Stretched Clusters for VMware Cloud on AWS environment:

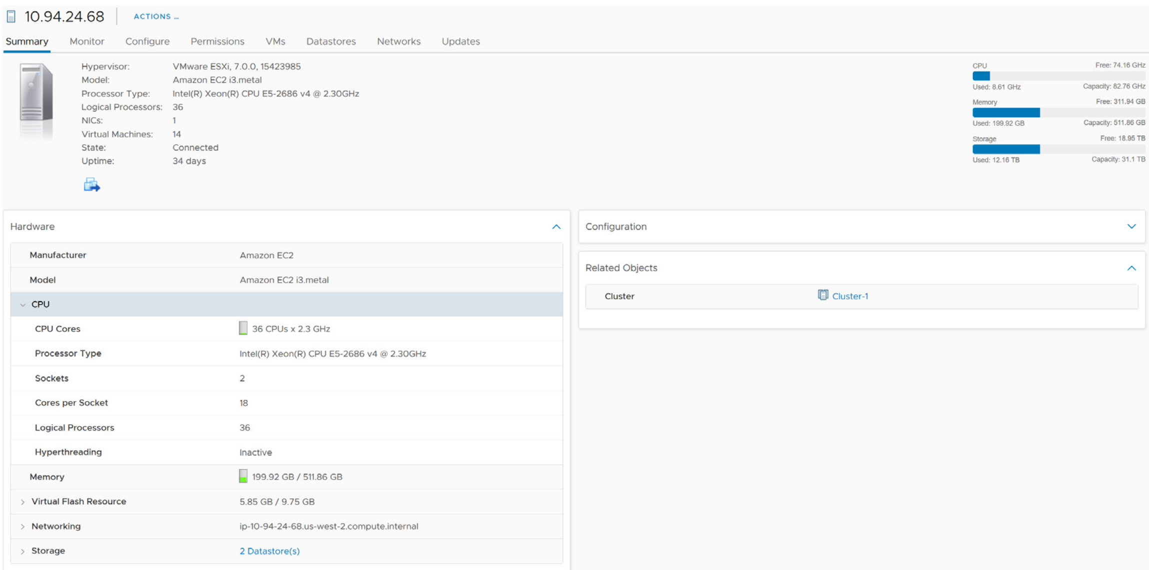


FIGURE 23. ESXI SERVER IN STRETCHED CLUSTERS FOR VMWARE CLOUD ON AWS

Software Resources

Software resources for the solution are as follows:

Software	Version	Purpose
VMware vCenter Server and ESXi	6.7 Patch 15160138	ESXi cluster to host VMs, VMware vCenter Server provides a centralized platform for managing VMware vSphere environments
ESXi Datastores		Pure AFA provides VMFS and vVol datastores, Tintri T880 provides NFS datastores and vSAN AFA provides Hyperconverged storage
Oracle Linux	7.6	Oracle database server nodes
Oracle Database 19c	19.3.0.0.0	Oracle database
Oracle Workload Generator for OLTP	SLOB 2.5.2.3	To generate Oracle workload

TABLE 18. SOFTWARE RESOURCES

Network Configuration

VMware vSphere® Distributed Switch™ (VDS) acts as a single virtual switch across all associated hosts in the data center. This setup enables VMs to maintain a consistent network configuration as they migrate across multiple hosts.

A port group defines properties regarding security, traffic shaping, and network adapter teaming. Jumbo frames (MTU=9000 bytes) were enabled on the vSphere vMotion interface and the default port group setting was used.

For site A, VDS **SiteA-10g-dVS** uses (4) 10GbE adapters per host:

- (2) 10GbE uplinks for VM traffic
- (2) 10GbE uplinks for VMkernel non-VM traffic

The following distributed switch port groups were created for Oracle VM traffic to balance traffic across the available uplinks:

- Port group **DPortGroup-1363** with VLAN ID 1363 is for VM user traffic
- Port group **DPortGroup-VLAN70-NFS** with VLAN ID 70 for NFS traffic

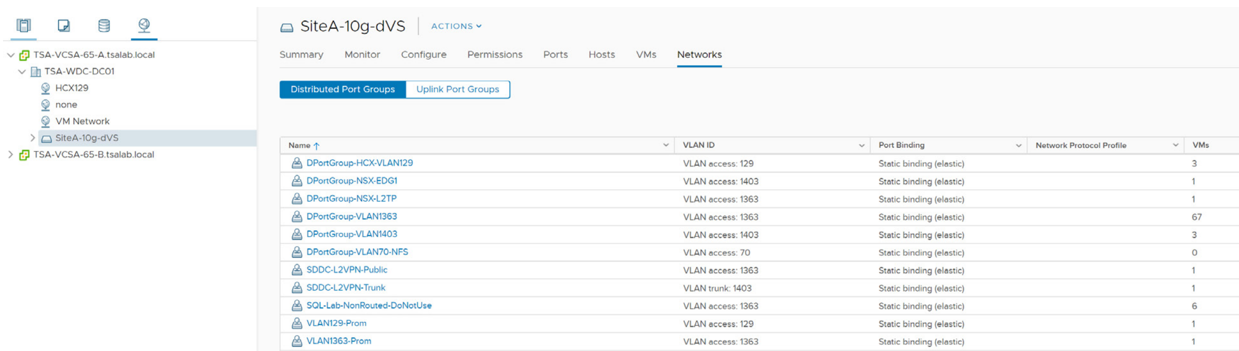


FIGURE 24. SITE A VSPHERE DISTRIBUTED SWITCH PORT GROUP CONFIGURATION

As mentioned earlier, site B is hosting production Oracle RAC and Non-Oracle RAC workloads.

For site B, VDS **SiteB-10g-dVS** uses (2) 10GbE adapter and (2) 1GbE adapter per host:

1. (2) 10GbE uplinks for VM traffic
2. (2) 1GbE uplinks for VMkernel non-VM traffic

The following distributed switch port groups were created for Oracle RAC and Oracle VM traffic to balance traffic across the available uplinks:

- Port group **DPortGroup-1403** with VLAN ID 1403 is for VM user traffic
- Port group **DPortGroup-VLAN70-NFS** with VLAN ID 70 for NFS traffic
- Port group **DPortGroup-OraclePrivate** with VLAN ID 72 is for Oracle RAC interconnect traffic with two active/active uplinks set to **Route based on originating virtual port**.
- Port group **DPortGroup-VLAN69-VSAN** with VLAN ID 69 for vSAN traffic

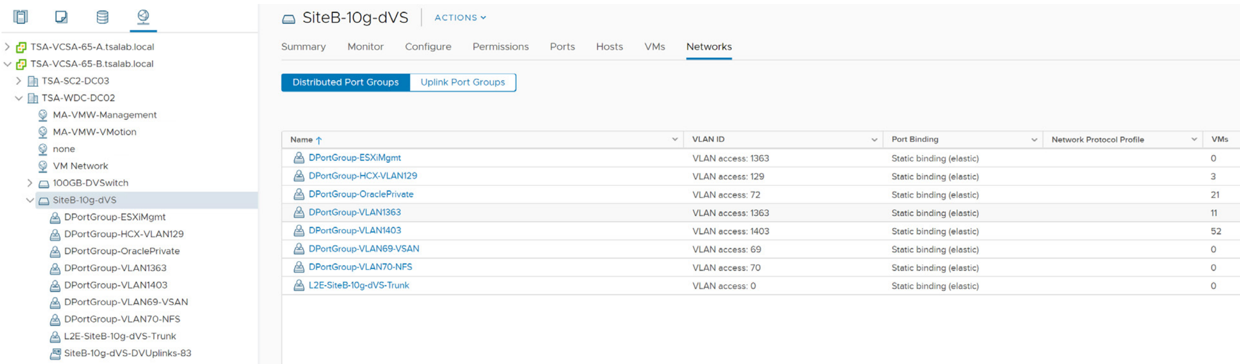


FIGURE 25. SITE B VSPHERE DISTRIBUTED SWITCH PORT GROUP CONFIGURATION

For VMware Cloud on AWS, each ESXi server has (1) 25GbE adapter per host as a visual representation of a pair of physical adapters.

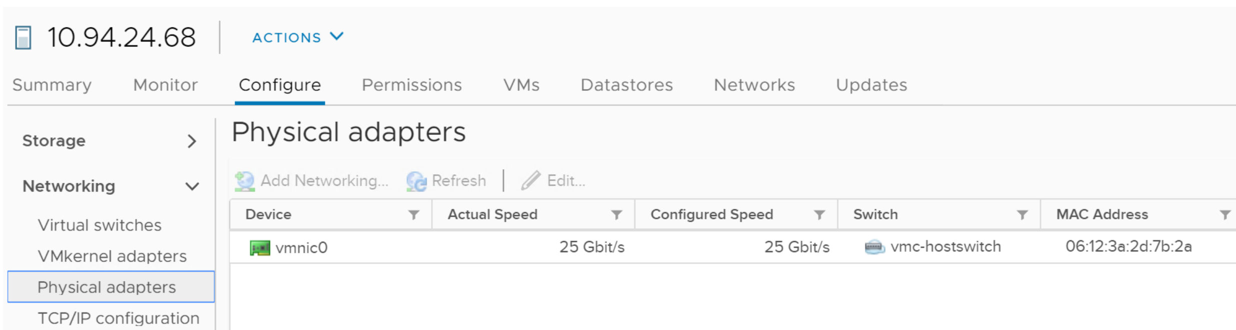


FIGURE 26. VMWARE CLOUD ON AWS PHYSICAL ADAPTER CONFIGURATION

To create a logical segment, navigate to the VMware Cloud on AWS Portal and click **Networking & Security**. Click **Segments**, then **Add Segments**.

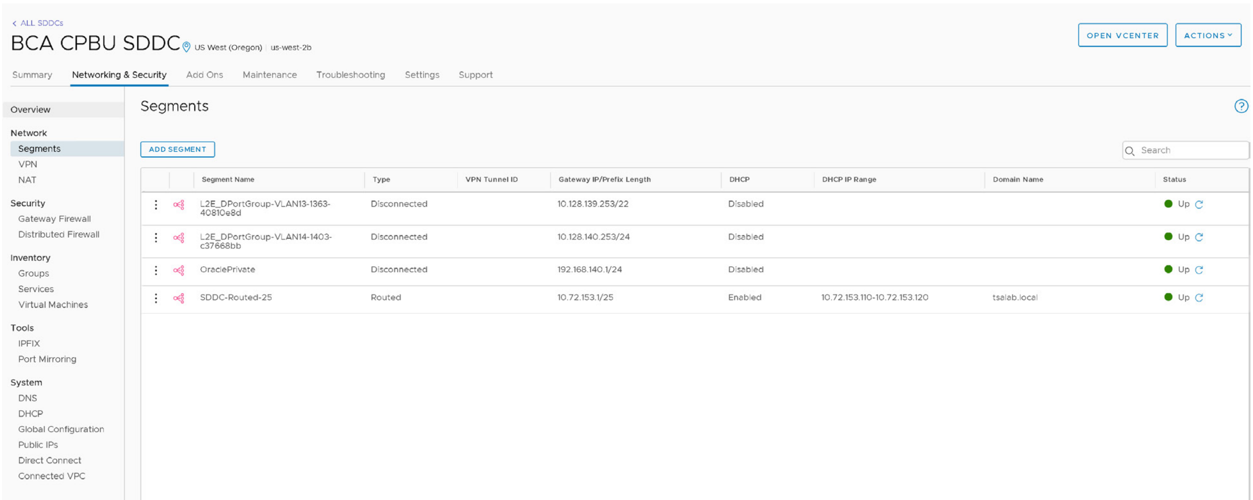


FIGURE 27. LOGICAL NETWORK DETAILS

Fill in the required details as shown above. Select the **Disconnected** option and specify the CIDR block of the segment in the **Gateway/Prefix Length** field. Click **Save** when done.

As mentioned before, a disconnected network segment has no uplink and provides an isolated network accessible only to VMs connected to it.

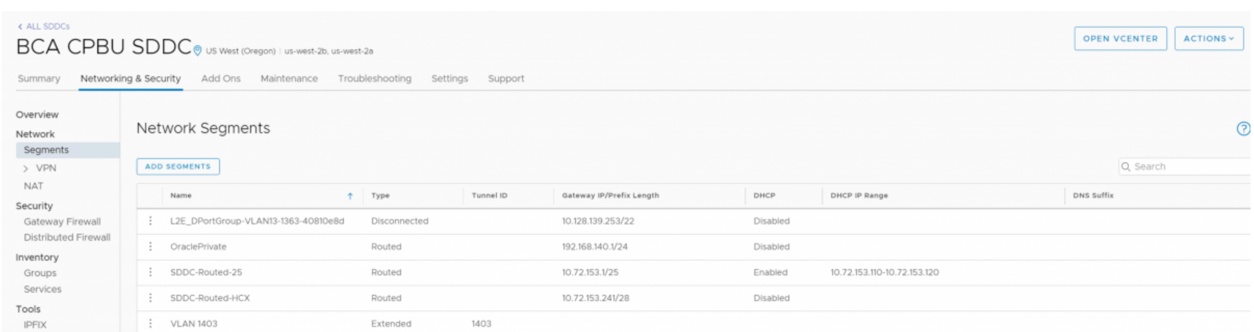


FIGURE 28. LOGICAL NETWORK ORACLEPRIVATE

Learn more about *VMware Cloud on AWS logical networks*.

The following extended segments were created for Oracle VM traffic between on-premises site A and VMware Cloud on AWS:

- Port group **BCA-L2VPN** for **L2VPN for VM user**: traffic enables VMs to keep the same subnet when migrating from on-premises data centers to the cloud and back.
- Port group **BCA-VPN-Network** for routed VM: traffic enables VMs to communicate—or ping each other—without being on the same subnet.

vSphere vMotion enables live migration of running—that is, powered on—VMs from an on-premises host to a host in VMware Cloud on AWS, with zero downtime for the application (less than one second switchover time), continuous service availability, and complete transaction integrity. Furthermore, by enabling certain advanced configurations, vSphere vMotion migration between on-premises VMs and VMware Cloud on AWS can be enabled across various VDS versions.

VMware Cloud on AWS provides multiple ways to establish network connectivity from on-premises environments, including different types of VPNs and AWS Direct Connect (DX). AWS DX is a service provided by AWS that allows creation of a high-speed, low latency connection between an on-premises data center and AWS services including VMware Cloud on AWS.

Learn more about *AWS Direct Connect*.

Learn more about *live vSphere vMotion migration between on-premises data centers and VMware Cloud on AWS*.

Storage Configuration

Site A features access to two storage units

- Pure x50 all-flash storage (Purity/FA 5.3.2) for both block FC storage and vVols
- Tintri T880 all-flash for NFS

Site B features access to three storage units

- Pure x50 all-flash storage (Purity/FA 5.3.2) for both block FC storage and vVols
- Tintri T880 all-flash for NFS
- vSAN 6.7 AFA for hyperconverged storage

On Site A, each of the four ESXi servers feature (2) Emulex LPe12000 8Gb PCIe fibre channel adapters for both Pure Storage backed FC block storage and vVols.

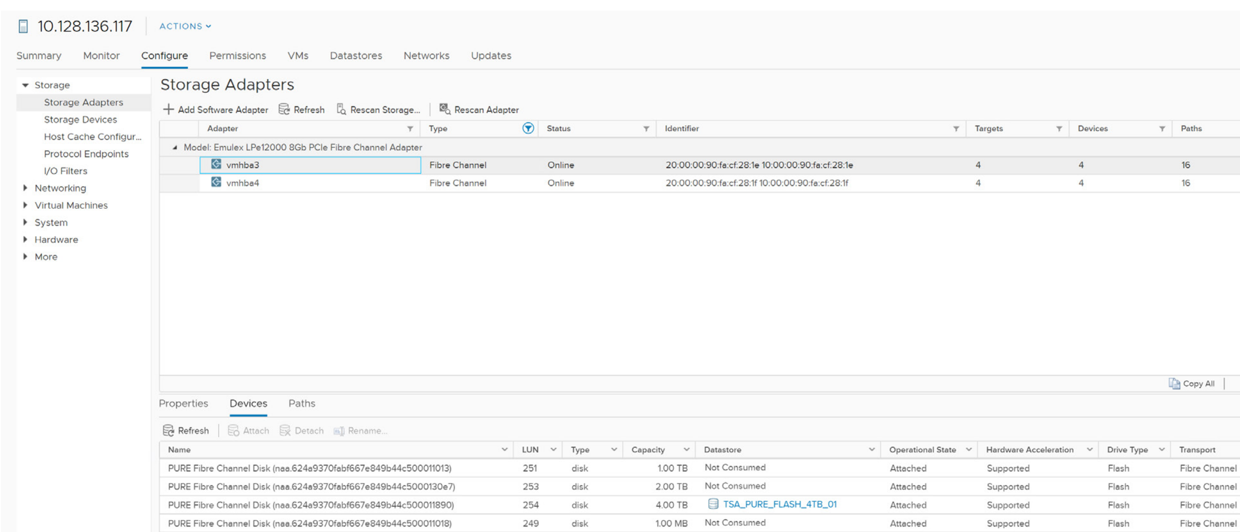


FIGURE 29. ESXI SERVER FC STORAGE CONNECTIONS

On Site A, each of the four ESXi servers had (2) 10Gbit connections to Tintri T880 NFS storage as shown below:

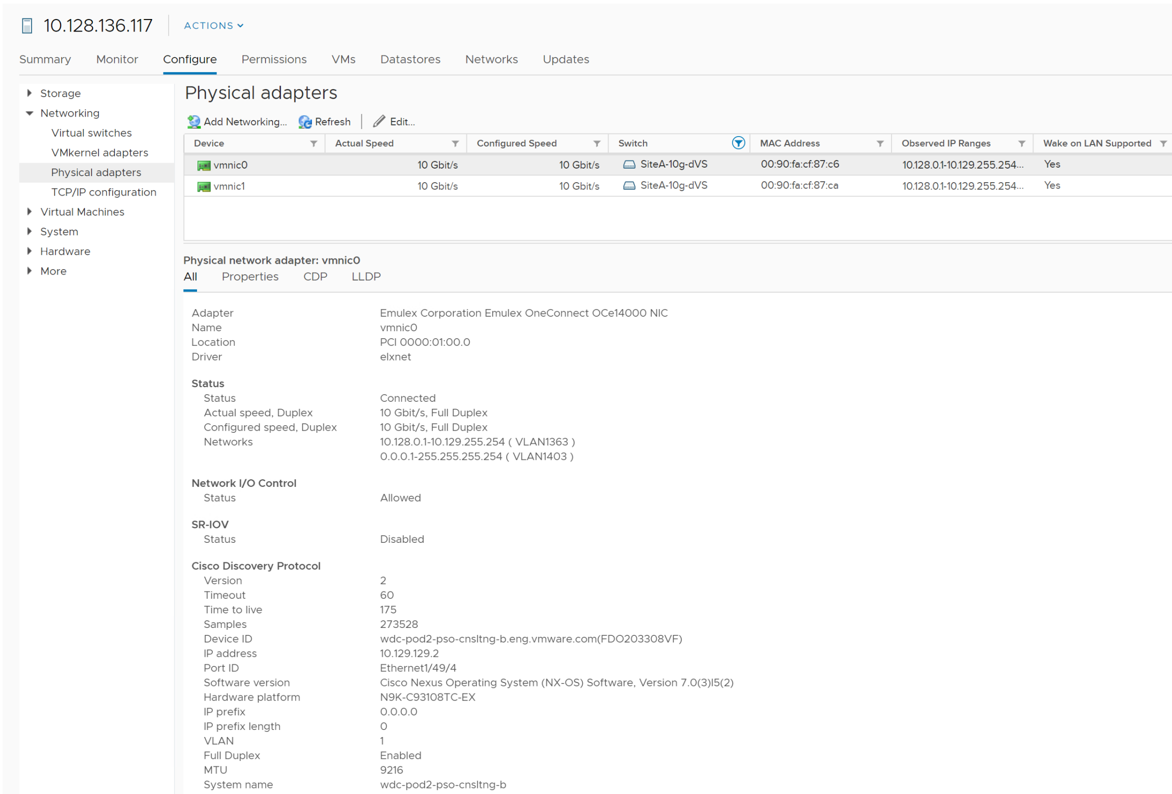


FIGURE 30. ESXI SERVER NETWORK CONNECTION DETAILS

The figure below shows FC VMFS and NFS datastores in Site A:

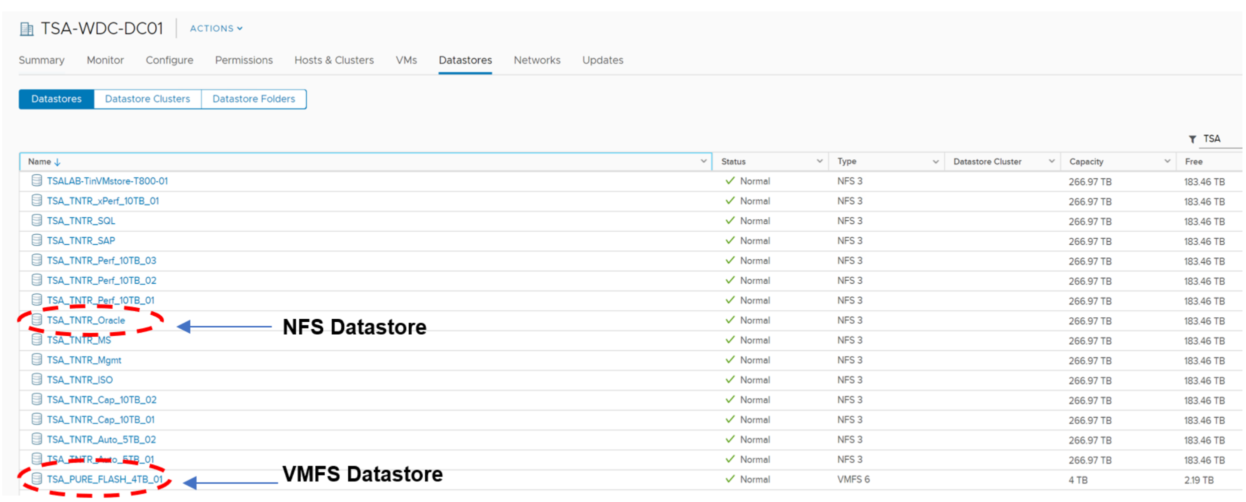


FIGURE 31. ESXI NFS AND VMFS DATASTORES

On Site B, each of the four ESXi servers feature (2) QLogic 8Gb FC host bus adapters for both block FC storage and vVols.

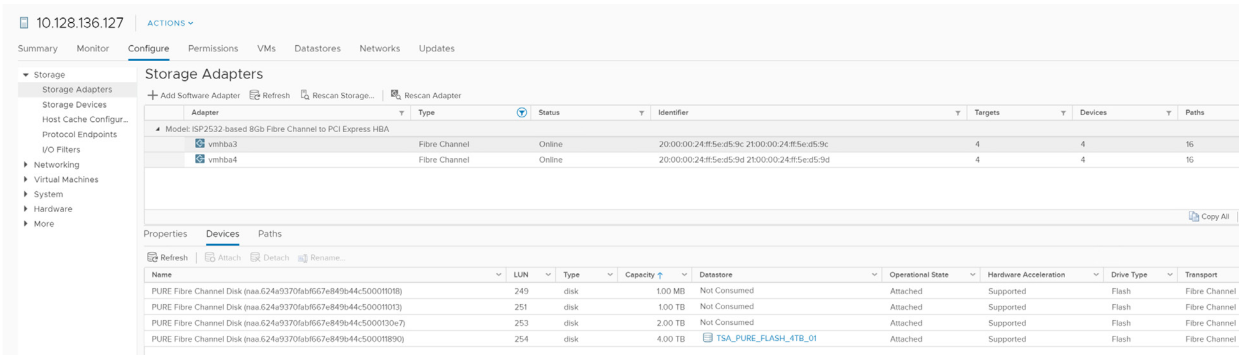


FIGURE 32. ESXI SERVER FC STORAGE CONNECTIONS

On Site B, each of the four ESXi servers feature (2) 10Gbit connections to Tintri T880 NFS storage and vSAN traffic as shown below:

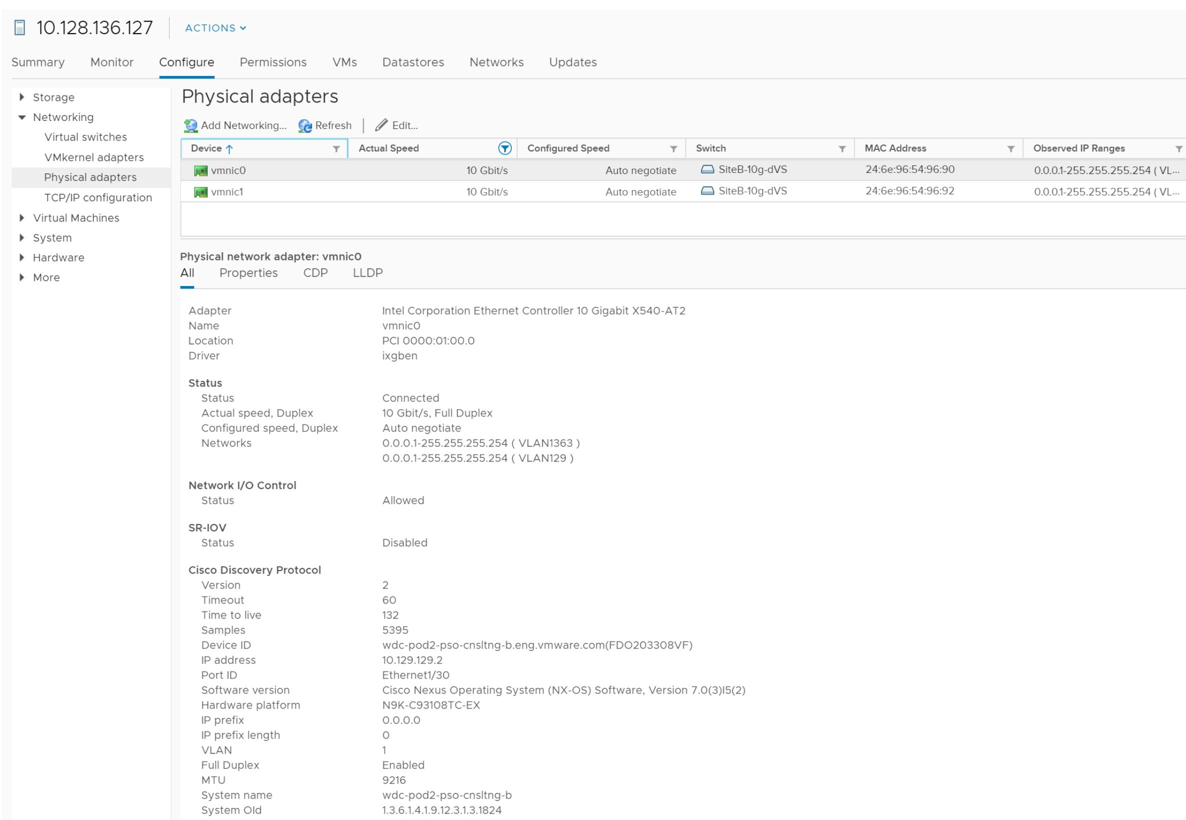


FIGURE 33. ESXI SERVER NETWORK CONNECTION DETAILS

On Site B, the following datastores are created on the four-node vSphere cluster:

- Pure x50 storage FC-backed VMFS datastore **TSA_PURE_FLASH_4TB_01**
- Pure x50 storage-backed vVol datastore **TSA_PURE_FLASH_VVOL**
- Tintri T880 storage-backed NFS (v3) datastore **TSA_TNTR_Oracle**
- vSAN AFA hyperconverged-backed datastore **vsanSiteB**

The figure below shows FC VMFS, vVol and NFS datastores in Site B:

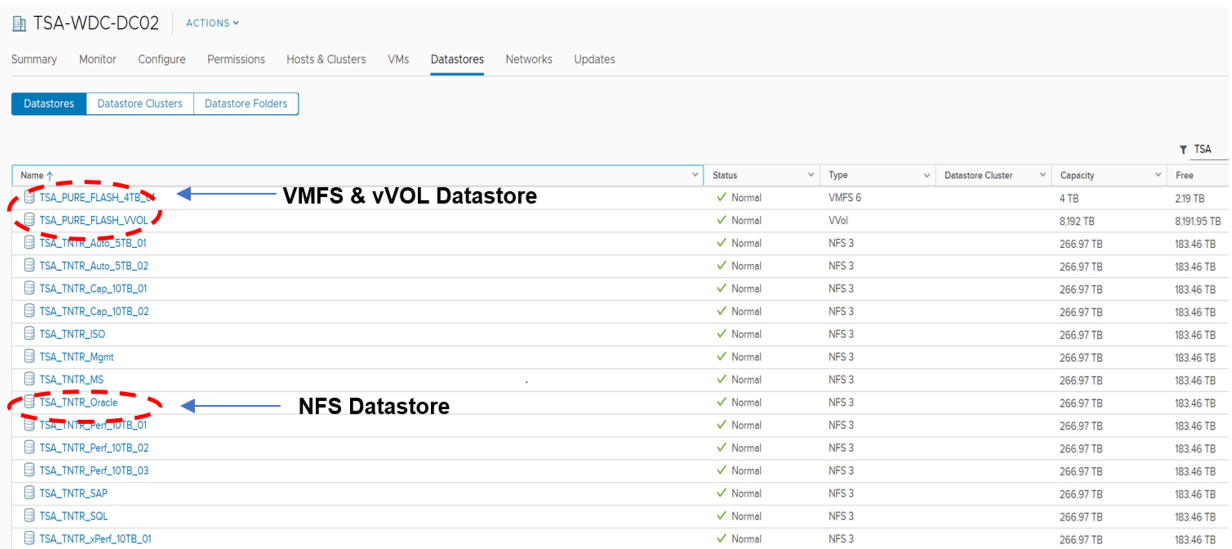


FIGURE 34. VSPHERE CLUSTER DATASTORES

The vSAN datastore in site B is shown below:

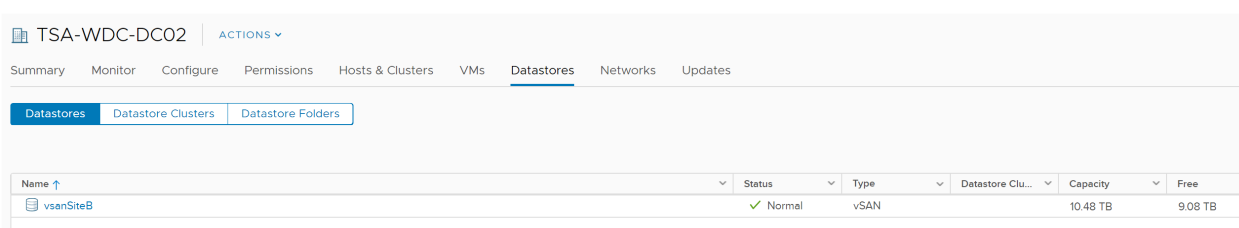


FIGURE 35. VSAN DATASTORE

Each ESXi server features (2) vSAN flash SSD disk groups. Each vSAN disk group has (1) 800GB SSD for cache and (1) 2TB SSD for capacity as shown below:

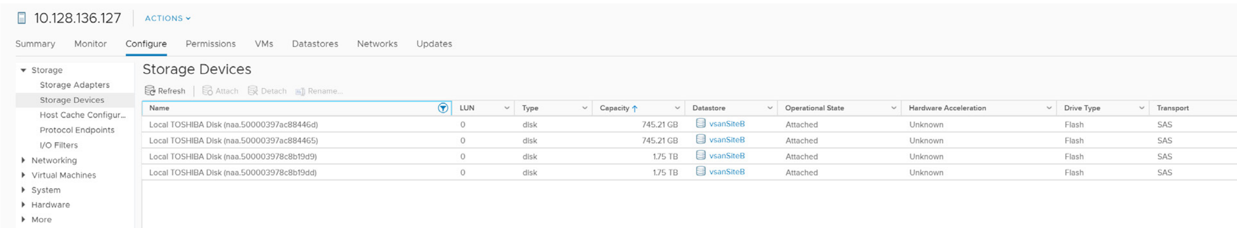


FIGURE 36. ESXI SERVER VSAN DISKS

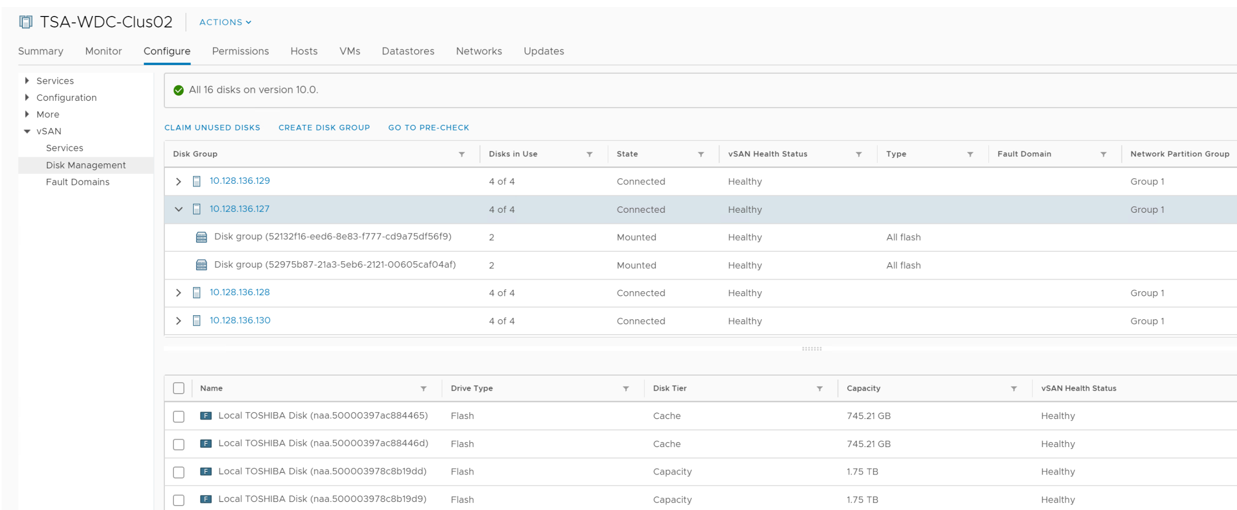


FIGURE 37. ESXI SERVER VSAN DISK GROUPS

For VMware vSAN:

- Prior to vSAN 6.7 Patch P01, the virtual disk must be EZT to enable multi-writer mode.
- Beginning with VMware vSAN 6.7 Patch P01 (ESXi 6.7 Patch Release ESXi670-201912001), Oracle RAC on vSAN does **not** require the shared VMDKs to be EZT (OSR=100) for multi-writer mode to be enabled.

Find further details in *KB 2121181* for VMware vSAN.

Below is an example of a storage policy for Oracle RAC on VMware vSAN prior to VMware vSAN 6.7 P01 (ESXi 6.7 Patch Release ESXi670-201912001).

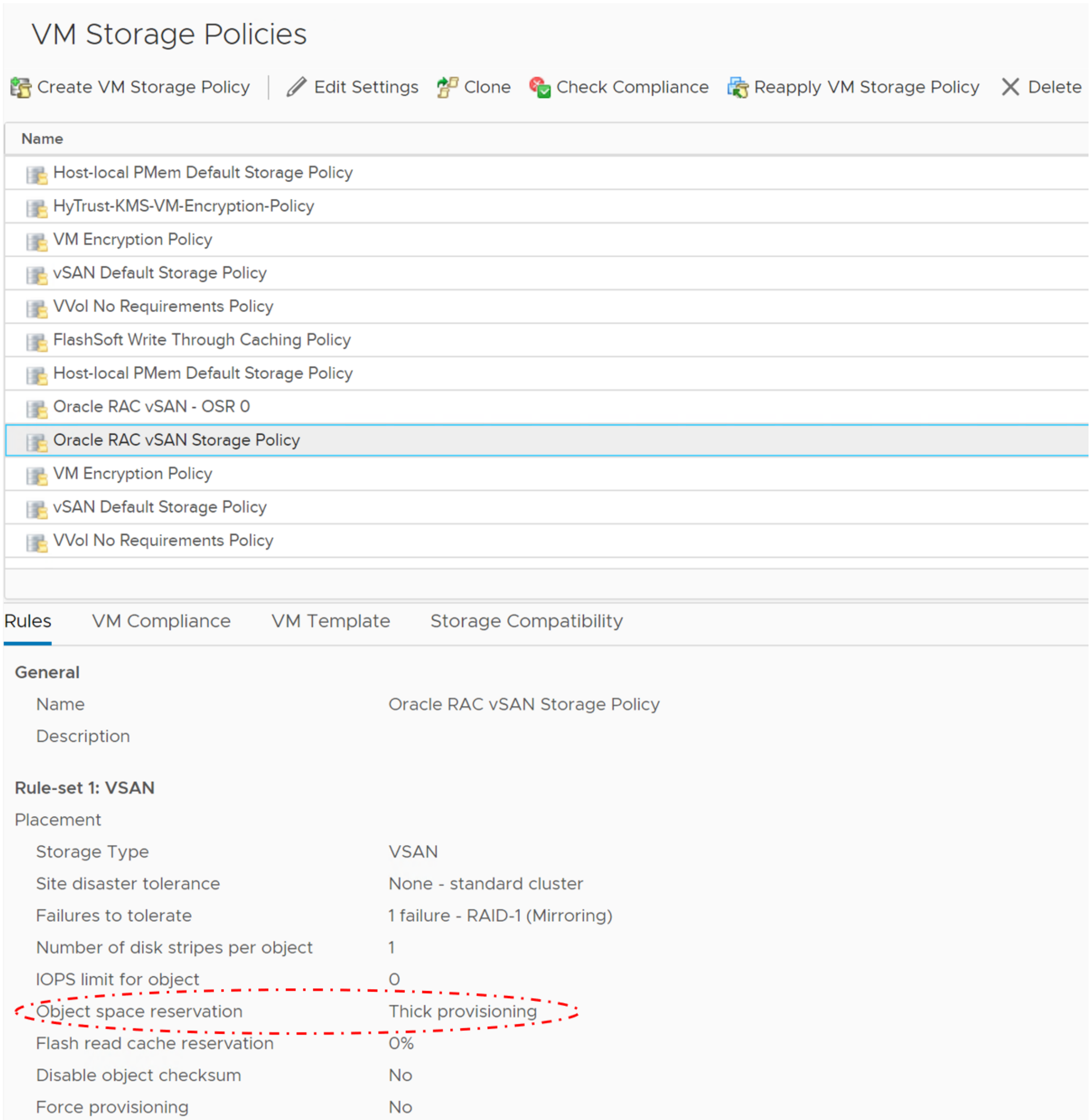


FIGURE 38. ORACLE RAC STORAGE POLICY PRIOR TO VSAN 6.7 P01

Below is an example of a storage policy for Oracle RAC on VMware vSAN beginning with VMware vSAN 6.7 P01 (ESXi 6.7 Patch Release ESXi670-201912001).

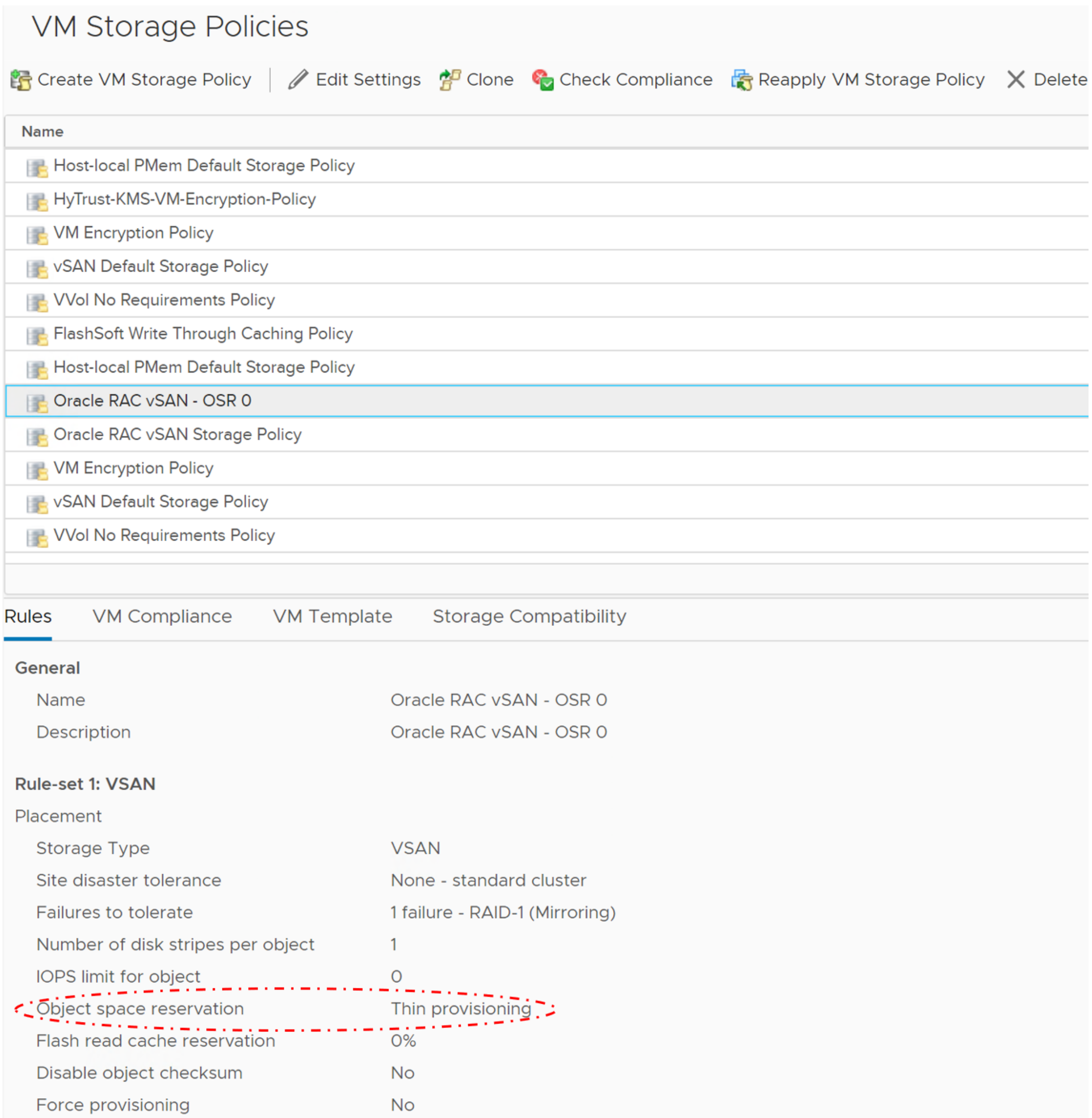


FIGURE 39. ORACLE RAC STORAGE POLICY BEGINNING WITH VSAN 6.7 P01

Find further details in *KB 2121181* for VMware vSAN.

In the case of VMware Cloud on AWS, which uses vSAN storage internally, all VMs running inside the cloud SDDC consume storage capacity and leverage storage services from the vSAN datastore. Management workloads and the workloads belonging to a single VMware Cloud on AWS cluster are located on the same vSAN cluster.

However, the cloud SDDC introduces a new vSAN capability that provides two logical datastores instead of one. One of these datastores (**vsanDatastore**) is used to store the management VMs. The other datastore (**WorkloadDatastore**) is used for the customer VMs. Both are part of the same physical datastore.

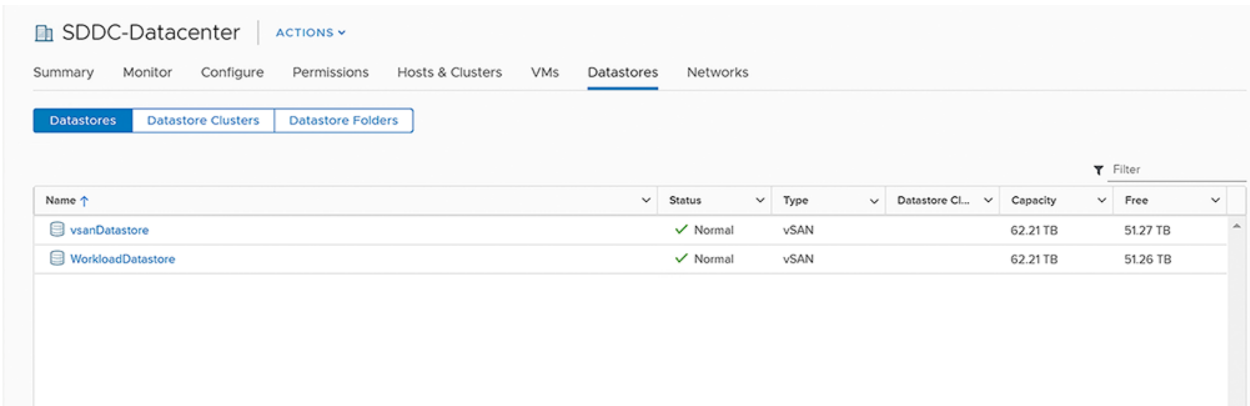


FIGURE 40. VSAN DATASTORES

VMware creates and operates a separate resource pool to manage customer workloads.

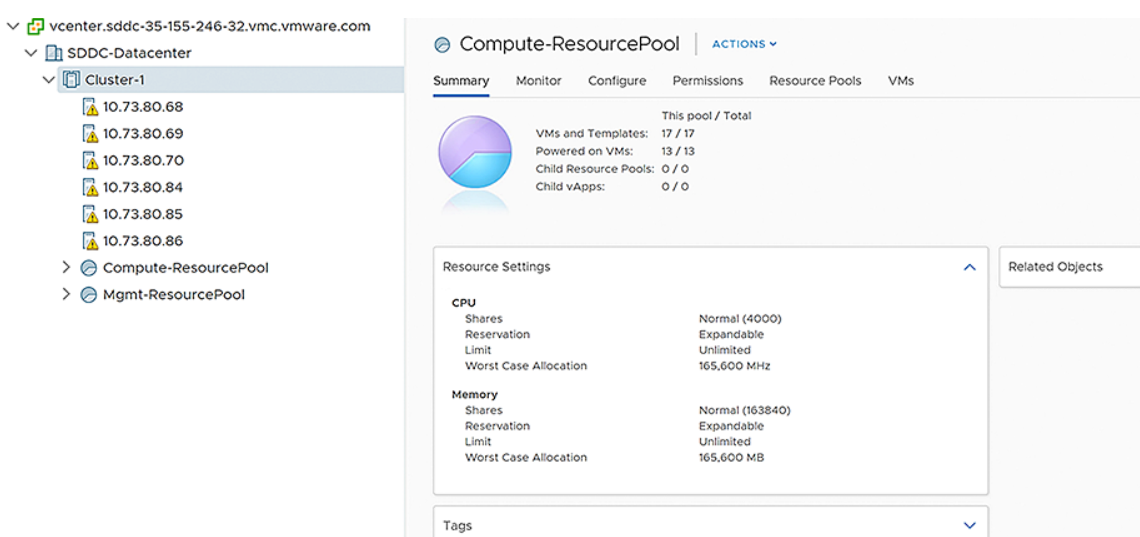


FIGURE 41. VMWARE COMPUTE RESOURCE POOL

Solution Validation

Site B is chosen as the site for all on-premises Oracle RAC deployments.

Site B is hosting production RAC and Non-RAC workloads. Site B is connected via VMware HCX and AWS Direct Connect to VMware Cloud on AWS.

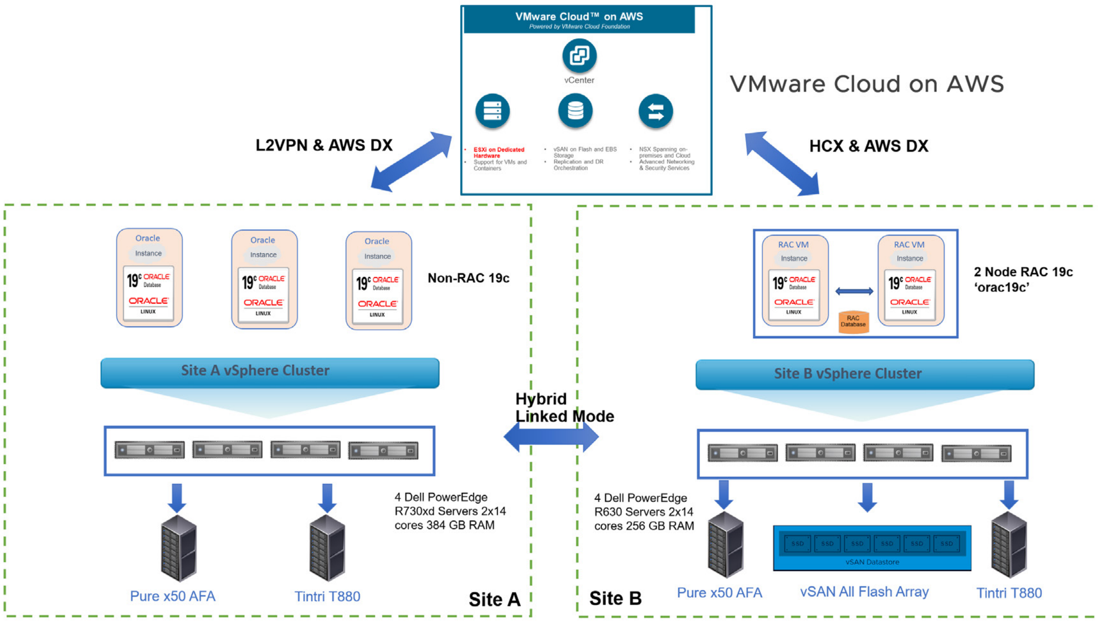


FIGURE 43. ORACLE WORKLOADS ON VMWARE CLOUD ON AWS

Different Oracle RAC clusters are deployed on the following storage platforms:

- FC-enabled VMFS datastore **TSA_PURE_FLASH_4TB_01** backed by Pure x50 Storage
- NFS (v3)-enabled datastore **TSA_TNTR_Oracle** backed by Tintri T880 Storage
- vSAN 6.7 AFA hyperconverged-enabled datastore **vsanSiteB**
- vVol-enabled datastore **TSA_PURE_FLASH_VVOL** backed by Pure x50 Storage
- Vendor-enabled storage on VMware vSphere Metro Storage Cluster
- Stretched vSAN 6.7 AFA hyperconverged-enabled datastore

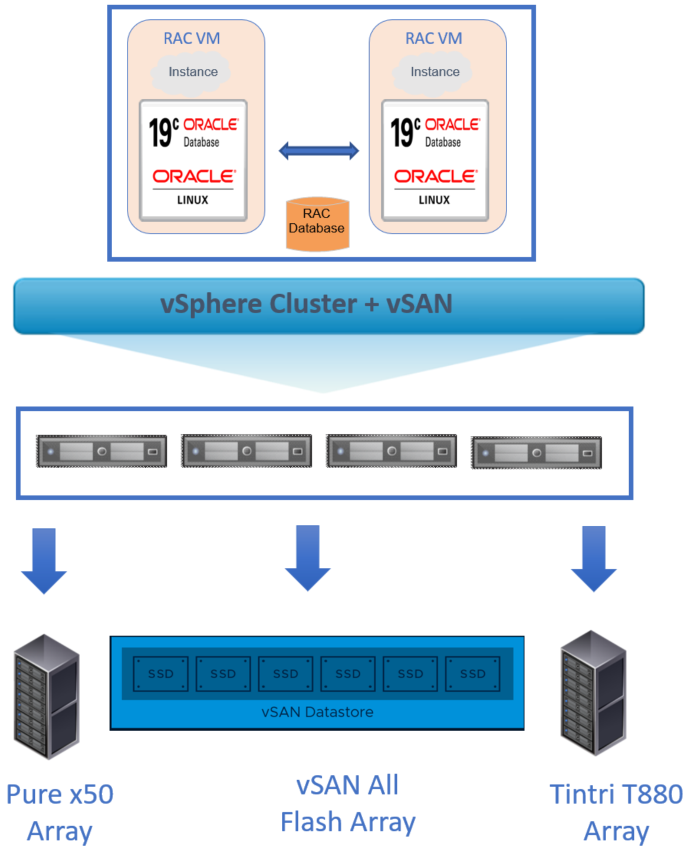


FIGURE 44. FOUR-NODE VSPHERE CLUSTER WITH CONNECTIVITY TO VSAN AFF, PURE X50 AND TINTRI T880 ARRAY

The deployments can be summarized as follows:

Location	RAC Deployment	DataStore Type	DataStore Name	Storage Vendor
On-premises	Traditional	VMFS FC	TSA_PURE_FLASH_4TB_01	Pure x50 AFA storage
On-premises	Traditional	VMFS NFS	TSA_TNTR_Oracle	Tintri T880 AFA storage
On-premises	Traditional	vsanDatastore	vsanSiteB	vSAN 6.7 AFA hyperconverged
On-premises	Traditional	VMware vVol	TSA_PURE_FLASH_VVOL	Pure x50 AFA storage
On-premises	Extended	Vendor Specific	Vendor Specific	vSphere Metro Storage Cluster
On-premises	Extended	vsanDatastore	vsanDatastore	vSAN 6.7 AFA hyperconverged

TABLE 19. ORACLE RAC DEPLOYMENTS

Solution Test Overview

This solution primarily validates the functional design of various Oracle RAC deployments, traditional and external, on various VMware platforms:

- VMFS datastore
- NFS datastore
- vSAN datastore
- vVol datastore
- VMware vSphere Metro Storage Cluster
- VMware vSAN Stretched Cluster

A two-node Oracle Database 19c RAC cluster was created for every deployment according to Oracle and VMware best practices.

Various test scenarios were conducted, which included:

- Abrupt termination of Oracle RAC crsd, ocssd, and evmd processes, observing if the process or node is restarted by the cluster
- Resiliency testing of Oracle RAC private interconnect, scan listeners, and agent
- vMotion of online Oracle RAC cluster (further details can be found *here*; demo of this study can be found *here*)
- In the case of Oracle RAC on vSAN storage, disk failure, disk-group failure, and storage-host failure scenarios were conducted (details of the test and its results can be found *here*)

Performance testing was not included as part of this reference architecture. Any performance data is a result of the combination of hardware configuration, software configuration, test methodology, test tool, and workload profile used in the testing.

Performance testing can be conducted by using the SLOB tool against the Oracle RAC cluster while generating a load on the database. Oracle AWR and Linux SAR reports can be captured to compare the performance and validate the testing use cases.

Oracle RAC Storage Deployment Guidelines

The steps for deploying an Oracle RAC, traditional or extended, on VMware vSphere are essentially the same on all VMware vSphere platforms.

There are, however, subtle differences in the way Oracle RAC shared storage is provisioned across various VMware vSphere platforms.

Oracle RAC shared-storage provisioning and the changes for the on-premise VMware Platforms are summarized in the table below:

VMware Platform	Datastore	Version	RAC shared VMDK requirement for multi-writer attribute	Reference
VMware vSphere	VMFS	ESXi 5.x and later	EZT	*KB 1034165*
VMware vSphere	NFS	ESXi 5.x and later	EZT	*KB 1034165*
VMware vSphere	vVol	ESXi 6.5 and later	Vendor specific	*KB 1034165* and *KB 2113013*
VMware vSAN	vsanDatastore	Versions prior to vSAN 6.7 Patch P01	EZT	*KB 2121181*
VMware vSAN	vsanDatastore	Beginning with VMware vSAN 6.7 Patch P01 (ESXi 6.7 Patch Release ESXi670-201912001)	Thin-provisioned	*KB 2121181*

TABLE 20. ORACLE RAC STORAGE DEPLOYMENTS

The following sections focus on deploying shared storage for an Oracle RAC Cluster using the multi-writer attribute on the following platforms:

- Oracle RAC storage on VMFS datastore
- Oracle RAC storage on NFS datastore
- Oracle RAC storage on vSAN datastore
- Oracle RAC storage on vVol datastore
- Extended Oracle RAC storage on VMware vSphere Metro Storage Cluster
- Extended Oracle RAC storage on VMware vSAN Stretched Cluster

Oracle RAC Storage on VMFS datastore

This section shows the steps to add a shared VMDK with the multi-writer attribute as an Oracle ASM disk to an Oracle RAC 19c cluster using an FC-enabled VMFS datastore TSA_PURE_FLASH_4TB_01 backed by Pure x50 Storage.

The remaining steps are described in section **Oracle RAC Deployment High Level Steps**.

The Pure Storage VMFS datastore TSA_PURE_FLASH_4TB_01 datastore is shown below:

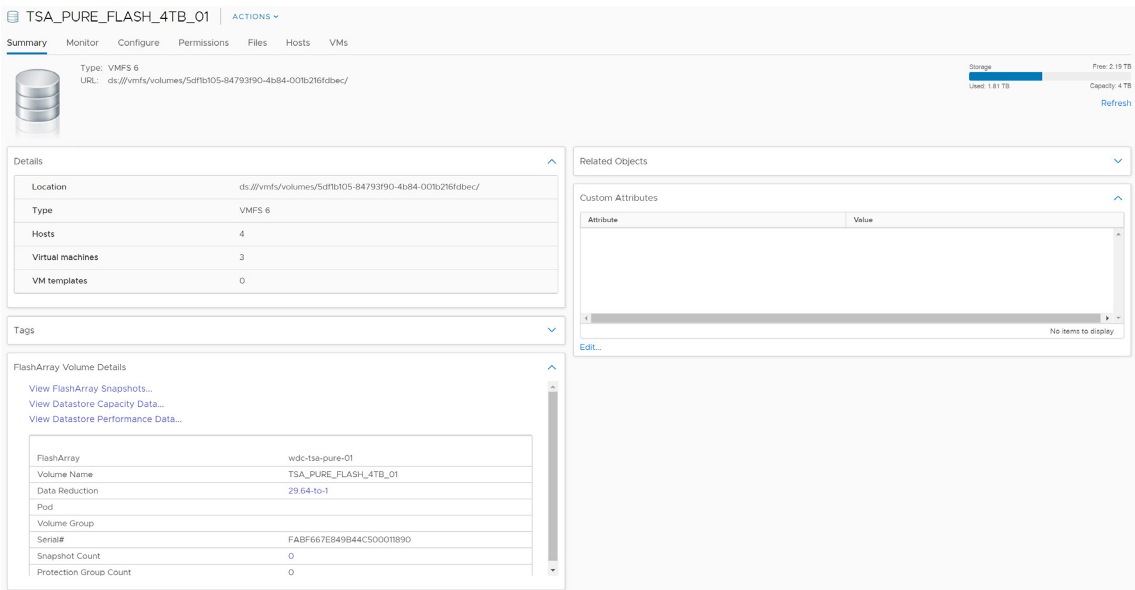


FIGURE 45. PURE STORAGE VMFS DATASTORE

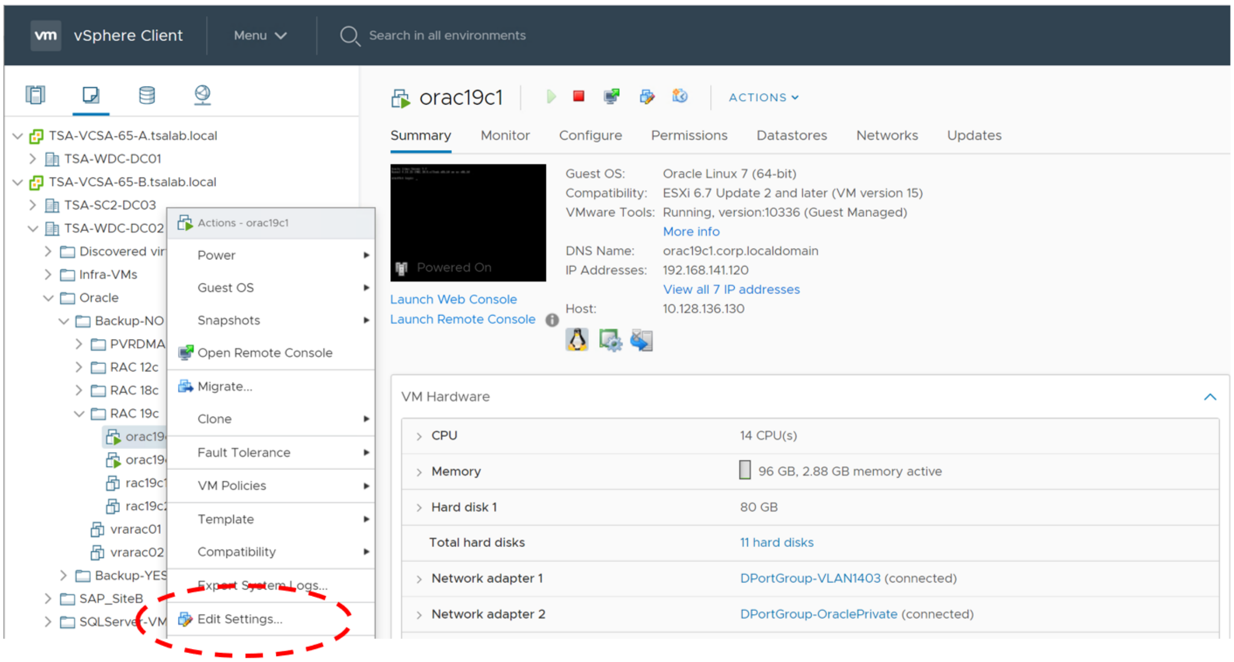
The shared storage provisioning guideline for FC VMFS datastore is shown as below:

VMware Platform	Datastore	Version	RAC shared VMDK requirement for multi-writer attribute	Reference
VMware vSphere	VMFS	ESXi 5.x and later	EZT	*KB 1034165*

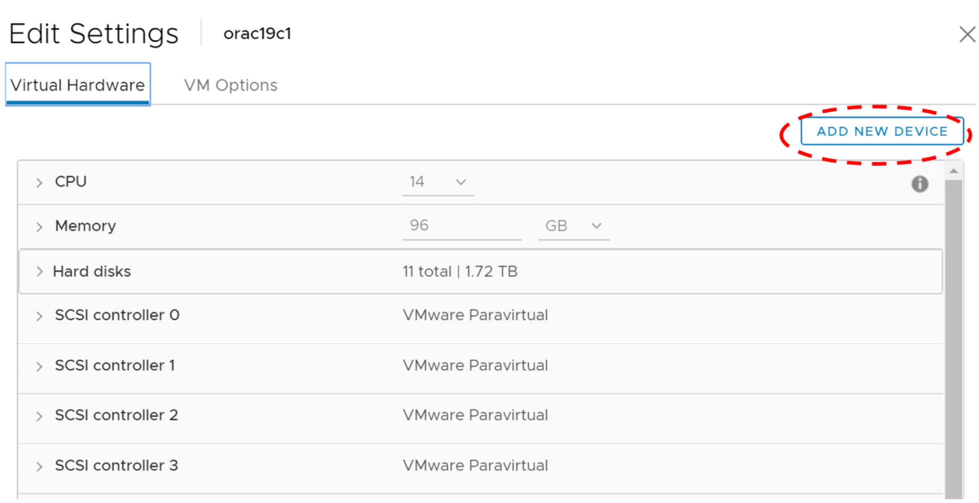
TABLE 21. SHARED STORAGE PROVISIONING GUIDELINE

The steps to add a 1TB VMDK as a shared Oracle ASM disk at SCSI position 1:0 to RAC VMs **orac19c1** and **orac19c2** provisioned from the VMFS datastore TSA_PURE_FLASH_4TB_01 are as shown below:

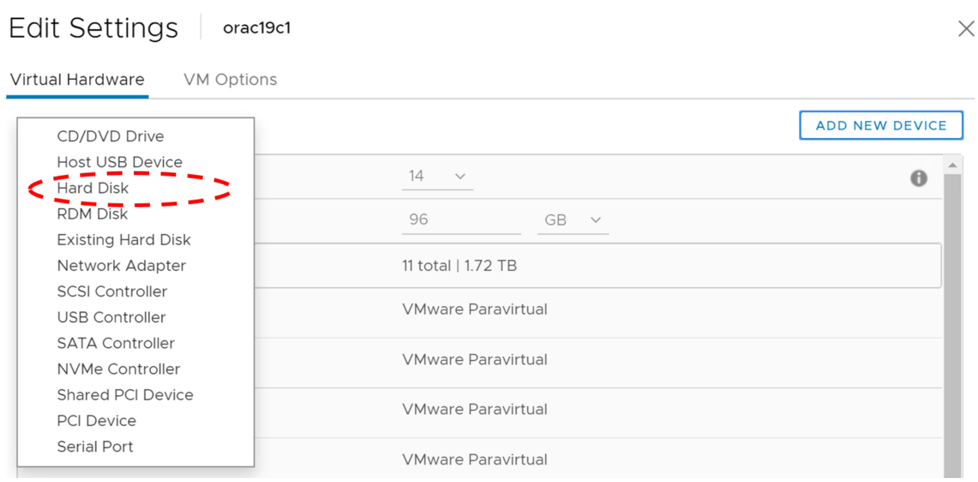
1. Right click Oracle RAC VM **orac19c1** and click **Edit Settings**.



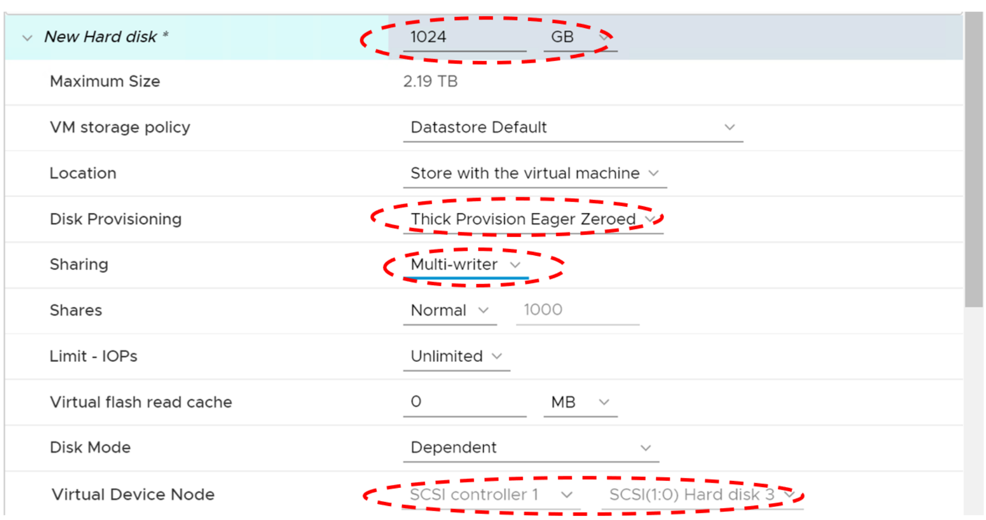
2. Click **Add New Device**.



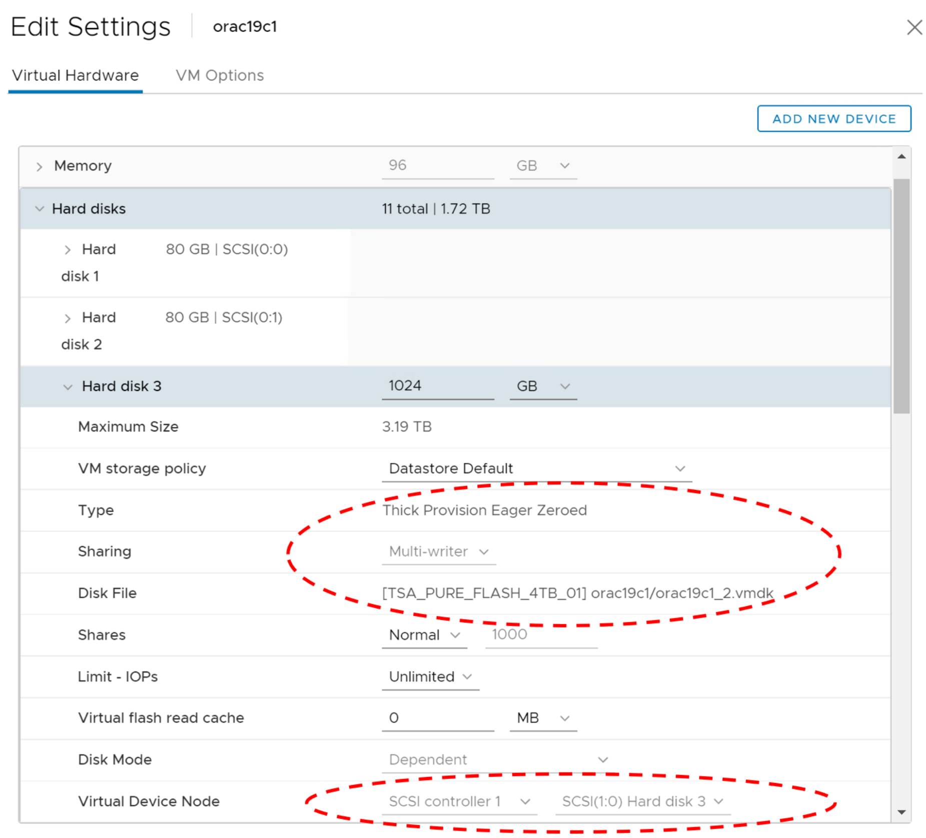
3. Click **Hard Disk**.



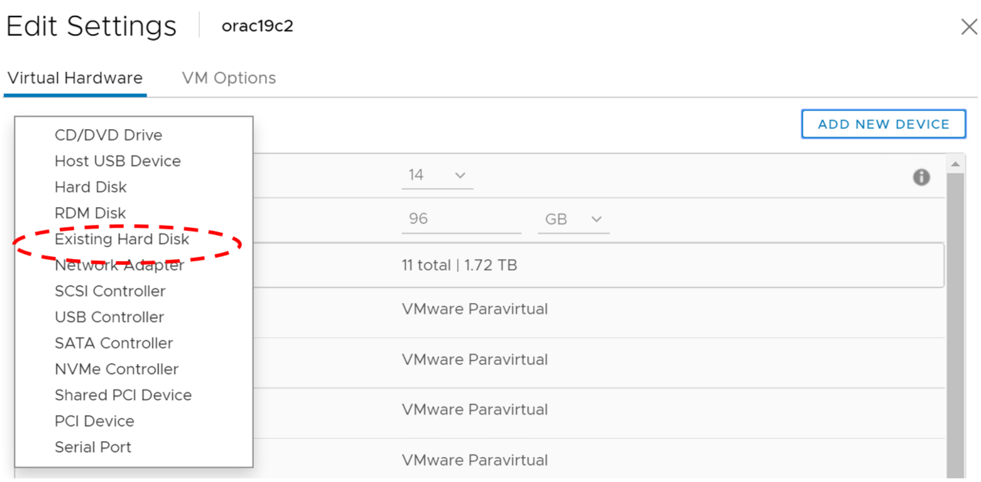
- Set the correct VMDK size (in this case, 1TB)
- Set VM storage policy to **Datastore Default**
- Set **Disk Provisioning** to **Thick Provision Eager Zeroed**
- Set **Sharing** to **Multi-writer**
- Set **Virtual Device Node** to SCSI position SCSI1:0
- Independent persistent mode is **not** required for enabling the multi-writer attribute
- Click **OK** to save



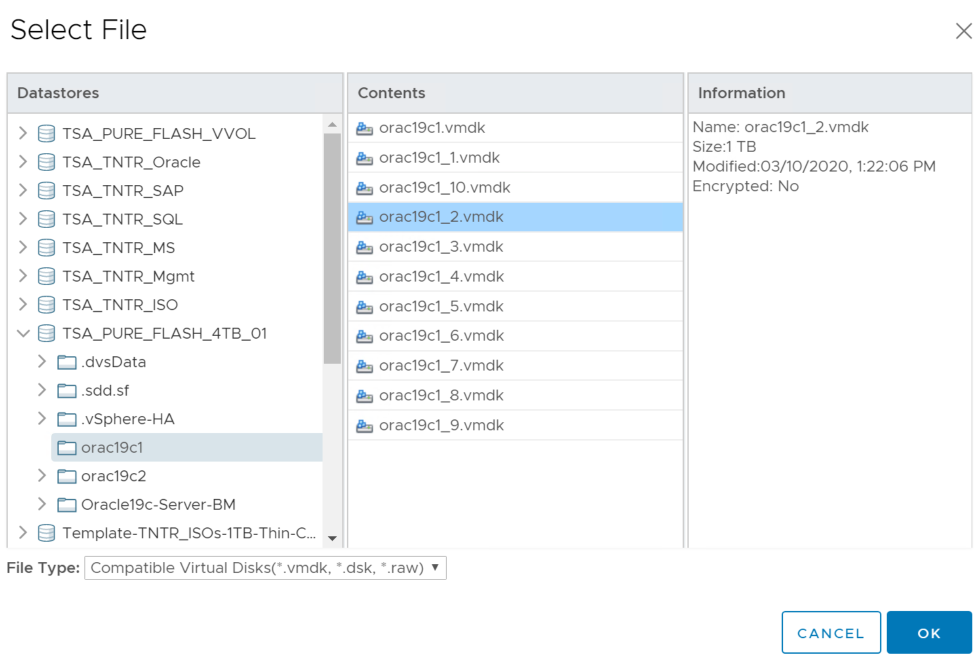
4. Ensure the following are properly indicated once the VMDK is created:
- Name of VMDK is **TSA_PURE_FLASH_4TB_01 orac19c1/orac19c1_2.vmdk**
- VMDK is provisioned on SCSI Controller SCSI1:0 position
- VMDK is provisioned as **Thick Provisioned Eager Zeroed**
- VMDK **Sharing** is set to **Multi-writer**
- **Disk Mode** remains at default setting **Dependent**



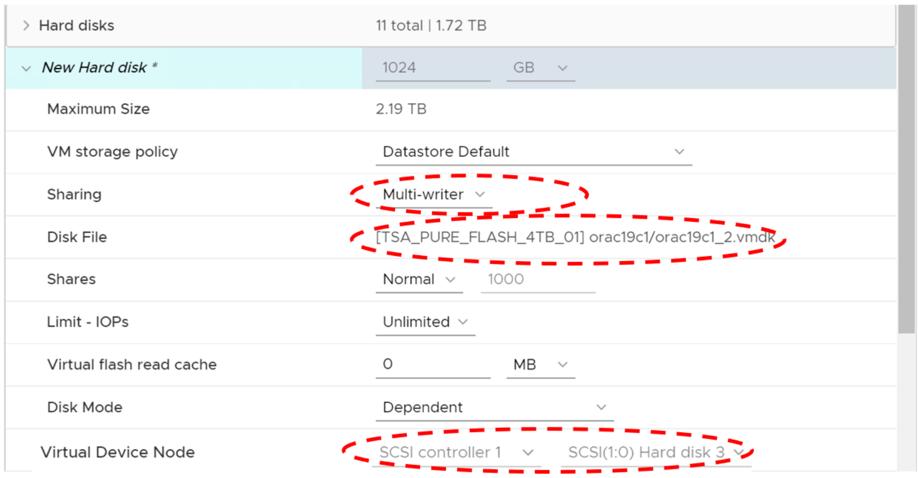
5. For VM **orac19c2**, add the same 1TB VMDK **TSA_PURE_FLASH_4TB_01 orac19c1/orac19c1_2.vmdk** created for Oracle ASM disk on VM **orac19c1** to PVSCSI controller on SCSI position 1:0.
6. Repeat steps **one** and **two** as completed for VM **orac19c1**.
7. For step **three**, instead of choosing **Hard Disk**, choose **Existing Hard Disk**.



8. Navigate to the VM **orac19c1** folder on **TSA_PURE_FLASH_4TB_01** datastore and select **orac19c1_2.vmdk** 1TB VMDK. Click **OK**.

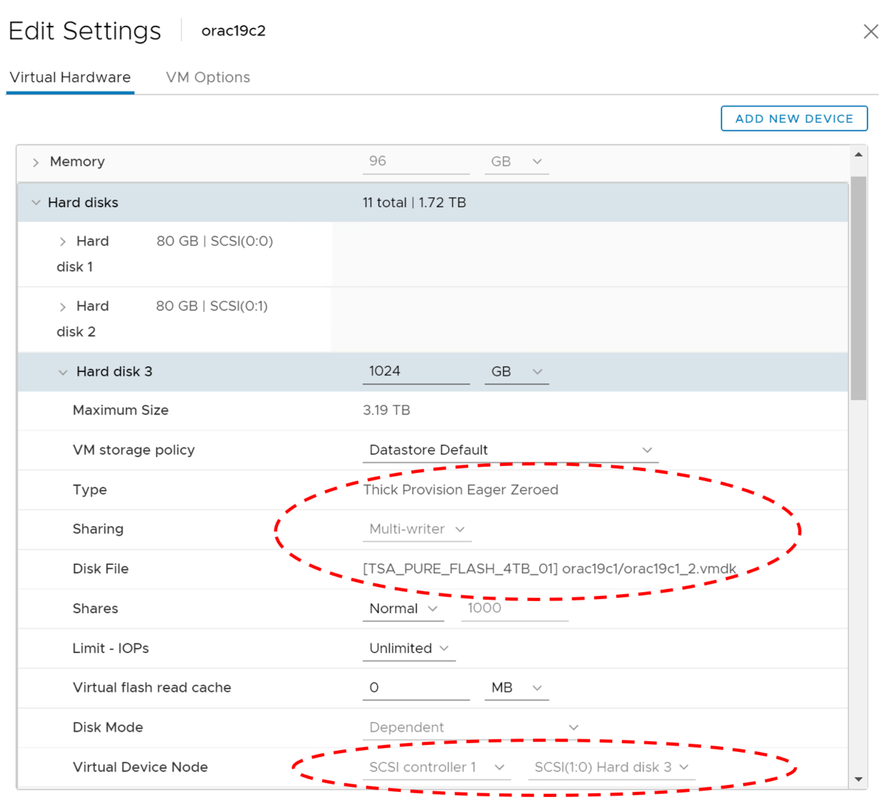


9. Provision the VMDK on the same SCSI controller SCSI 1:0 position as designated for Oracle RAC VM **orac19c1**. Follow steps five through seven and click **OK** to save.

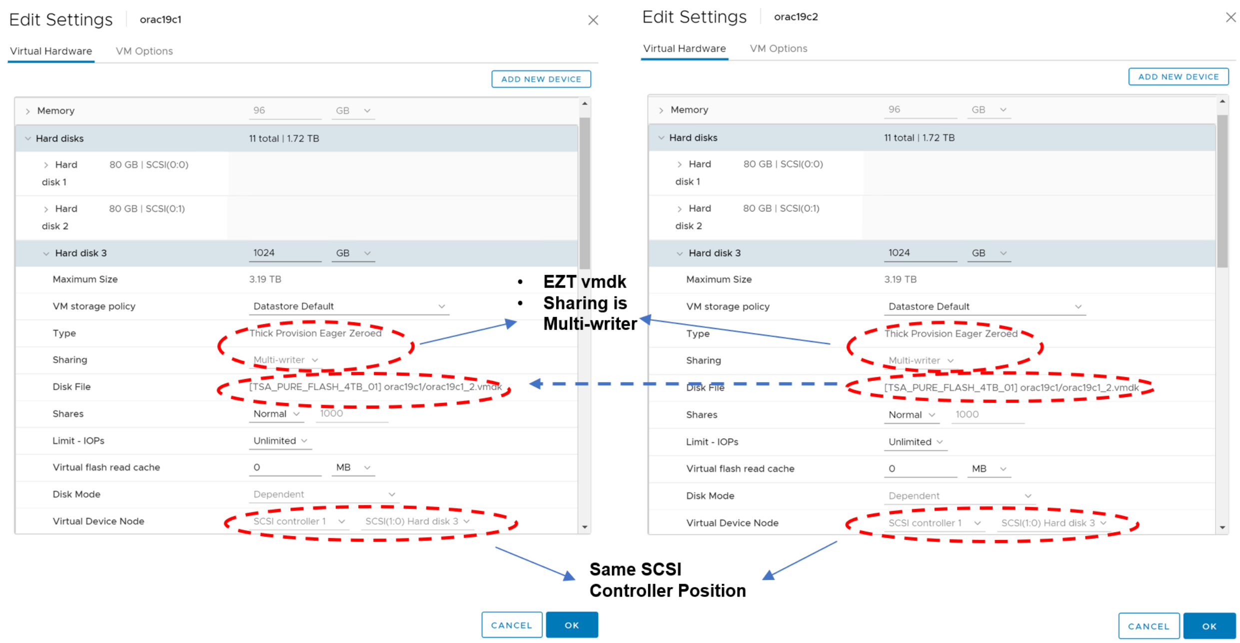


10. Ensure the following are indicated:

- Name of VMDK is **[TSA_PURE_FLASH_4TB_01] orac19c1/orac19c1_2.vmdk**, as is VM **orac19c1**
- VMDK is provisioned on SCSI Controller SCSI 1:0 position
- VMDK is **Thick Provisioned Eager Zeroed**
- VMDK **Sharing** is set to **Multi-writer**
- **Disk Mode** is set at the default **Dependent**

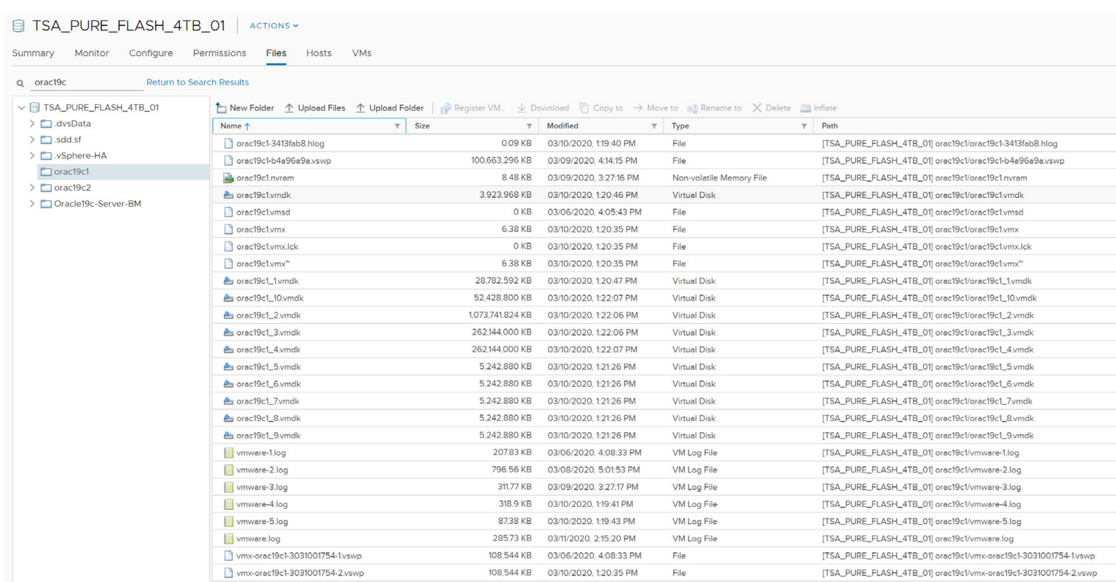


11. Newly provisioned 1TB shared VMDK on Oracle RAC VM **orac19c1** and **orac19c2** is shown below. Note that Oracle RAC VM **orac19c2** 1TB VMDK is referring to Oracle RAC VM **orac19c1** VMDK.

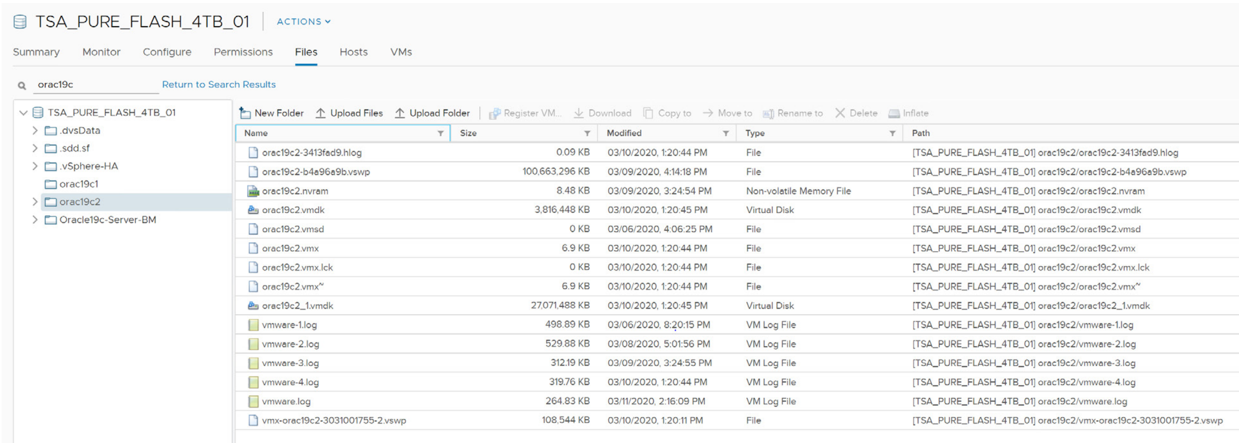


12. Repeat the steps above to provision the remaining Oracle RAC shared disks.

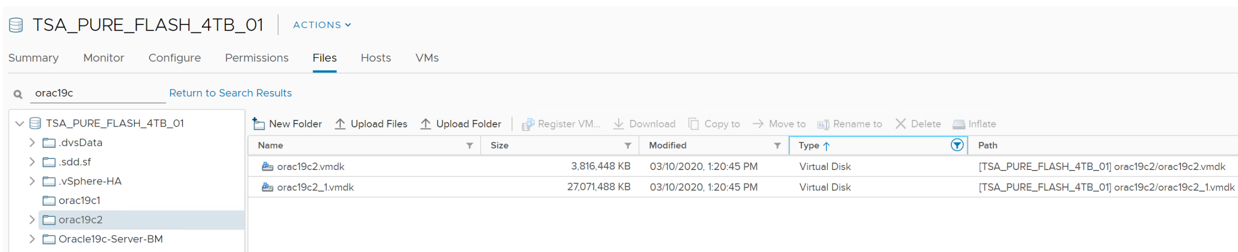
The Oracle RAC VM **ora19c1** VMDKs are as shown below in the FC VMFS datastore:



The Oracle RAC VM **ora19c2** VMDKs are as shown below in the FC VMFS datastore:



As shown below, ensure that the only VMDKs created on Oracle RAC VM **ora19c2** folder are the OS and Oracle binaries:



Another way to see the same data is to check the contents of the .vmx file for Oracle RAC VM **orac19c2** as shown below. Note that only VMDKs created for Oracle RAC VM **ora19c2** are the OS and Oracle binaries. Remaining disks are a reference to the shared disks created on Oracle RAC VM **orac19c1**.

```
[root@wdc-esx30:/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c2] cat orac19c2.vmx | grep -i fileName | grep -i vmdk
scsi0:0.fileName = "orac19c2.vmdk"
scsi0:1.fileName = "orac19c2_1.vmdk"
scsi1:0.fileName = "/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1/orac19c1_2.vmdk"
scsi2:0.fileName = "/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1/orac19c1_3.vmdk"
scsi2:1.fileName = "/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1/orac19c1_4.vmdk"
scsi3:0.fileName = "/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1/orac19c1_5.vmdk"
scsi3:1.fileName = "/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1/orac19c1_6.vmdk"
scsi3:2.fileName = "/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1/orac19c1_7.vmdk"
scsi3:3.fileName = "/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1/orac19c1_8.vmdk"
scsi3:4.fileName = "/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1/orac19c1_9.vmdk"
scsi3:5.fileName = "/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1/orac19c1_10.vmdk"
[root@wdc-esx30:/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c2]
```

Oracle RAC Storage on NFS datastore

This section details the steps required to add a shared VMDK with multi-writer attribute as an Oracle ASM disk to an Oracle RAC 19c cluster, using an NFS datastore **TSA_TNTR_Oracle** backed by Tintri storage.

In the event of NFS datastores that do not support vSphere APIs for array integration (VAAI), and therefore allow creation of EZT VMDKs via the web client, refer to *KB 2147691* for steps to create EZT VMDKs.

The remaining steps required to deploy an Oracle 19c RAC cluster are described in the section **Oracle RAC Deployment High Level Steps** below.

The Tintri NFS datastore **TSA_TNTR_Oracle** is shown as follows:

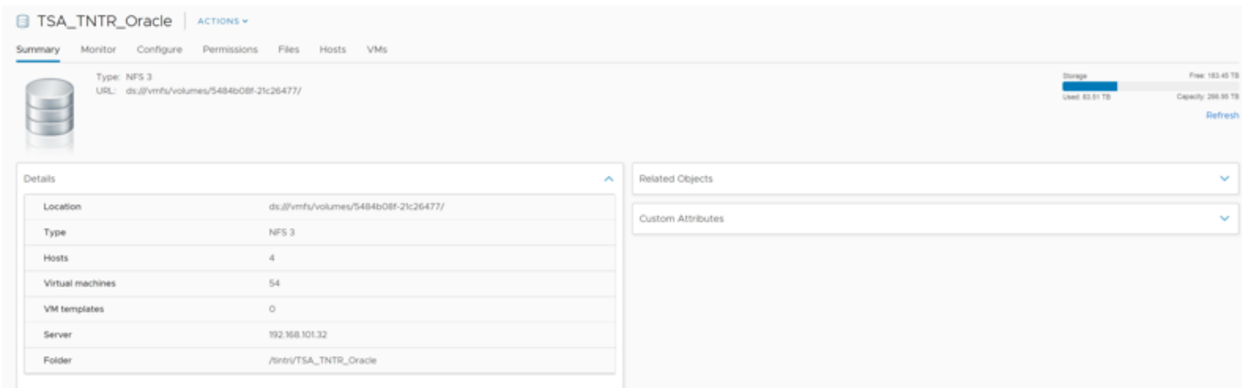


FIGURE 46. TINTRI NFS DATASTORE TSA_TNTR_ORACLE

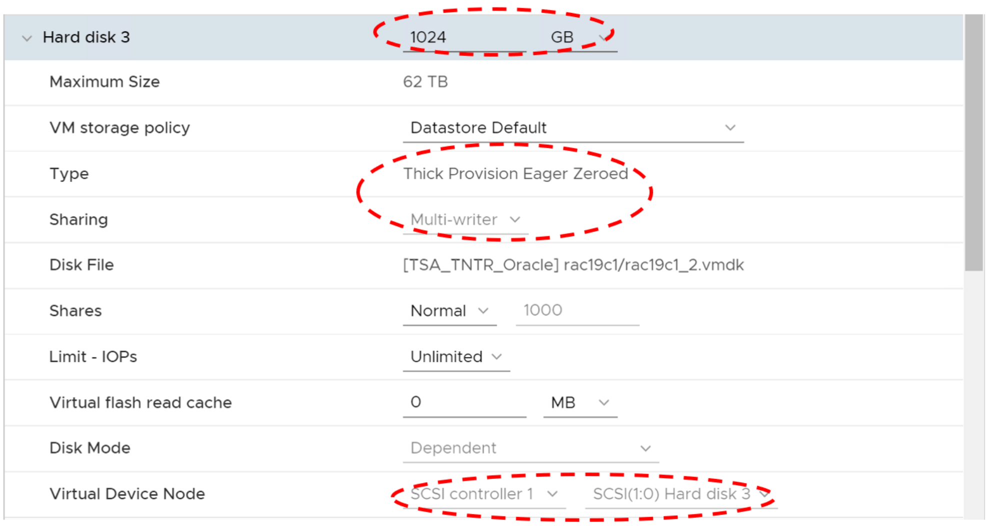
The shared storage provisioning guideline for NFS datastore is shown as below

VMware Platform	Datastore	Version	Oracle RAC shared VMDK requirement for multi-writer attribute	Reference
VMware vSphere	NFS	ESXi 5.x and later	EZT	*KB 1034165*

TABLE 22. NFS DATASTORE SHARED PROVISIONING GUIDELINE

The steps to add a 1TB VMDK as a shared Oracle ASM disk at SCSI position 1:0 to Oracle RAC VMs **rac19c1**, and **rac19c2** provisioned from the NFS datastore **TSA_TNTR_Oracle**, are the same as those of the VMFS datastore and steps outlined below:

1. Right click Oracle RAC VM **rac19c1** and click **Edit Settings**
2. Click **Add New Device**
3. Click **Hard Disk**
   – Set the correct VMDK size, in this case 1TB
   – Set the VM storage policy to **Datastore Default**
   – Set the **Disk Provisioning** to **Thick Provision Eager Zeroed**
   – Set the **Sharing** to **Multi-writer**
   – Set the **Virtual Device Node** to SCSI position SCSI1:0
   – Independent persistent mode is **not** required for enabling the multi-writer attribute
   – Click **OK** to save



4. Ensure the following settings are indicated once the VMDK is created:
   – Name of VMDK is **[TSA_TNTR_Oracle] rac19c1/rac19c1_2.vmd**k
   – VMDK is provisioned on SCSI Controller SCSI1:0 position
   – VMDK is **Thick Provisioned Eager Zeroed**
   – VMDK Sharing is set to **Multi-writer**
   – **Disk Mode** is set to the default **Dependent**

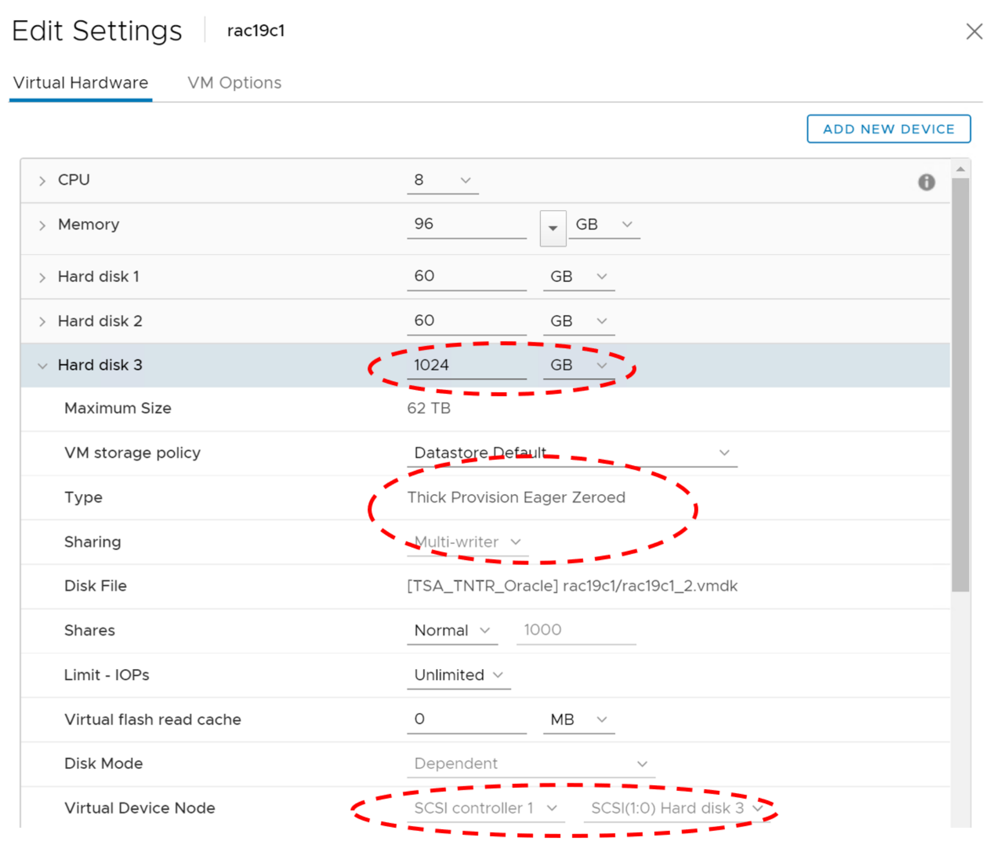
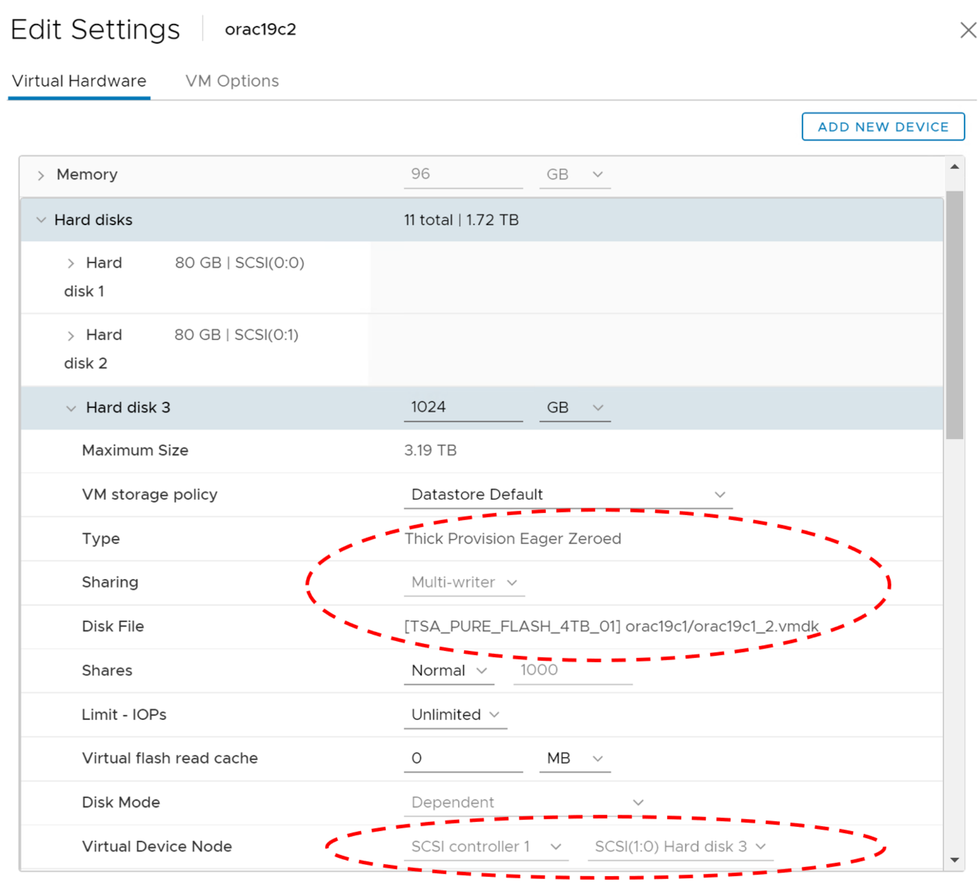


Figure 47. PVSCSI Controller Backed Oracle ASM DATA EZT VMDK

5. For VM **rac19c2**, add the same 1TB VMDK **[TSA_TNTR_Oracle] rac19c1/rac19c1_2.vmdk** file created for Oracle ASM disk on VM **rac19c1** to PVSCSI controller on SCSI position 1:0.
6. Repeat steps **one** and **two** as previously completed for VM **rac19c1**.
7. For step **three**, instead of choosing **Hard Disk**, choose **Existing Hard Disk**. Navigate to the VM **rac19c1** folder on **TSA_TNTR_Oracle** datastore and select **rac19c1_2.vmdk** 1TB VMDK. Click **OK**.



8. Ensure the following settings are indicated:
   – Name of VMDK is **[TSA_TNTR_Oracle] rac19c1/rac19c1_2.vmdk**, as is the case for VM **orac19c1**
   – VMDK is provisioned on SCSI Controller SCSI 1:0 position
   – VMDK is **Thick Provisioned Eager Zeroed**
   – VMDK **Sharing** is set to **Multi-writer**
   – **Disk Mode** is set to the default **Dependent**

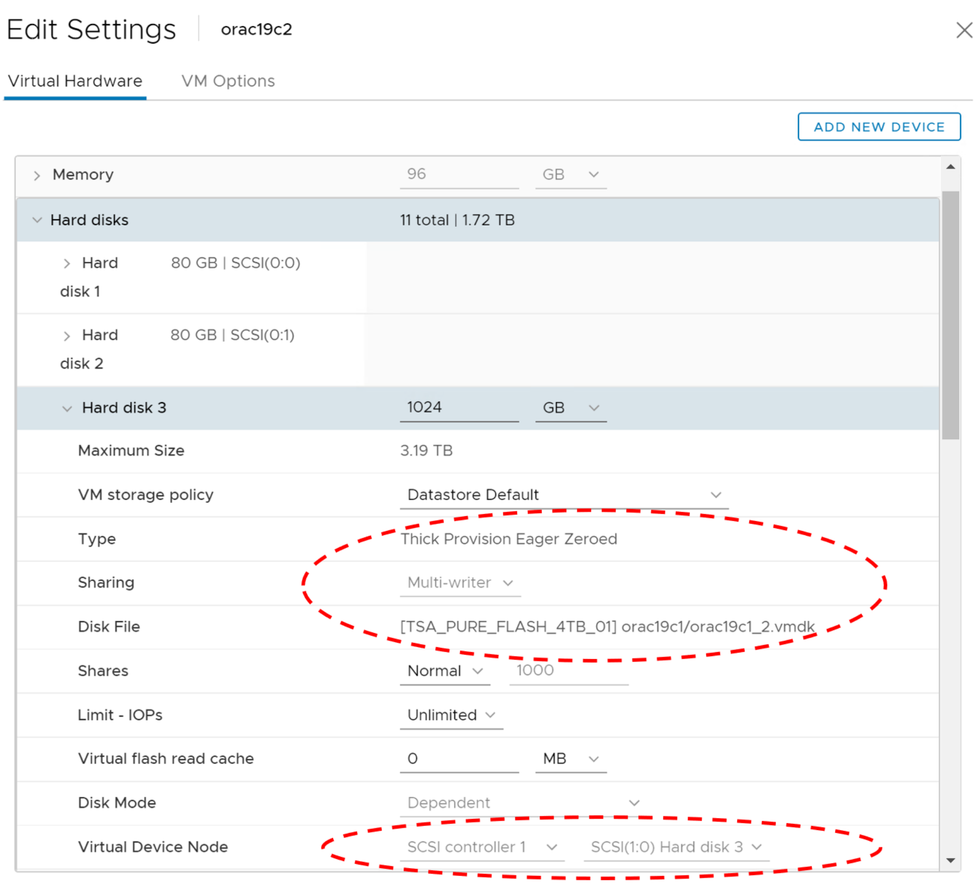
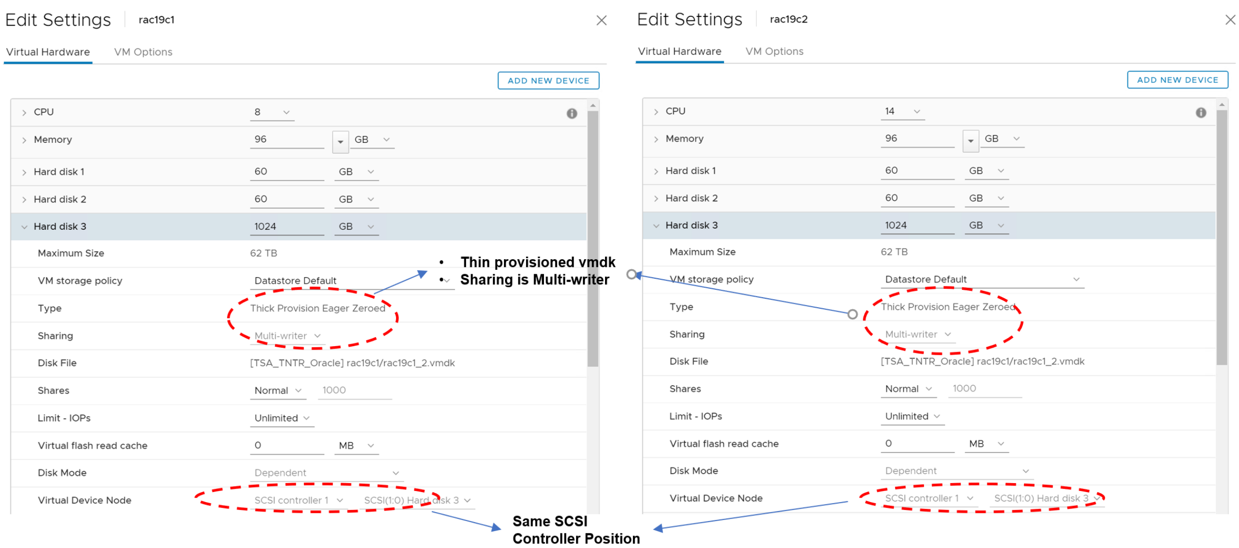


FIGURE 48. PVSCSI CONTROLLER BACKED ORACLE ASM DATA EZT VMDK

9. The freshly provisioned 1TB shared VMDK on Oracle RAC VM **orac19c1** and **orac19c2** is shown below. Note that Oracle RAC VM **orac19c2** 1TB VMDK refers to Oracle RAC VM **orac19c1** VMDK.



10. Repeat the steps above to provision the remaining Oracle RAC shared disks.

The Oracle RAC VM **ra19c1** VMDKs are shown below in the NFS datastore **TSA_TNTR_Oracle**:

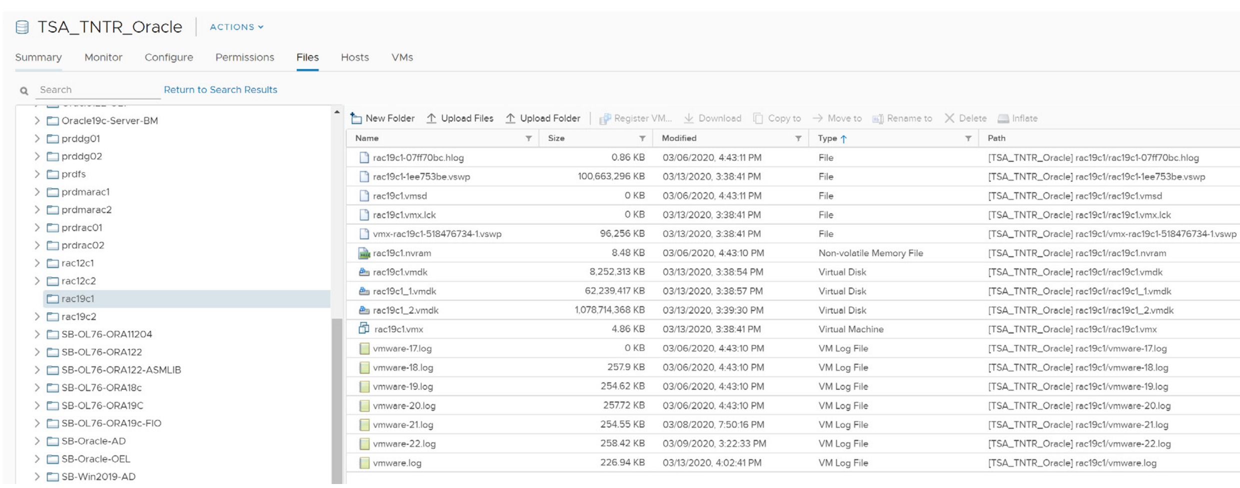


FIGURE 49. ORACLE RAC VM RA19CI IN NFS DATASTORE

The Oracle RAC VM **ra19c2** VMDKs are shown below in the NFS datastore **TSA_TNTR_Oracle**:

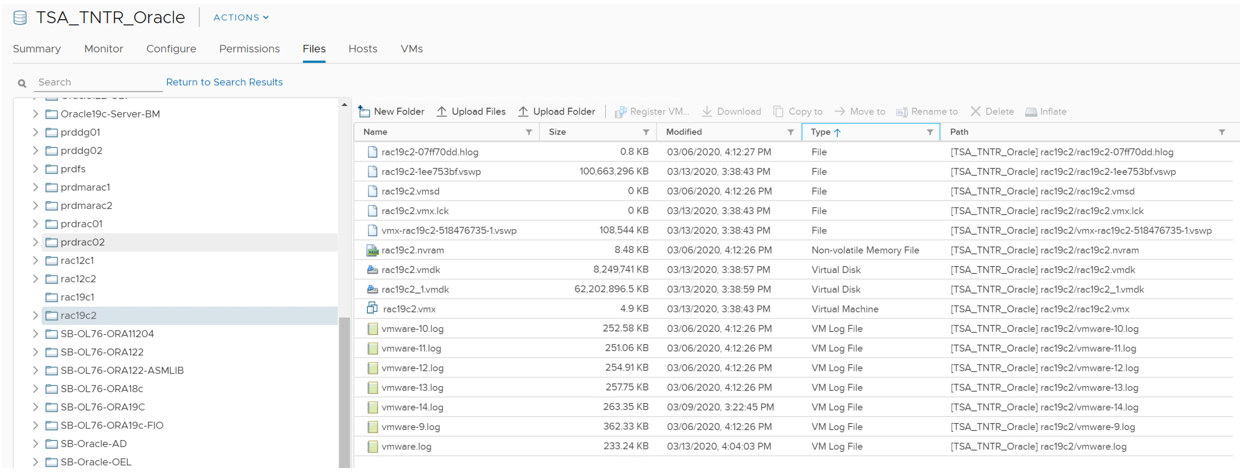


FIGURE 50. ORACLE RAC VM RA19C2 IN NFS DATASTORE

Note that the only VMDKs created on Oracle RAC VM **ora19c2** folder are the OS and Oracle binaries, as indicated by the figure below:

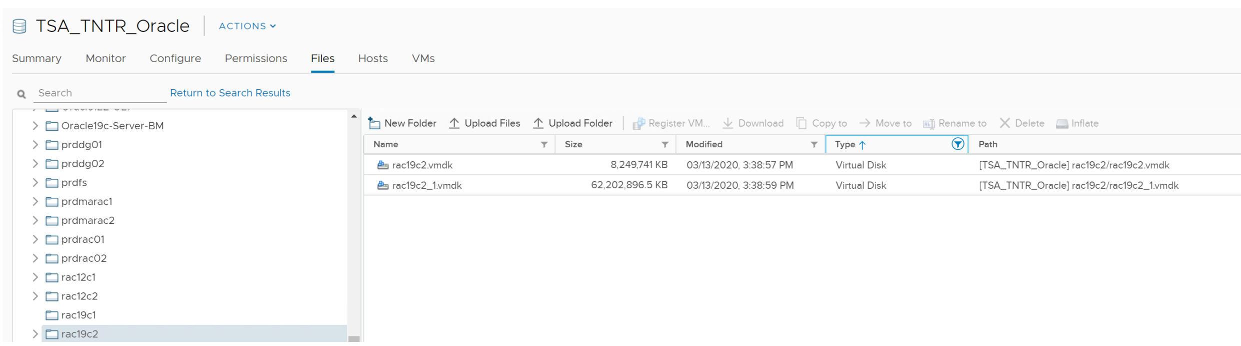


FIGURE 51. OS AND ORACLE BINARIES

Another way to see the same data is to check the contents of the .vmx file for Oracle RAC VM **rac19c2**. Note that the only VMDKs created for Oracle RAC VM **ra19c2** are the OS and Oracle binaries, while the remaining disks refer to the shared disks created on Oracle RAC VM **rac19c1**.

```
[root@wdc-esx30:/vmfs/volumes/5484b08f-21c26477/rac19c2] cat rac19c2.vmx | grep -i scsi | grep -i filename
scsi0:0.fileName = "rac19c2.vmdk"
scsi0:1.fileName = "rac19c2_1.vmdk"
scsi1:0.fileName = "/vmfs/volumes/5484b08f-21c26477/rac19c1/rac19c1_2.vmdk"
[root@wdc-esx30:/vmfs/volumes/5484b08f-21c26477/rac19c2]
```

FIGURE 52. ORACLE RAC VM RAC19C2 .VMX FILE

Oracle RAC Storage on vSAN datastore

This section outlines the steps required to add a shared VMDK with the multi-writer attribute as an Oracle ASM disk to an Oracle RAC 19c cluster using a vSAN datastore **vsanSiteB**.

The remaining steps to deploy an Oracle 19c RAC cluster are described in the **Oracle RAC Deployment High Level Steps** section of this document.

The vSAN datastore **vsanSiteB** is shown below:

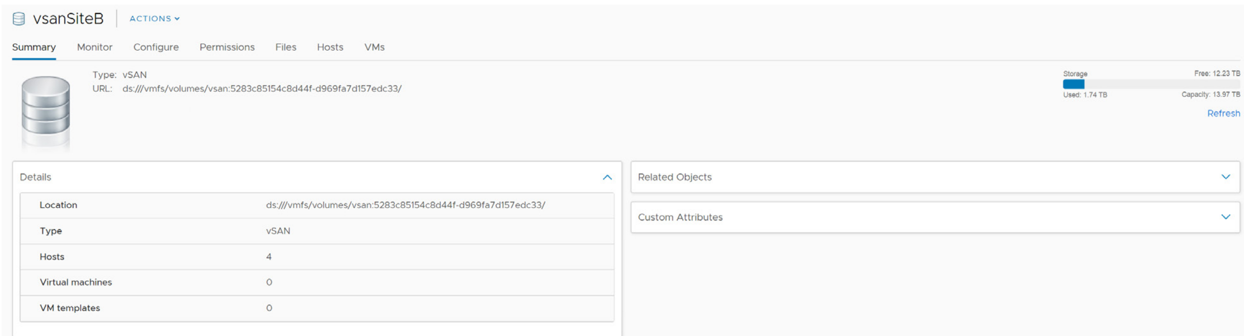


FIGURE 53. VSAN DATASTORE VSANSITEB

The shared storage provisioning guideline for vSAN datastore is as follows:

VMware Platform	Datastore	Version	Oracle RAC shared VMDK requirement for multi-writer attribute	Reference
VMware vSAN	vsanDatastore	Prior vSAN 6.7 Patch P01	EZT	*KB 2121181*
VMware vSAN	vsanDatastore	Beginning with VMware vSAN 6.7 Patch P01 (ESXi 6.7 Patch Release ESXi670-201912001)	Thin-Provisioned	*KB 2121181*

TABLE 23. VSAN DATASTORE SHARED PROVISIONING GUIDELINE

As mentioned, for vSAN versions prior to vSAN 6.7 Patch P01, the virtual disk must be EZT to enable multi-writer mode. Further guidance regarding the addition of a shared EZT VMDK to an Oracle RAC cluster using the multi-writer attribute can be found at *Oracle Real Application Clusters on VMware vSAN*.

Beginning with VMware vSAN 6.7 Patch P01 (ESXi 6.7 Patch Release ESXi670-201912001), Oracle RAC on vSAN does not require the shared VMDKs to be EZT (OSR=100) for multi-writer mode to be enabled.

The steps required to add a 1TB VMDK as a shared Oracle ASM disk at SCSI position 1:0, to Oracle RAC VMs **vsanrac1** and **vsanrac2** provisioned from the vSAN datastore **vsanSiteB**, are the same as those for Oracle RAC on previous vSAN versions. The only difference is the shared disk can be thin-provisioned with the multi-writer attribute, beginning with vSAN 6.7 Patch P01.

Below is an example of a storage policy (**Oracle RAC vSAN - OSR 0**) created for the purposes of provisioning shared VMDKs thin-provisioned for Oracle RAC on VMware vSAN, beginning with VMware vSAN 6.7 P01 (ESXi 6.7 Patch Release ESXi670-201912001).

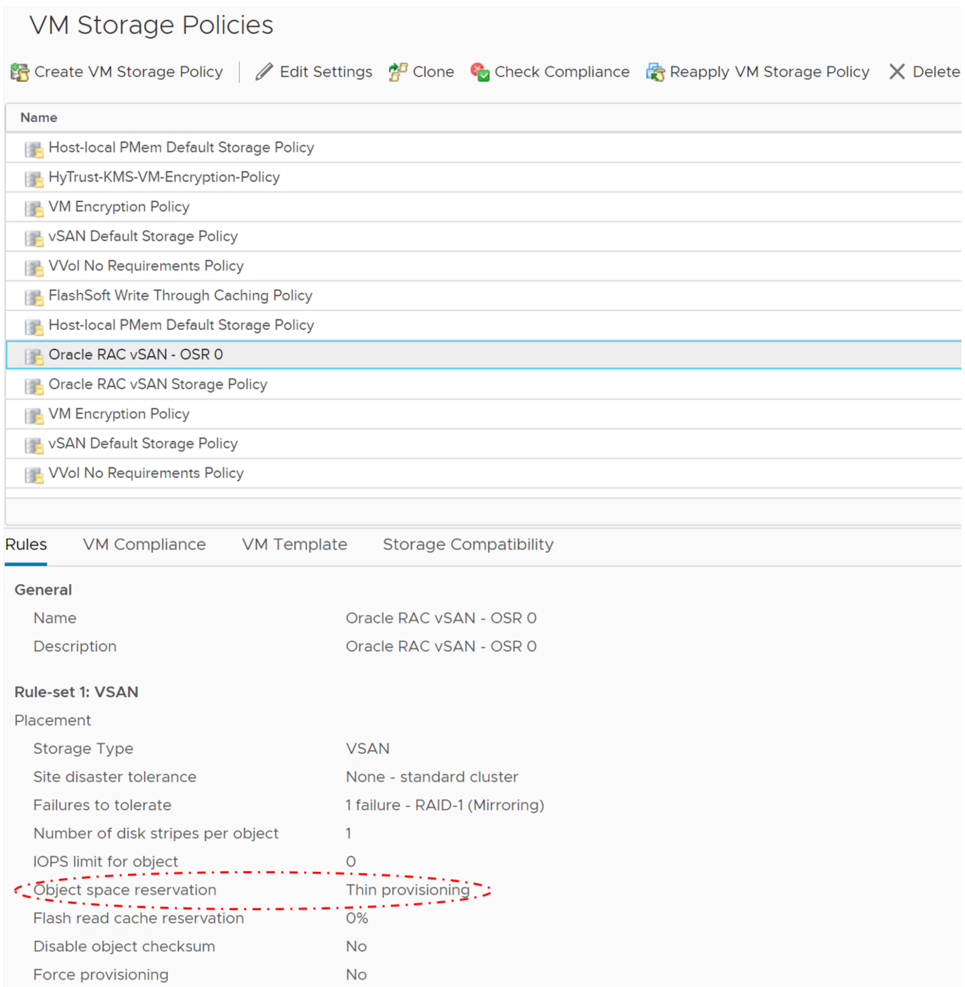
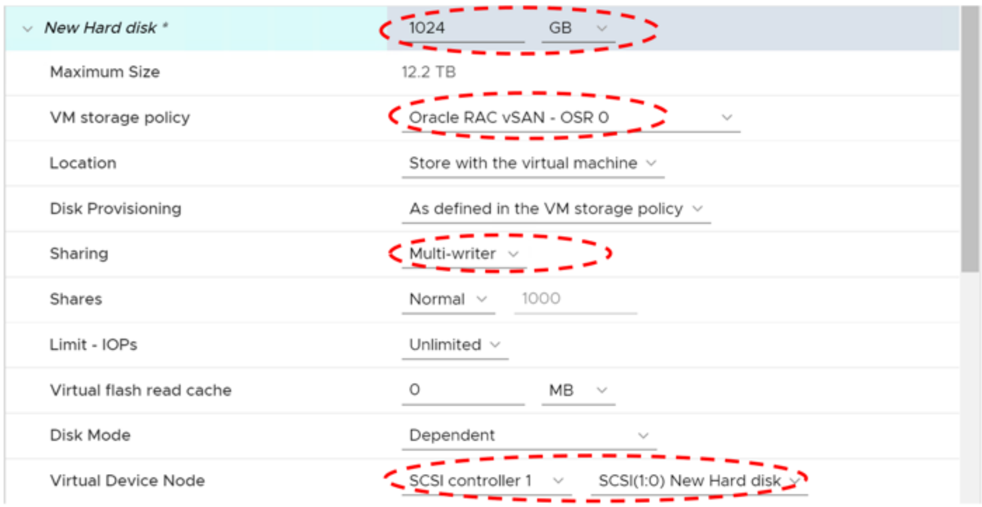


FIGURE 54. ORACLE RAC STORAGE POLICY BEGINNING WITH VSAN 6.7 P01

1. Right click Oracle RAC VM **vsanrac1** and click **Edit Settings**
2. Click **Add New Device**
3. Click **Hard Disk**
   – Set the correct VMDK size (in this case 1TB)
   – Set the VM storage policy to **Oracle RAC vSAN - OSR 0**
   – Set the disk provisioning to **As defined in the VM Storage policy**
   – Set the **Sharing** to **Multi-writer**

– Set the **Virtual Device Node** to SCSI position SCSI1:0
– Independent persistent mode is **not** required for enabling the multi-writer attribute
– Click **OK** to save



4. Ensure the following settings are indicated once the VMDK is created:
– Name of VMDK is **vsanSiteB cc176c5e-bde7-9cc7-a219-246e965377b8/vsanrac1_2.vmdk**
– VMDK is provisioned on SCSI Controller SCSI1:0 position
– VMDK **Sharing** is set to **Multi-writer**
– **Disk Mode** is left at default of **Dependent**

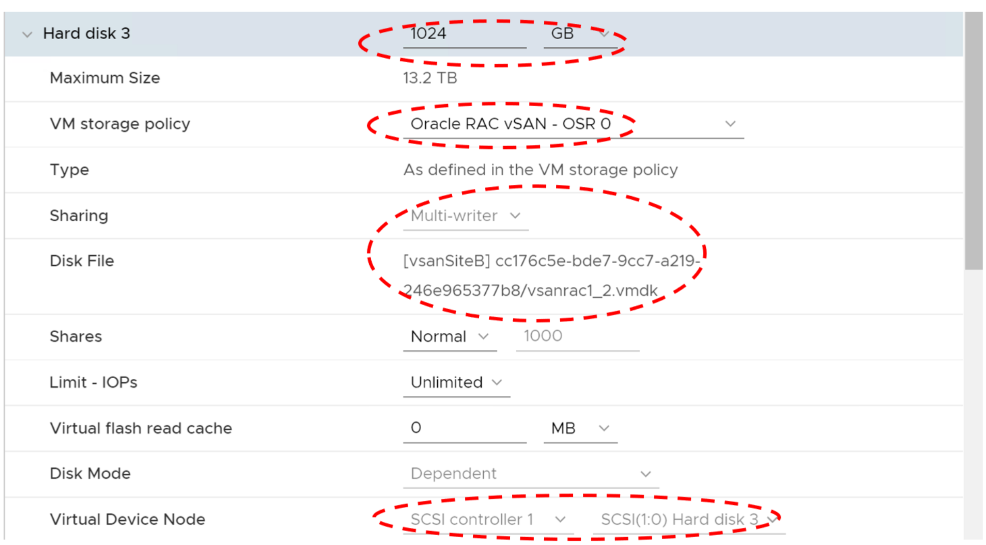
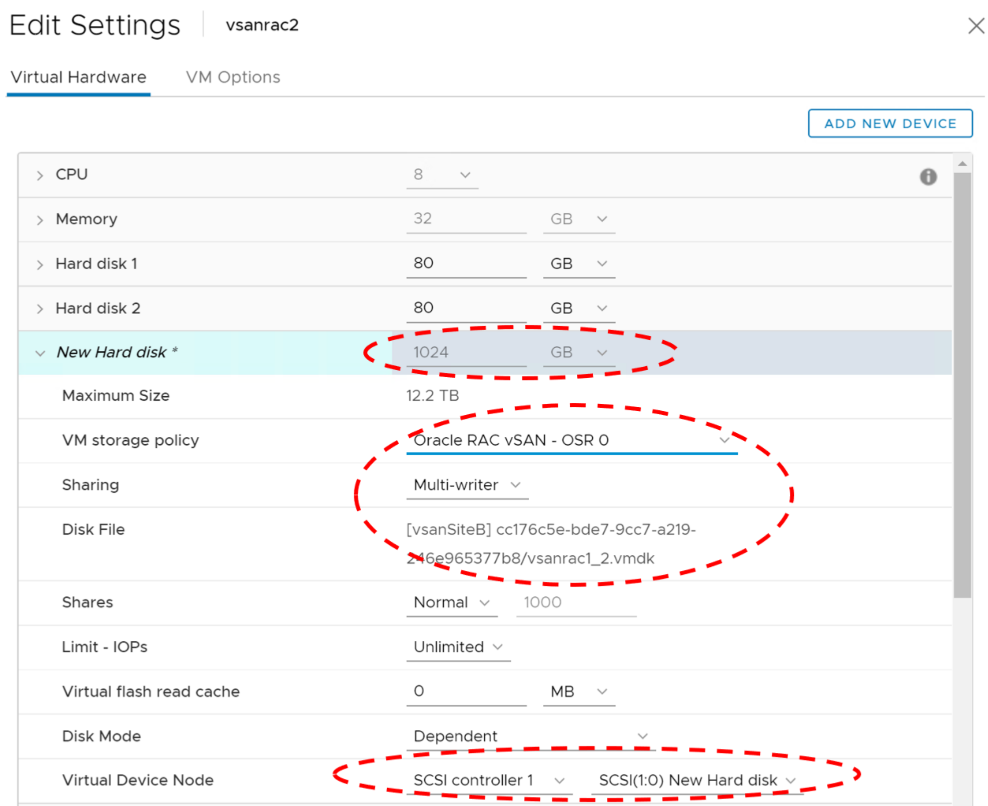


FIGURE 55. PVSCSI CONTROLLER BACKED ORACLE ASM DATA VMDK

5. For VM **vsanrac2**, add the same 1TB VMDK **[vsanSiteB] cc176c5e-bde7-9cc7-a219-246e965377b8/vsanrac1_2.vmdk** created for Oracle ASM disk on VM **vsanrac1** to PVSCSI Controller on SCSI position 1:0.
6. Repeat steps **one** and **two** as previously completed for VM **vsanrac1**.
7. For step **three**, instead of choosing **Hard Disk**, choose **Existing Hard Disk**. Navigate to the VM **vsanrac1** folder on **vsanSiteB** datastore and select **vsanrac1_2.vmdk**. Click **OK**.



8. Ensure the following settings are indicated:
   - Name of VMDK is **[vsanSiteB] cc176c5e-bde7-9cc7-a219-246e965377b8/vsanrac1_2.vmdk**
   - VMDK is provisioned on SCSI Controller SCSI1:0 position
   - VMDK **Sharing** is set to **Multi-writer**
   - **Disk Mode** is set to the default **Dependent**

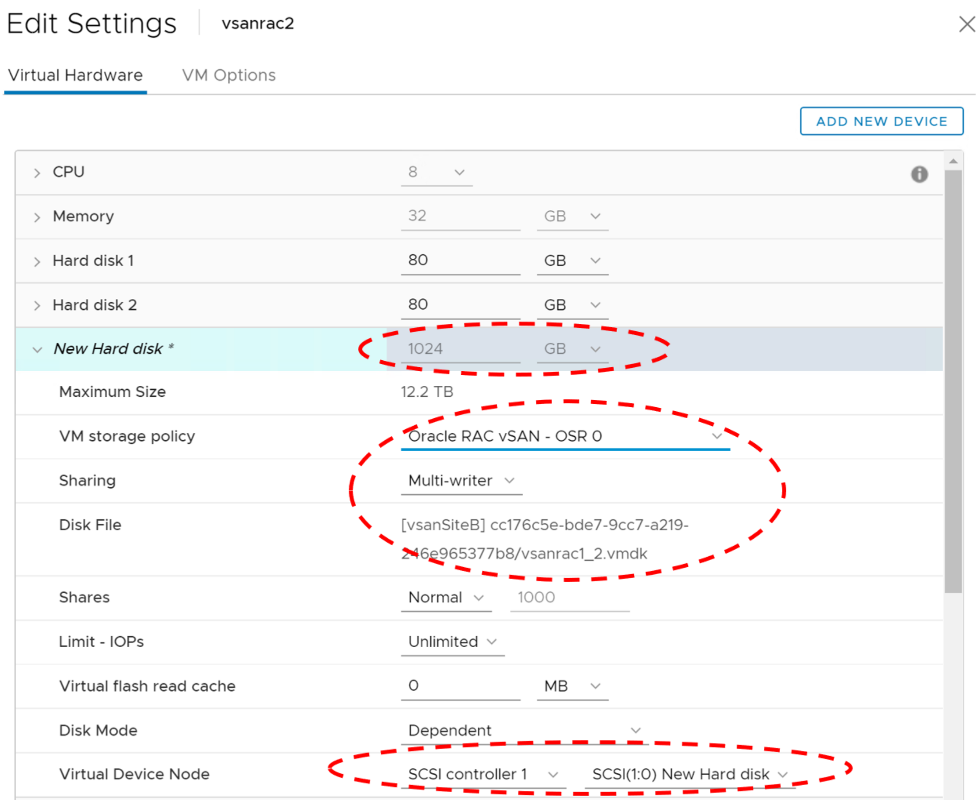
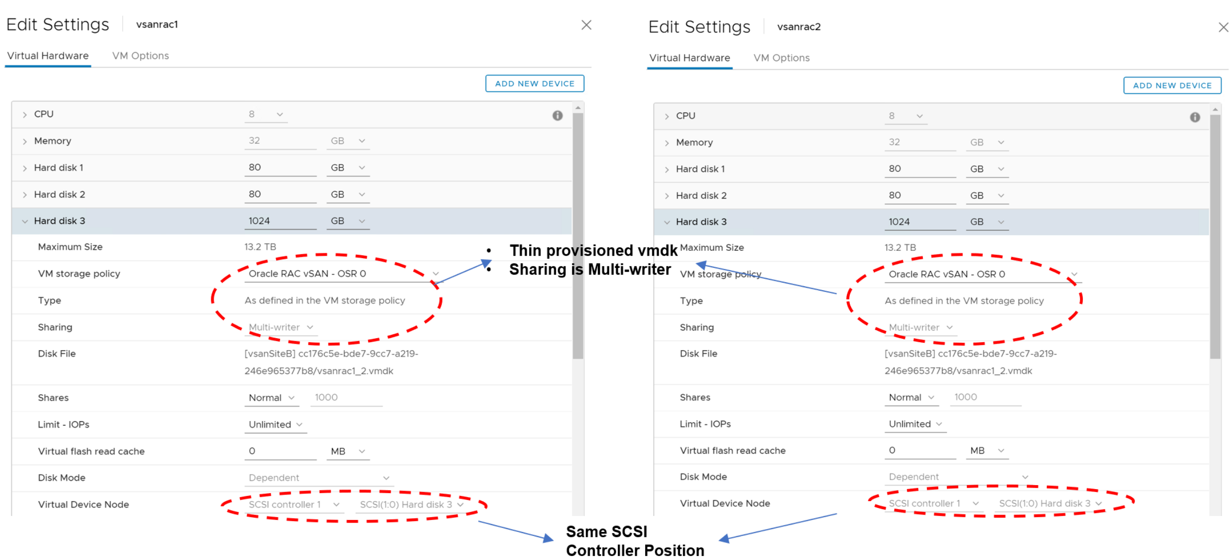


FIGURE 56. FIGURE PVSCSI CONTROLLER BACKED ORACLE ASM DATA VMDK

9. Freshly provisioned 1TB shared VMDK on Oracle RAC VM **vsanrac1** and **vsanrac2** is shown below. Note that Oracle RAC VM **vsanrac2** 1TB VMDK refers to Oracle RAC VM **vsanrac1** VMDK.



10. Repeat the steps above to provision the remaining Oracle RAC shared disks.

The Oracle RAC VM **vsanrac1** VMDKs are shown below in vSAN datastore **vsanSiteB**:

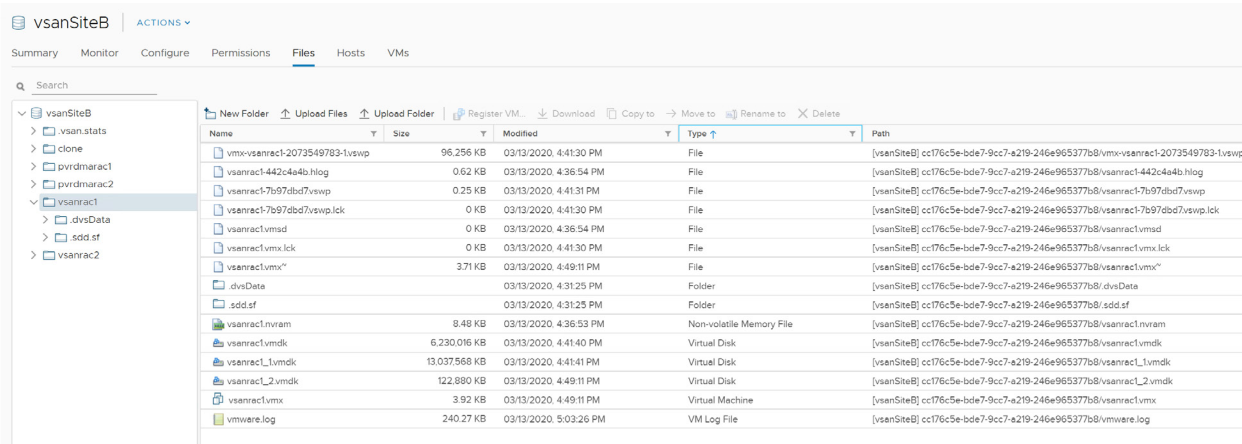


FIGURE 57. ORACLE RAC VM VSANRAC1 IN VSAN DATASTORE VSANSITEB

The Oracle RAC VM **vsanrac2** VMDKs are shown below in vSAN datastore **vsanSiteB**:

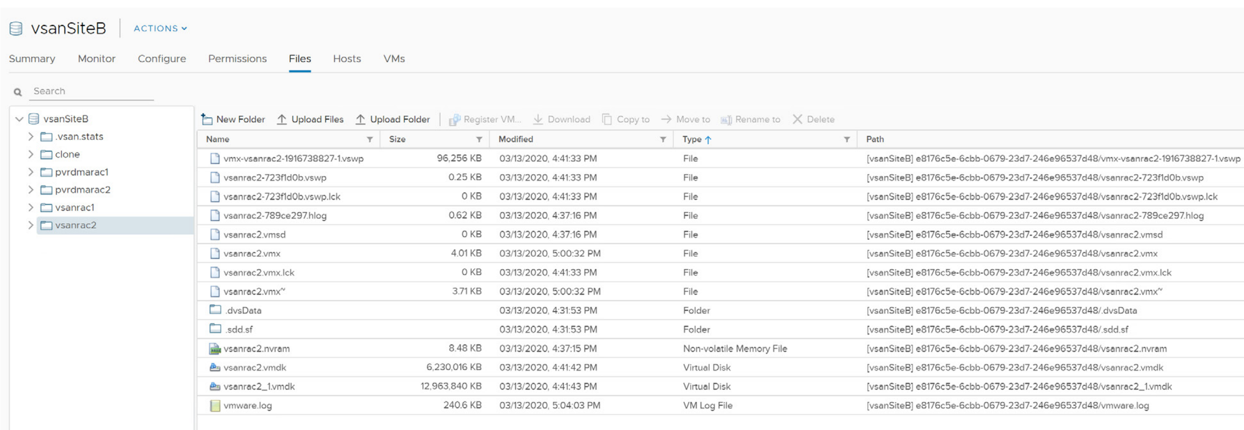


FIGURE 58. ORACLE RAC VM VSANRAC2 IN VSAN DATASTORE VSANSITEB

Note that the only VMDKs created on Oracle RAC VM **vsanrac2** folder are the OS and Oracle binaries as indicated below:

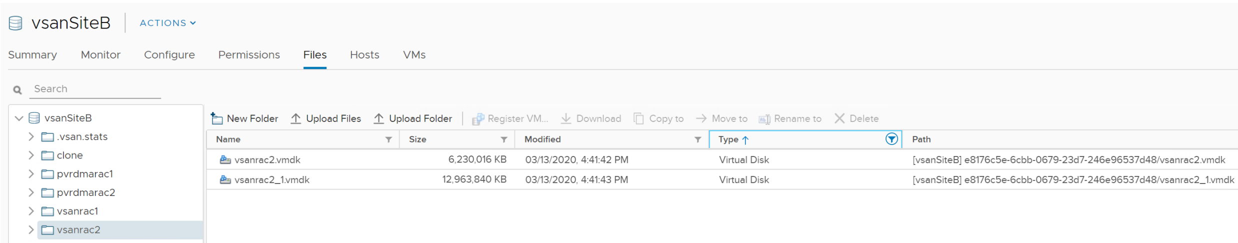


FIGURE 59. OS AND ORACLE BINARIES

Another way to see the same data is to check the contents of the .vmx file for Oracle RAC VM **vsanrac2** as below. Note that the only VMDKs created for Oracle RAC VM **vsanrac2** are the OS and Oracle binaries, with remaining disks referring to the shared disks created on Oracle RAC VM **vsanrac1**.

```
[root@wdc-esx30:/vmfs/volumes/vsan:5283c85154c8d44f-d969fa7d157edc33/e8176c5e-6cbb-0679-23d7-246e96537d48] cat vsanrac2.vmx | grep -i scsi | grep -i filename
scsi0:0.fileName = "vsanrac2.vmdk"
scsi0:1.fileName = "vsanrac2_1.vmdk"
scsi1:0.fileName = "/vmfs/volumes/vsan:5283c85154c8d44f-d969fa7d157edc33/cc176c5e-bde7-9cc7-a219-246e965377b8/vsanrac1_2.vmdk"
[root@wdc-esx30:/vmfs/volumes/vsan:5283c85154c8d44f-d969fa7d157edc33/e8176c5e-6cbb-0679-23d7-246e96537d48]
```

FIGURE 60. ORACLE RAC VM VSANRAC2 .VMX FILE

Learn more about *Oracle RAC on VMware vSAN*.

Oracle RAC Storage on vVol datastore

This section outlines the steps necessary to add a shared VMDK with the multi-writer attribute, as an Oracle ASM disk to an Oracle RAC 19c cluster, using a vVol datastore **TSA_PURE_FLASH_VVOL** backed by Pure x50 Storage.

The remaining steps to deploy an Oracle 19c RAC cluster are described in the **Oracle RAC Deployment High Level Steps** section of this document.

The Pure Storage vVol datastore **TSA_PURE_FLASH_VVOL** is shown below:

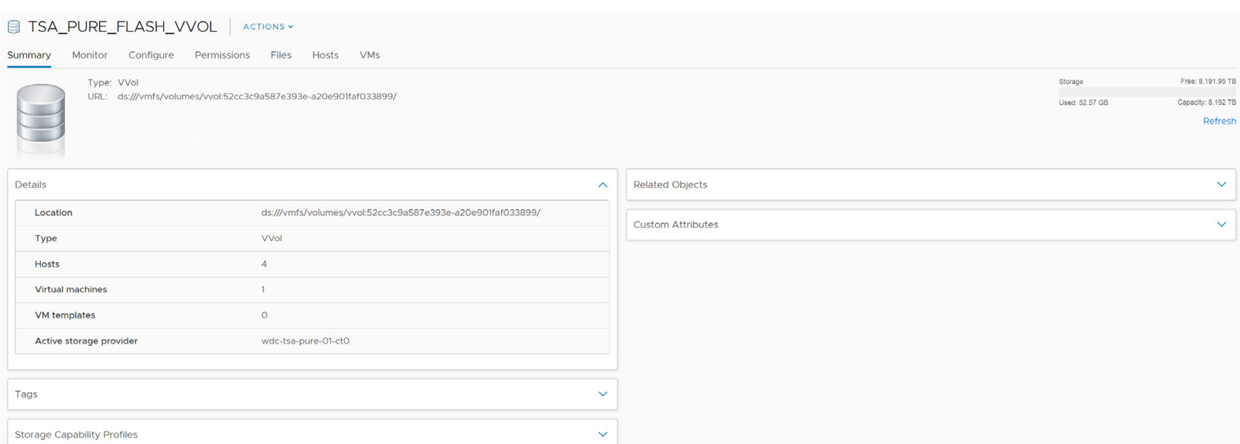


FIGURE 61. PURE STORAGE VVOL DATASTORE TSA_PURE_FLASH_VVOL

The shared storage provisioning guideline for vVol datastore is detailed below:

VMware Platform	Datastore	Version	Oracle RAC shared VMDK requirement for multi-writer attribute	Reference
VMware vSphere	vVol	ESXi 6.5 and later	Vendor specific	*KB 1034165* and *KB 2113013*

TABLE 24. VVOL DATASTORE SHARED STORAGE PROVISIONING GUIDELINE

It is recommended to work with a storage-specific vendor regarding the correct guidance for implementing VMware vVols in that vendor's environment, as each storage-vendor implementation of vVols is often different.

From the Pure FlashArray User Guide (found in the **Help** section under **FlashArray Storage Capacity and Utilization** in the Pure FlashArray GUI):

Provisioning

The provisioned size of a volume is its capacity as reported to hosts. As with conventional disks, the size presented by a FlashArray volume is nominally fixed, although it can be increased or decreased by an administrator. To optimize physical storage utilization, however, FlashArray volumes are thin and micro provisioned.

- **Thin provisioning**: like conventional arrays that support thin provisioning, FlashArrays do not allocate physical storage for volume sectors that no host has ever written, or for trimmed (expressly deallocated by host or array administrator command) sector addresses.
- **Micro provisioning**: unlike conventional thin provisioning arrays, FlashArrays allocate only the exact amount of physical storage required by each host-written block after reduction. In FlashArrays, there is no concept of allocating storage in chunks of some fixed size.

Observation

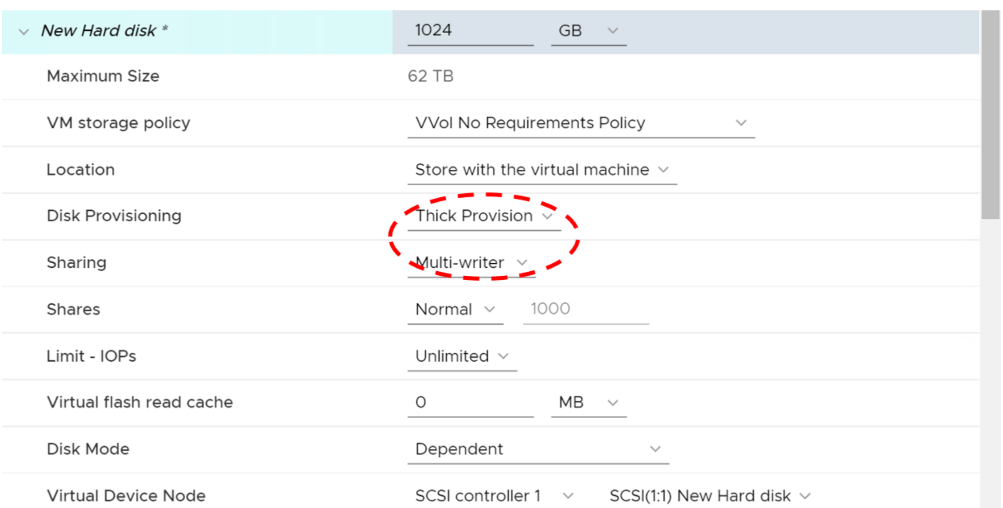


FIGURE 62. EZT VMDK CREATION/MIGRATION ON PURE STORAGE FLASHARRAY WITH VVOLS

When attempting to create or migrate an EZT VMDK on the Pure Storage FlashArray, utilizing vVol technology, the following error is reported: "Error creating disk Error creating VVol Object. This may be due to insufficient available space on the datastore or the datastore's inability to support the selected provisioning type."

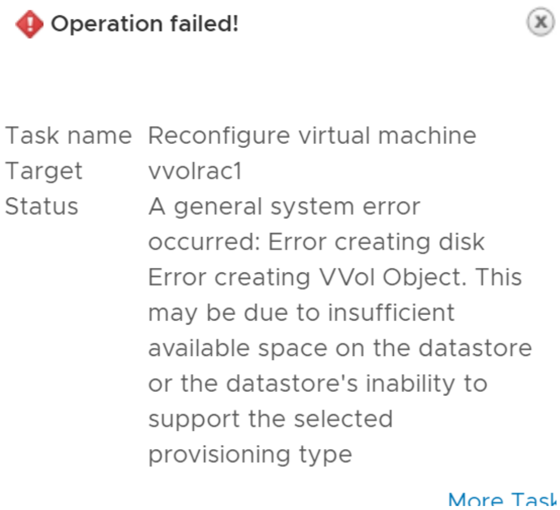


FIGURE 63. EZT VMDK ERROR ON PURE STORAGE FLASHARRAY WITH VVOLS

Reason: The error is due to the FlashArray rejecting the request for a thick-provisioned VM upon creation or migration to the FlashArray. The Pure Storage FlashArray utilizes both thin and micro provisioning technologies for space management and is unable to reserve unused space when thick provisioned VMs are created. Thus, it was decided to disallow that request to ensure administrators are able to properly manage and account for space usage on the FlashArray.

Allowing a FlashArray or vSphere administrator to create thick provisioned VMs on vVols can present problems later if admins are expecting that space to be reserved when it has not been properly reinforced. To ensure FlashArray or vSphere administrators are able to properly account for space, allow thin-provisioned VMs are allowed when utilizing vVols.

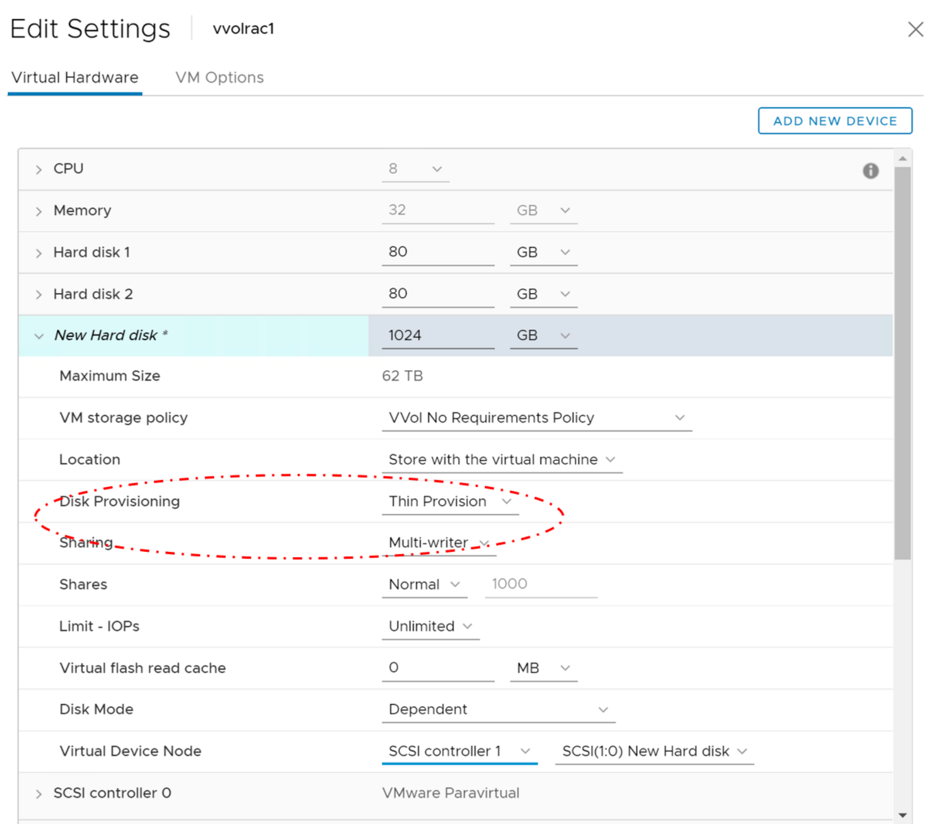
Learn more on this topic in the Pure Storage support solution *vVol: Why am I unable to create a thick provisioned VM?*

Conclusion: The steps required to add a shared EZT VMDK using the multi-writer attribute, to an Oracle RAC cluster using a Pure Storage Flash Array x50 vVol datastore, is the same as those for adding a Shared EZT VMDK using the multi-writer attribute on a FC VMFS datastore. The difference is the shared disk on the Pure Storage Flash Array x50 vVol datastore has to be thin-provisioned with the multi-writer attribute. All other settings remain the same.

Below are the steps required to add a 1TB VMDK as a shared Oracle ASM disk at SCSI position 1:0 to Oracle RAC VMs **vvolrac1** and **vvolrac2**, provisioned from the vVol datastore **TSA_PURE_FLASH_VVOL**.

1. Right click Oracle RAC VM **vvolrac1** and click **Edit Settings**
2. Click **Add New Device**
3. Click **Hard Disk** and provision the shared VMDK as shown below:
   – Set the correct VMDK size (in this case 1TB)
   – Leave the VM storage policy to default (**VVol No Requirements Policy**)
   – Leave the VM storage policy to default (**Store with the virtual machine**)

– Set the **Disk Provisioning** to **Thin Provision** for the reasons stated above
– Set the **Sharing** to **Multi-writer**
– Set the **Virtual Device Node** to SCSI position SCSI1:1
– Independent persistent mode is **not** required for enabling the multi-writer attribute
– Click **OK** to save



4. Ensure the following settings are indicated once the VMDK is created:
– Name of VMDK is **[TSA_PURE_FLASH_VVOL] rfc4122.fba904a5-0488-4900-a85a-efe871b831e0/vvolrac1_2.vmdk**
– VMDK is provisioned on SCSI Controller SCSI1:0 position
– VMDK is **Thin Provisioned**
– VMDK **Sharing** is set to **Multi-writer**
– **Disk Mode** is set to the default **Dependent**

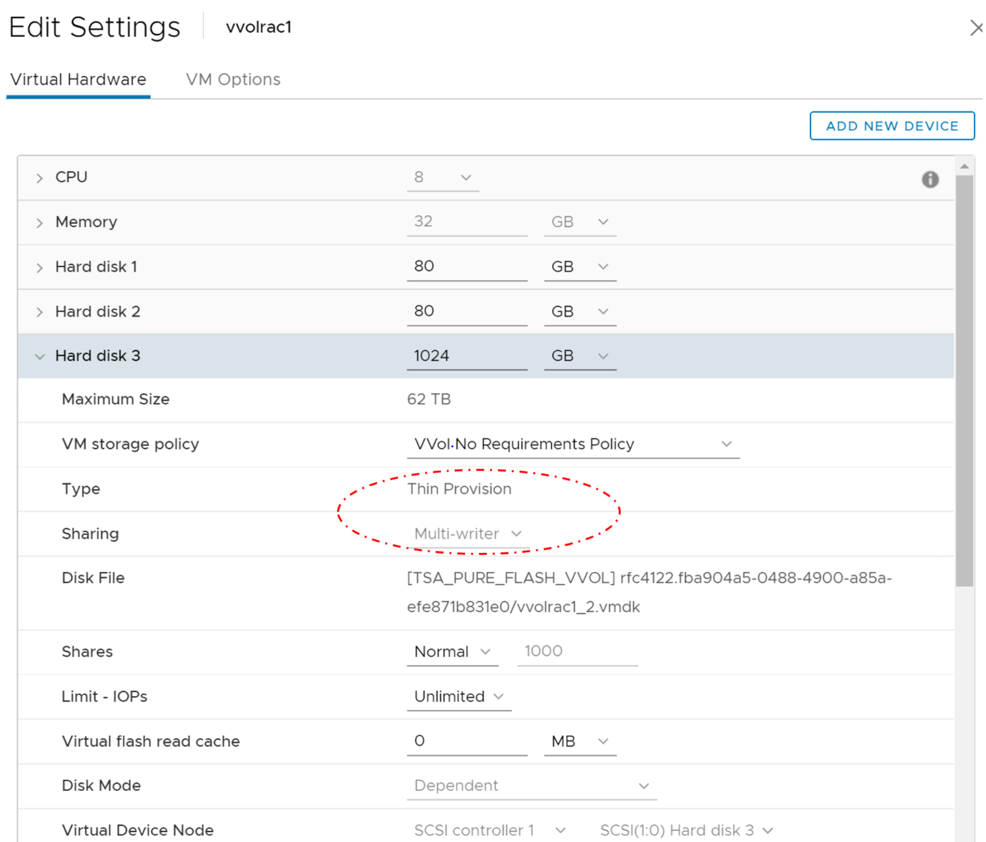
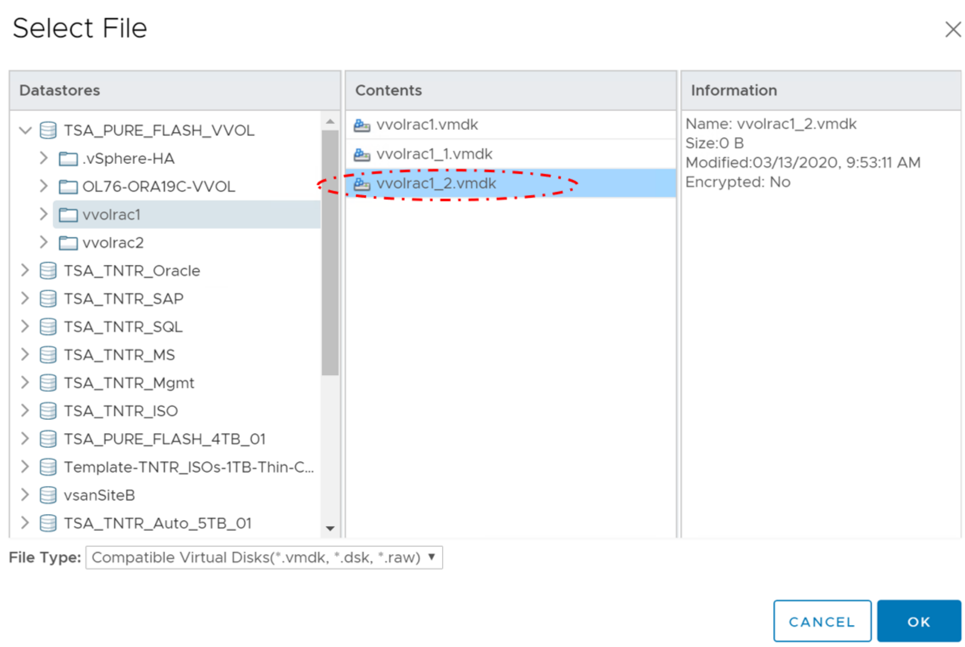
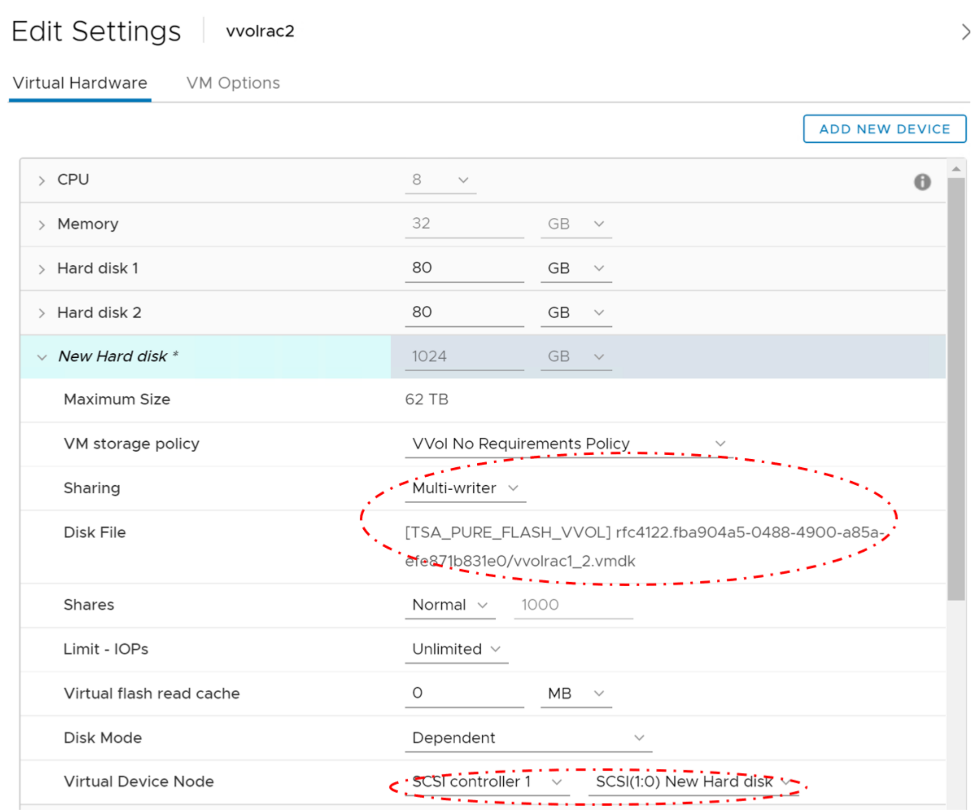


FIGURE 64. PVSCSI CONTROLLER BACKED ORACLE ASM DATA VMDK

5. For VM **vvolrac2**, add the same 1TB VMDK **[TSA_PURE_FLASH_VVOL] rfc4122.fba904a5-0488-4900-a85a-efe871b831e0/vvolrac1_2.vmdk** created for Oracle ASM disk on VM **vvolrac2** to PVSCSI controller on SCSI position 1:0.
6. Repeat steps **one** and **two** as previously completed for VM **vvolrac1**
7. For step **three**, instead of choosing **Hard Disk**, choose **Existing Hard Disk**
8. Navigate to the VM vvolrac1 folder on **TSA_PURE_FLASH_VVOL** datastore and select **vvolrac1_2.vmdk** 1TB VMDK. Click **OK**.



9. Provision the VMDK on the same SCSI Controller SCSI1:0 position, as is the case for Oracle RAC VM **vvolrac1**. Follow steps **four** and **five** and click **OK** to save.



10. Ensure the following settings are reflected:
    – Name of VMDK is **[TSA_PURE_FLASH_VVOL] rfc4122.fba904a5-0488-4900-a85a-efe871b831e0/vvolrac1_2.vmdk**
    – VMDK is provisioned on SCSI Controller SCSI1:0 position
    – VMDK is **Thin Provisioned**
    – VMDK **Sharing** is set to **Multi-writer**
    – **Disk Mode** is set to the default **Dependent**

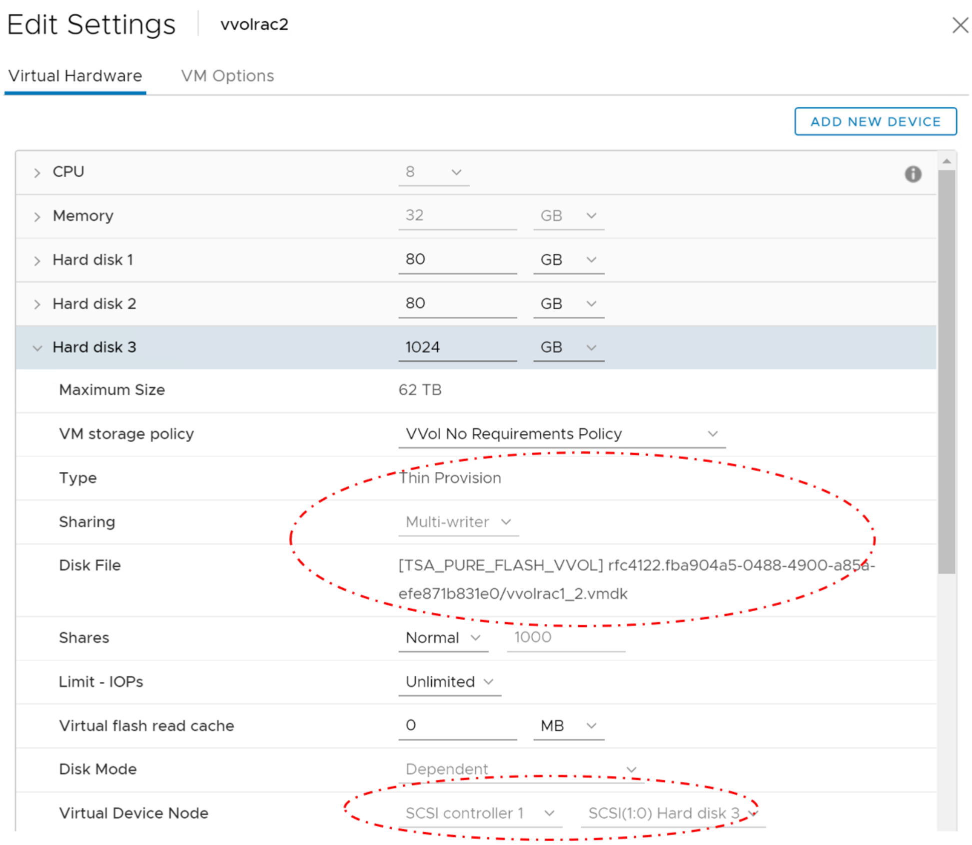
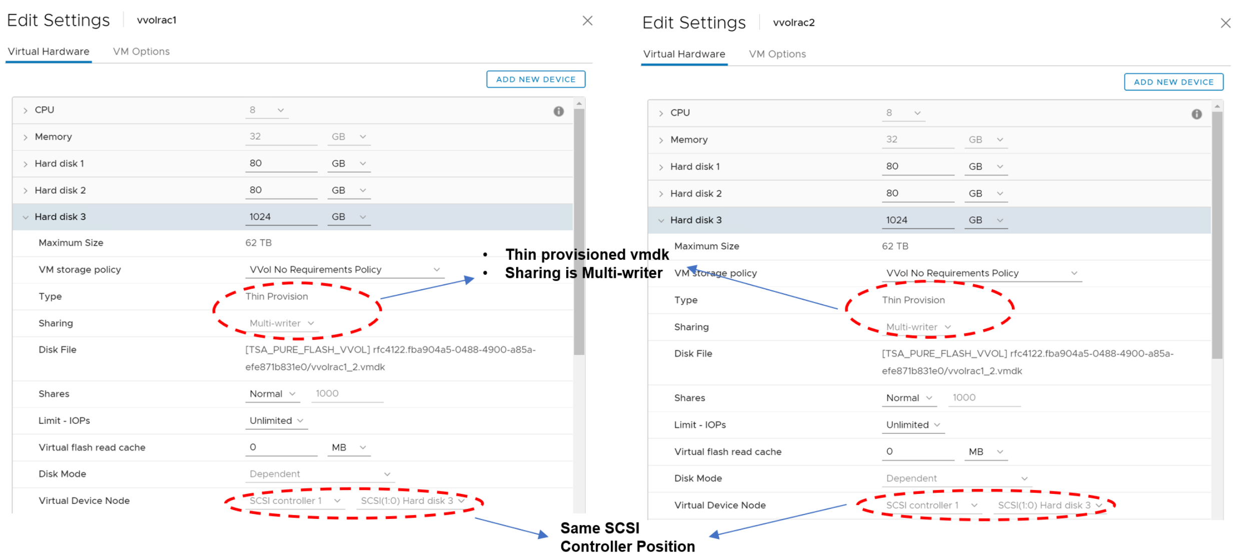


FIGURE 65. PVSCSI CONTROLLER BACKED ORACLE ASM DATA VMDK

11. The freshly provisioned 1TB shared VMDK on Oracle RAC VM **vvolrac1** and **vvolrac2** is shown below. Note that Oracle RAC VM **vvolrac2** 1TB VMDK is referring to Oracle RAC VM **vvolrac1** VMDK.



12. Repeat the steps above to provision the remaining Oracle RAC shared disks.

The Oracle RAC VM **vvolrac1** VMDKs are shown below in the vVol datastore **TSA_PURE_FLASH_VVOL**:

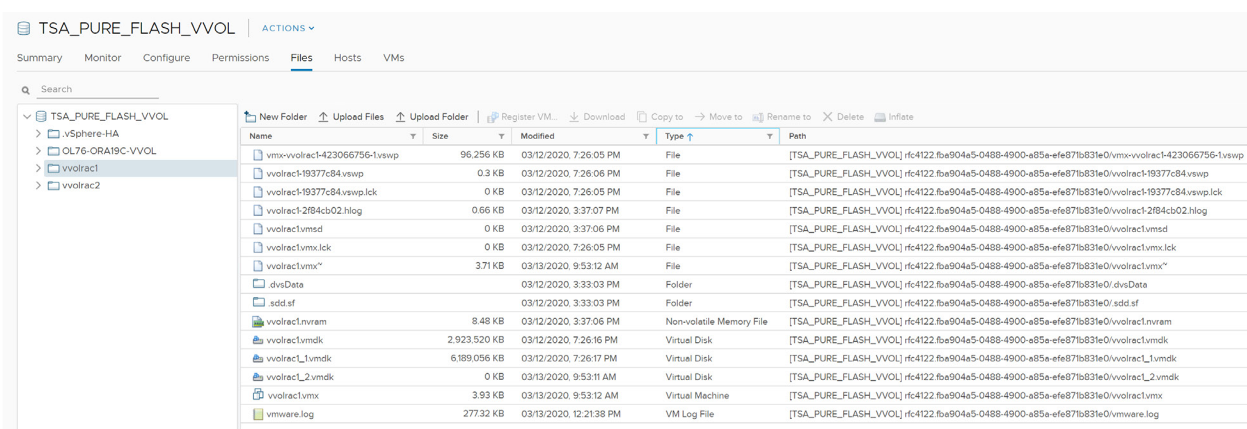


FIGURE 66. ORACLE RAC VM VVOLRAC1 IN VVOL DATASTORE TSA_PURE_FLASH_VVOL

The Oracle RAC VM **vvolrac2** VMDKs are shown below in the vVol datastore **TSA_PURE_FLASH_VVOL**:

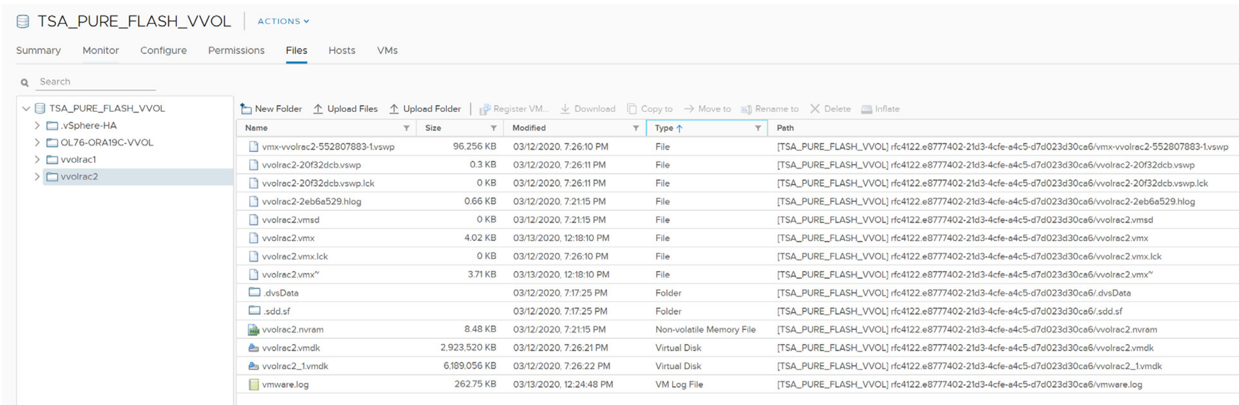


FIGURE 67. ORACLE RAC VM VVOLRAC2 IN VVOL DATASTORE TSA_PURE_FLASH_VVOL

Note that the only VMDKs created on Oracle RAC VM **vvolrac2** folder are the OS and Oracle binaries as indicated below. Oracle RAC VM **vvolrac1** contains all three VMDKs, the OS, Oracle binaries, and Oracle database VMDKs.

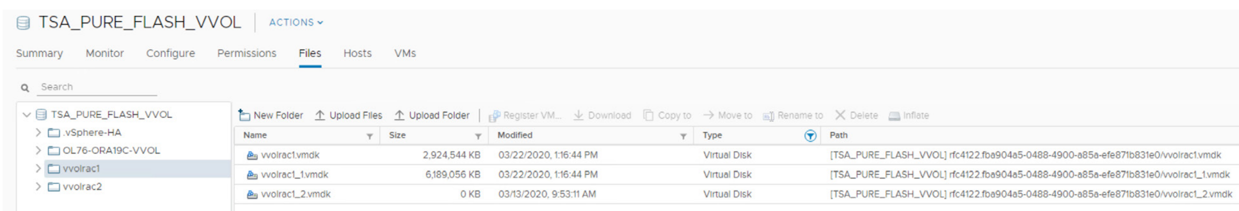


FIGURE 68. ORACLE RAC VM VVOLRAC1 FOLDER

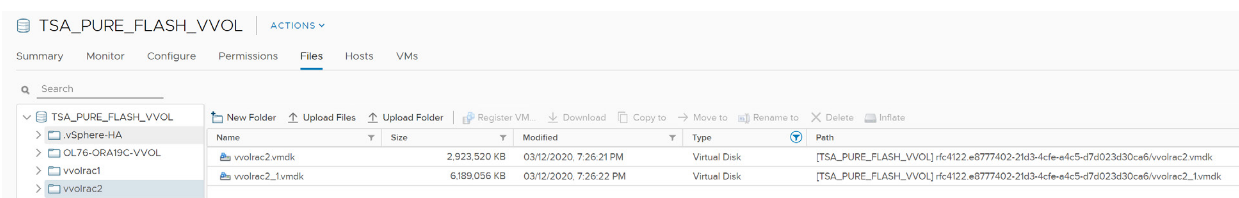


FIGURE 69. ORACLE RAC VM VVOLRAC2 FOLDER

Another way to see the same data is to check the contents of the .vmx file for Oracle RAC VM **vvolrac2** as below. Observe that the only VMDKs created for Oracle RAC VM **vvolrac2** are the OS and Oracle binaries and the rest of disks are merely referencing to the shared disks created on Oracle RAC VM **vvolrac1**.

```
[root@wdc-esx30:/vmfs/volumes/vvol:52cc3c9a587e393e-a20e901faf033899/rfc4122.e8777402-21d3-4cfe-a4c5-d7d023d30ca6] cat vvolrac2.vmx | grep -i scsi | grep -i filename
scsi0:0.fileName = "vvolrac2.vmdk"
scsi0:1.fileName = "vvolrac2_1.vmdk"
scsi1:0.fileName = "/vmfs/volumes/vvol:52cc3c9a587e393e-a20e901faf033899/rfc4122.fba904a5-0488-4900-a85a-efe871b831e0/vvolrac1_2.vmdk"
[root@wdc-esx30:/vmfs/volumes/vvol:52cc3c9a587e393e-a20e901faf033899/rfc4122.e8777402-21d3-4cfe-a4c5-d7d023d30ca6]
```

FIGURE 70. ORACLE RAC VM VVOLRAC2 .VMX FILE

It's important to note that following an SSH to the ESXi server and navigation to the vVol datastore mount to check both Oracle RAC VMs vvolrac1 and **vvolrac2** folders, there are no **XXXX-flat.vmdk**, we only see the .vmdk files to be found.

The VMDK file contents point to the actual vVol, which backs the **XXXX-flat.vmdk**:

```
[root@wdc-esx30:/vmfs/volumes/vvol:52cc3c9a587e393e-a20e901faf033899/rfc4122.fba904a5-0488-4900-a85a-efe871b831e0] pwd
/vmfs/volumes/TSA_PURE_FLASH_VVOL/vvolrac1
[root@wdc-esx30:/vmfs/volumes/vvol:52cc3c9a587e393e-a20e901faf033899/rfc4122.fba904a5-0488-4900-a85a-efe871b831e0] ls -l *.vmdk
-rw-------    1 root     root           620 Mar 22 20:16 vvolrac1.vmdk
-rw-------    1 root     root           566 Mar 22 20:16 vvolrac1_1.vmdk
-rw-------    1 root     root           545 Mar 13 16:53 vvolrac1_2.vmdk
[root@wdc-esx30:/vmfs/volumes/vvol:52cc3c9a587e393e-a20e901faf033899/rfc4122.fba904a5-0488-4900-a85a-efe871b831e0] cat vvolrac1.vmdk
# Disk DescriptorFile
version=4
encoding="UTF-8"
CID=25e0fb03
parentCID=ffffffff
createType="vmfs"

# Extent description
RW 167772160 VMFS "vvol://52cc3c9a587e393e-a20e901faf033899/rfc4122.86f18bd8-9103-473d-9c80-f61fa24797ee"

# The Disk Data Base
#DDB

ddb.adapterType = "lsilogic"
ddb.deletable = "true"
ddb.geometry.cylinders = "10443"
ddb.geometry.heads = "255"
ddb.geometry.sectors = "63"
ddb.longContentID = "5546172c14f11ec5bb13bcf225e0fb03"
ddb.thinProvisioned = "1"
ddb.toolsInstallType = "4"
ddb.toolsVersion = "10336"
ddb.uuid = "60 00 C2 92 0c a5 bd b9-3f 82 3a 31 b7 01 a7 a5"
ddb.virtualHWVersion = "14"
[root@wdc-esx30:/vmfs/volumes/vvol:52cc3c9a587e393e-a20e901faf033899/rfc4122.fba904a5-0488-4900-a85a-efe871b831e0]
```

FIGURE 71. ORACLE RAC VMDK FILE

From a Pure Storage volume perspective, administrators can see different vVols being created for both Oracle RAC VMs:

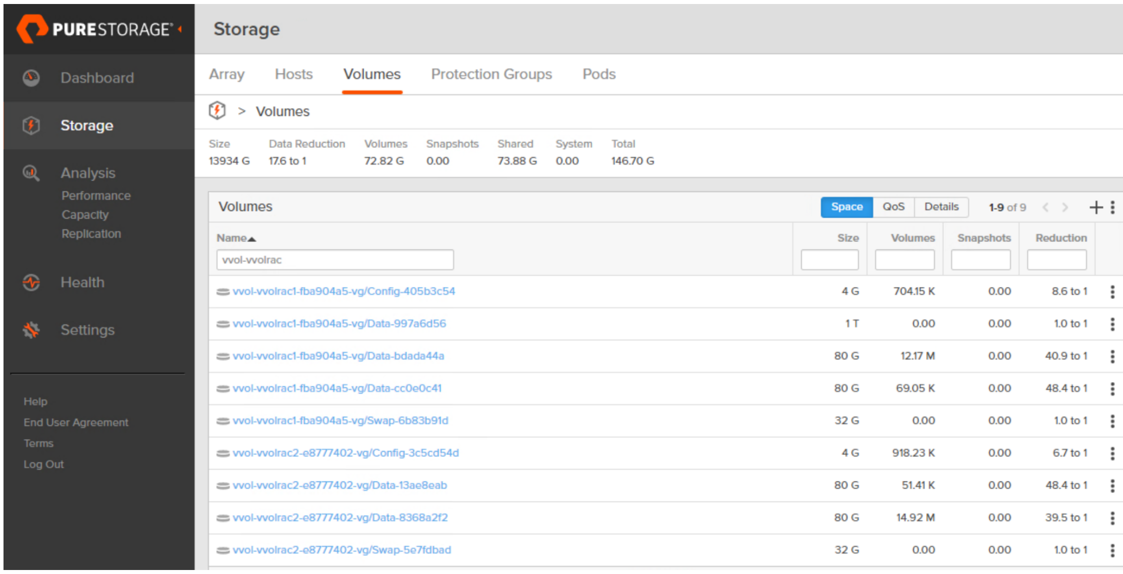


FIGURE 72. PURE STORAGE VOLUME

This is in contrast to an Oracle RAC VM **orac19c1** on FC VMFS datastore where both **-flat.vmdk** and **.vmdk** files are visible:

```
[root@wdc-esx30:/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1] pwd
/vmfs/volumes/TSA_PURE_FLASH_4TB_01/orac19c1
[root@wdc-esx30:/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1] ls -l *.vmdk
-rw-------    1 root     root     85899345920 Mar 28 22:33 orac19c1-flat.vmdk
-rw-------    1 root     root             553 Mar 10 20:20 orac19c1.vmdk
-rw-------    1 root     root     85899345920 Mar 28 22:33 orac19c1_1-flat.vmdk
-rw-------    1 root     root             501 Mar 10 20:20 orac19c1_1.vmdk
-rw-------    1 root     root     53687091200 Mar  7 02:24 orac19c1_10-flat.vmdk
-rw-------    1 root     root             452 Mar 12 03:26 orac19c1_10.vmdk
-rw-------    1 root     root     1099511627776 Mar  7 01:37 orac19c1_2-flat.vmdk
-rw-------    1 root     root             454 Mar 12 03:26 orac19c1_2.vmdk
-rw-------    1 root     root     268435456000 Mar  7 02:12 orac19c1_3-flat.vmdk
-rw-------    1 root     root             452 Mar 12 03:26 orac19c1_3.vmdk
-rw-------    1 root     root     268435456000 Mar  7 02:13 orac19c1_4-flat.vmdk
-rw-------    1 root     root             452 Mar 12 03:26 orac19c1_4.vmdk
-rw-------    1 root     root     5368709120 Mar  7 02:24 orac19c1_5-flat.vmdk
-rw-------    1 root     root             449 Mar 12 03:25 orac19c1_5.vmdk
-rw-------    1 root     root     5368709120 Mar  7 02:24 orac19c1_6-flat.vmdk
-rw-------    1 root     root             449 Mar 12 03:25 orac19c1_6.vmdk
-rw-------    1 root     root     5368709120 Mar  7 02:24 orac19c1_7-flat.vmdk
-rw-------    1 root     root             449 Mar 12 03:25 orac19c1_7.vmdk
-rw-------    1 root     root     5368709120 Mar  7 02:24 orac19c1_8-flat.vmdk
-rw-------    1 root     root             449 Mar 12 03:25 orac19c1_8.vmdk
-rw-------    1 root     root     5368709120 Mar  7 02:24 orac19c1_9-flat.vmdk
-rw-------    1 root     root             449 Mar 12 03:25 orac19c1_9.vmdk
[root@wdc-esx30:/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1] cat orac19c1.vmdk
# Disk DescriptorFile
version=1
encoding="UTF-8"
CID=c67e051b
parentCID=ffffffff
createType="vmfs"

# Extent description
RW 167772160 VMFS "orac19c1-flat.vmdk"

# The Disk Data Base
#DDB

ddb.adapterType = "lsilogic"
ddb.deletable = "true"
ddb.geometry.cylinders = "10443"
ddb.geometry.heads = "255"
ddb.geometry.sectors = "63"
ddb.longContentID = "18cbf71965e9999a222ae369c67e051b"
ddb.thinProvisioned = "1"
ddb.toolsInstallType = "4"
ddb.toolsVersion = "10336"
ddb.uuid = "60 00 C2 9b 01 a6 0f 48-b7 b4 e8 08 2a 2d 46 6e"
ddb.virtualHWVersion = "14"
[root@wdc-esx30:/vmfs/volumes/5df1b105-84793f90-4b84-001b216fdbec/orac19c1]
```

FIGURE 73. ORACLE RAC VM ORAC19C1 ON FC VMFS DATASTORE

Below is an example of an Oracle RAC VM **rac19c1** VMDK file on NFS datastore where both **-flat.vmdk** and **.vmdk** files are visible:

```
[root@wdc-esx30:/vmfs/volumes/5484b08f-21c26477/rac19c1] pwd
/vmfs/volumes/TSA_TNTR_Oracle/rac19c1
[root@wdc-esx30:/vmfs/volumes/5484b08f-21c26477/rac19c1] ls -l *.vmdk
-rw-------    1 root     root     64424509440 Mar 28 23:11 rac19c1-flat.vmdk
-rw-------    1 root     root             550 Mar 13 22:38 rac19c1.vmdk
-rw-------    1 root     root     64424509440 Mar 28 23:11 rac19c1_1-flat.vmdk
-rw-------    1 root     root             552 Mar 13 22:38 rac19c1_1.vmdk
-rw-------    1 root     root     1099511627776 Mar 28 23:11 rac19c1_2-flat.vmdk
-rw-------    1 root     root             476 Mar 13 22:39 rac19c1_2.vmdk
[root@wdc-esx30:/vmfs/volumes/5484b08f-21c26477/rac19c1] cat rac19c1.vmdk
# Disk DescriptorFile
version=1
encoding="UTF-8"
CID=d4eff194
parentCID=ffffffff
createType="vmfs"

# Extent description
RW 125829120 VMFS "rac19c1-flat.vmdk"

# The Disk Data Base
#DDB

ddb.adapterType = "lsilogic"
ddb.deletable = "true"
ddb.geometry.cylinders = "7832"
ddb.geometry.heads = "255"
ddb.geometry.sectors = "63"
ddb.longContentID = "01eb354741155b84593415aed4eff194"
ddb.thinProvisioned = "1"
ddb.toolsInstallType = "4"
ddb.toolsVersion = "10309"
ddb.uuid = "60 00 C2 98 d1 02 5d e8-c3 28 1e de e4 f9 ef 7b"
ddb.virtualHWVersion = "4"
[root@wdc-esx30:/vmfs/volumes/5484b08f-21c26477/rac19c1]
```

FIGURE 74. ORACLE RAC VM RAC19C1 VMDK FILE

Oracle RAC Storage on In-Guest iSCSI, NFS and dNFS Storage

The following table shows the storage options supported for Oracle Grid Infrastructure binaries and files:

Storage Option	OCR and Voting Files	Oracle Clusterware Binaries	Oracle RAC Database Binaries	Oracle RAC Database Data Files	Oracle RAC Database Recovery Files
Oracle Automatic Storage Management (Oracle ASM) **Note**: Loopback devices are not supported for use with Oracle ASM	Yes	No	No	Yes	Yes
Oracle Automatic Storage Management Cluster File System (Oracle ACFS)	No	No	Yes	Yes for Oracle Database 12c Release 1 (12.1) and later	Yes for Oracle Database 12c Release 1 (12.1) and later
Local file system	No	Yes	Yes	No	No
OCFS2	No	No	Yes	Yes	Yes
Network file system (NFS) on a certified network-attached storage (NAS) filer **Note:** Direct NFS Client does not support Oracle Clusterware files	Yes	Yes	Yes	Yes	Yes
Direct-attached storage (DAS)	No	No	Yes	Yes	Yes
Shared disk partitions (block devices or raw devices)	No	No	No	No	No

FIGURE 75. SUPPORTED STORAGE OPTIONS FOR ORACLE GRID INFRASTRUCTURE

Use the following guidelines when choosing storage options:

- Choose any combination of the supported storage options for each file type provided all requirements listed for the chosen storage options are satisfied.
- Use Oracle ASM or shared file system to store Oracle Clusterware files.
- Direct use of raw or block devices is not supported. Only use raw or block devices under Oracle ASM are allowed.

Learn more about *supported storage types*.

Guidelines for configuring Oracle RAC on in-guest storage can be found below:

- Guidelines for configuring Oracle RAC storage using in-guest NFS storage can be found *here* and *here*.
- Guidelines for configuring Oracle RAC storage using in-guest Oracle Direct NFS storage (dNFS) can be found *here*.
- Guidelines for configuring Oracle RAC storage using in-guest iSCSI storage can be found *here*.

Discussion of Oracle RAC storage using in-guest NFS/dNFS /iSCSI is beyond the scope of this paper.

Oracle RAC Storage on Raw Device Mapping (Physical and Virtual)

This section outlines the steps required to add a shared physical RDM as an Oracle ASM disk, to an Oracle RAC 19c cluster, using a physical LUN from an FC SAN storage.

The remaining steps to deploy an Oracle RAC 19c cluster are described in the **Oracle RAC Deployment High Level Steps** section of this document.

Important observations need to be kept in mind include:

- SCSI bus-sharing ensures VM can register the keys for SCSI 3 persistent reservation.
- In order to use physical RDMs as shared storage for Oracle RAC, the multi-writer attribute should **not** be set as physical bus-sharing indirectly leads to disk opened in multi-writer mode.

The following steps should be followed when adding shared RDM(s) in physical compatibility mode to Oracle RAC VMs:

- Set **SCSI Bus Sharing** to **Physical** for those SCSI controllers to which the shared RDM(s) will be added.
- Set the compatibility mode for the shared RDM to **Physical** for physical compatibility mode.

Shown below are two Oracle RAC VMs **rdmrac1** and **rdmrac2**. Both VMs are running Oracle Enterprise Linux 7.6. Both VMs are on the Pure Storage FC datastore **TSA_PURE_FLASH_4TB_01**.

Oracle RAC VM **rdmrac1** and **rdmrac2** have four SCSI controllers each. SCSI controllers 0,1,2 and 3 are paravirtual controllers.

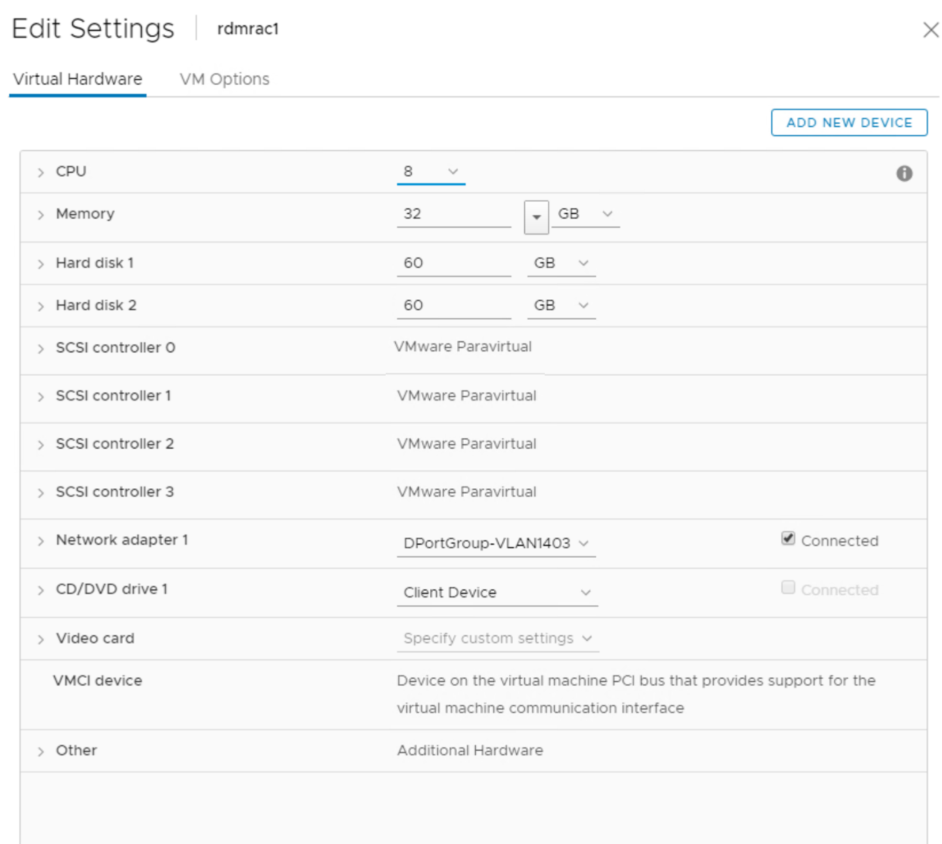
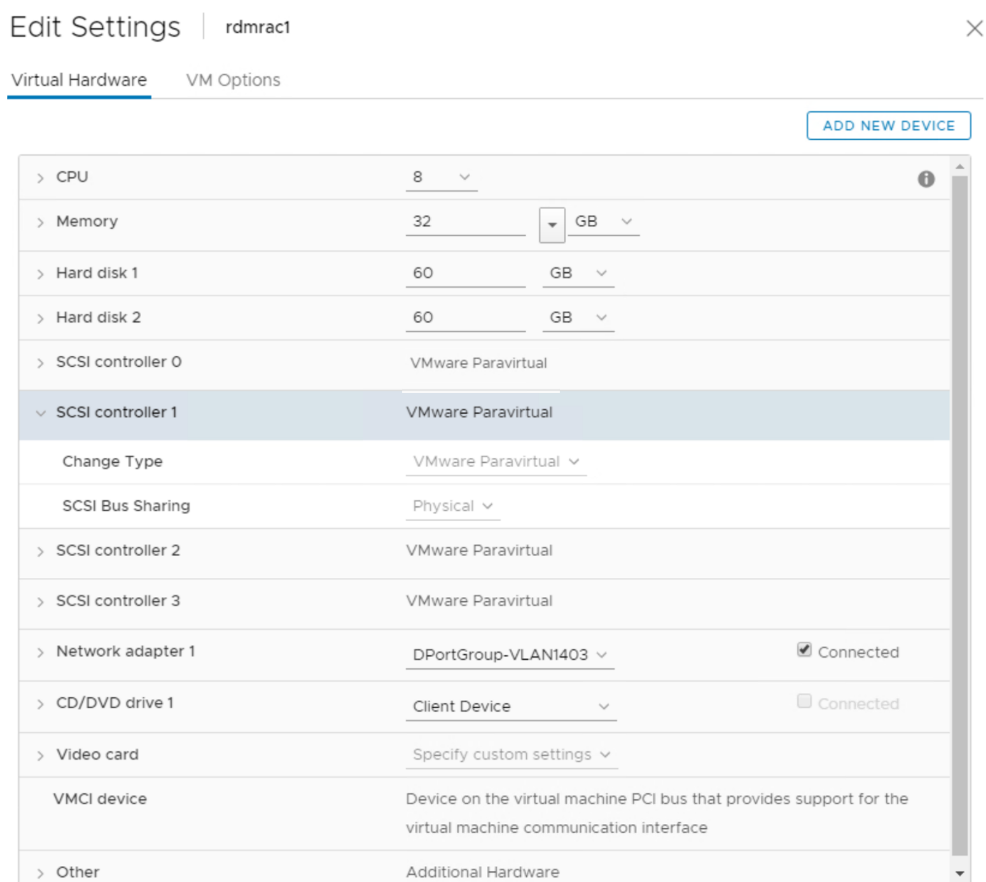


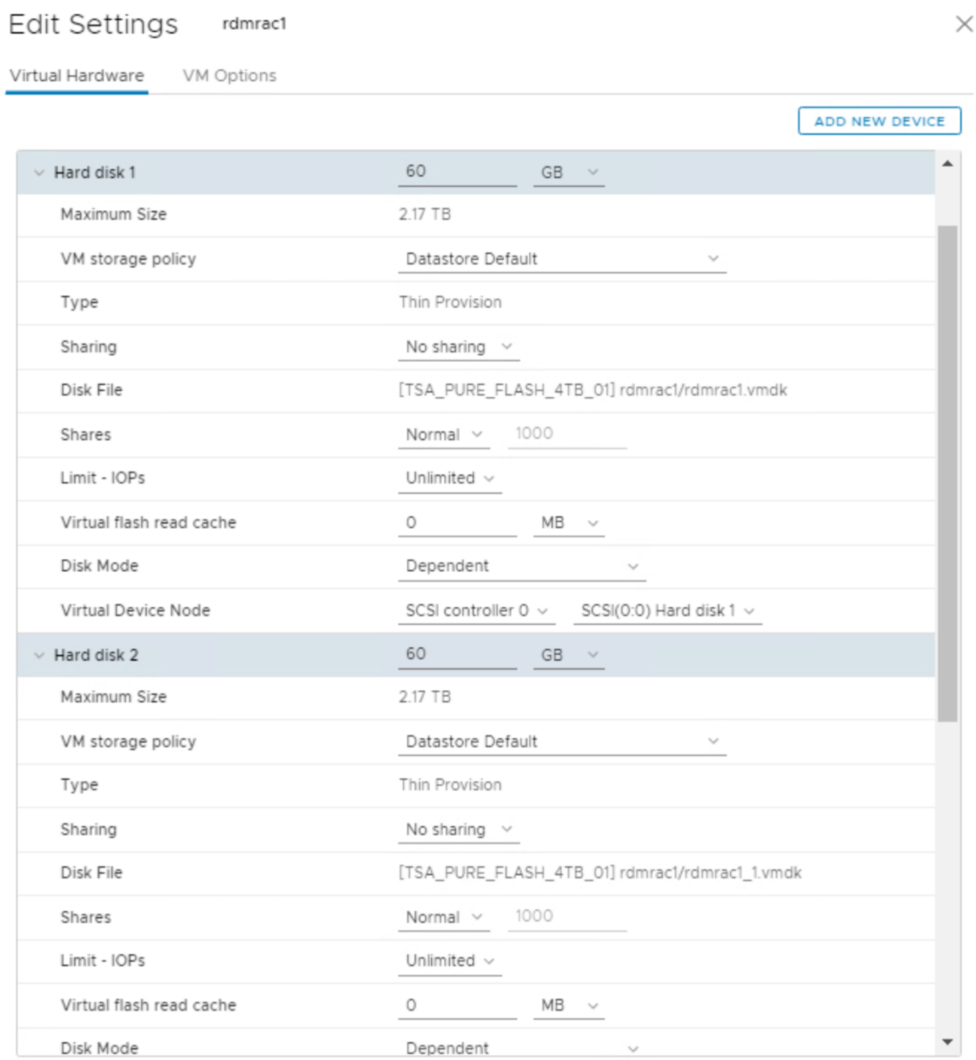
FIGURE 76. ORACLE RAC VM RDMRAC1

Oracle RAC VM **rdmrac1** SCSI controller one **SCSI Bus Sharing** is set to **Physical**.



The Oracle RAC VM **rdmrac1** disk setup as shown below:

- SCSI 0:0 for Operating System (/), size 60G on Pure Storage FC datastore **TSA_PURE_FLASH_4TB_01**
- SCSI 0:1 for Oracle binaries (/u01), size 60G on Pure Storage FC datastore **TSA_PURE_FLASH_4TB_01**



The Oracle RAC VM **rdmrac2** is set up similarly to Oracle RAC VM **rdmrac1**.

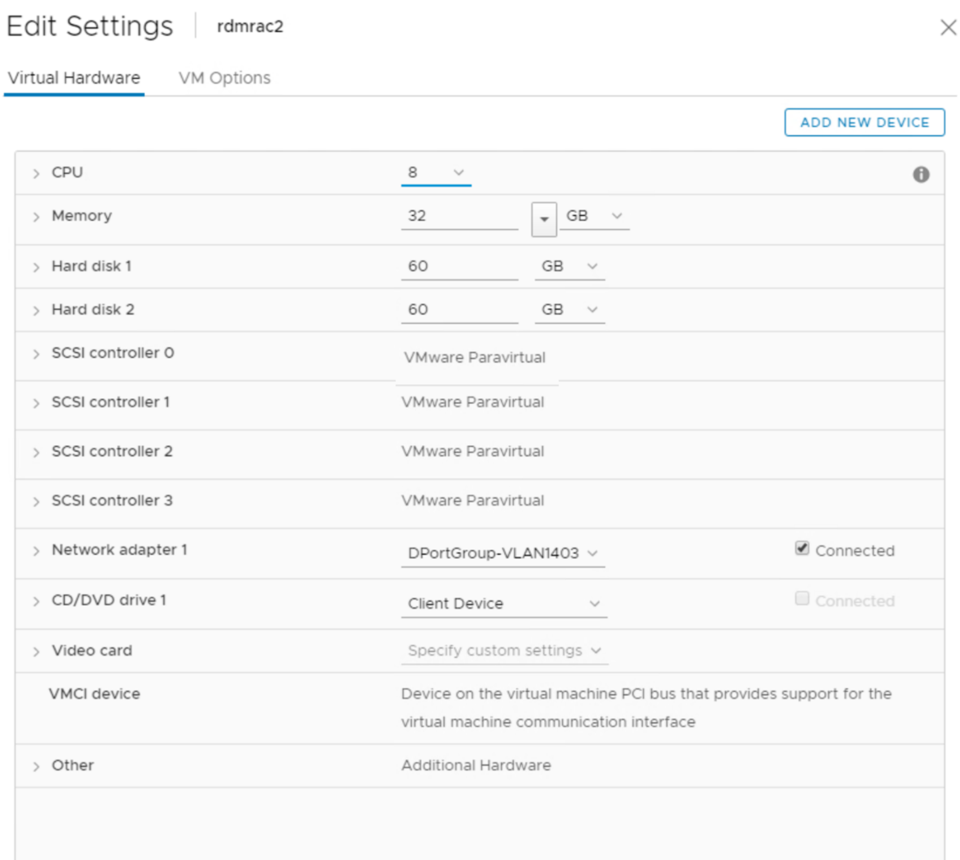
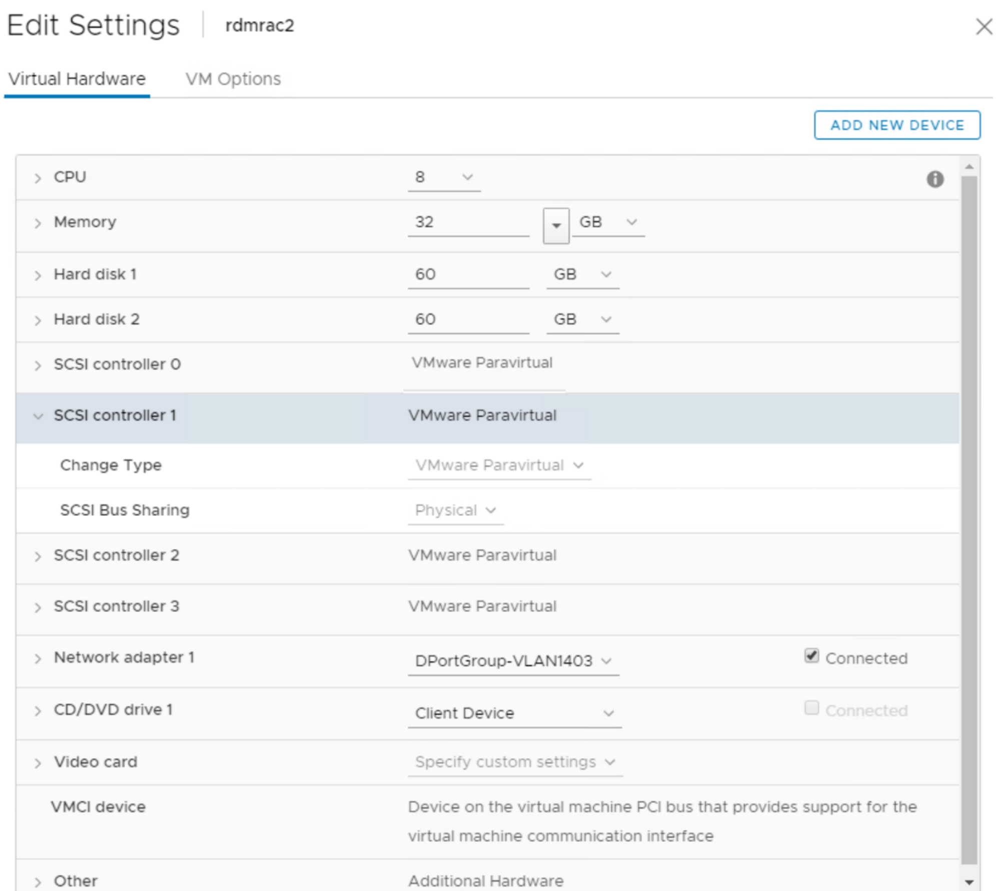


FIGURE 77. ORACLE RAC VM RDMRAC2

Oracle RAC VM **rdmrac2** SCSI controller one **SCSI Bus Sharing** is set to **Physical**



A Pure Storage LUN **TSA_PURE_FLASH_ORACLE_RDM_RAC_2TB_01** was created with size 1TB and mapped to the ESXi servers on Site B.

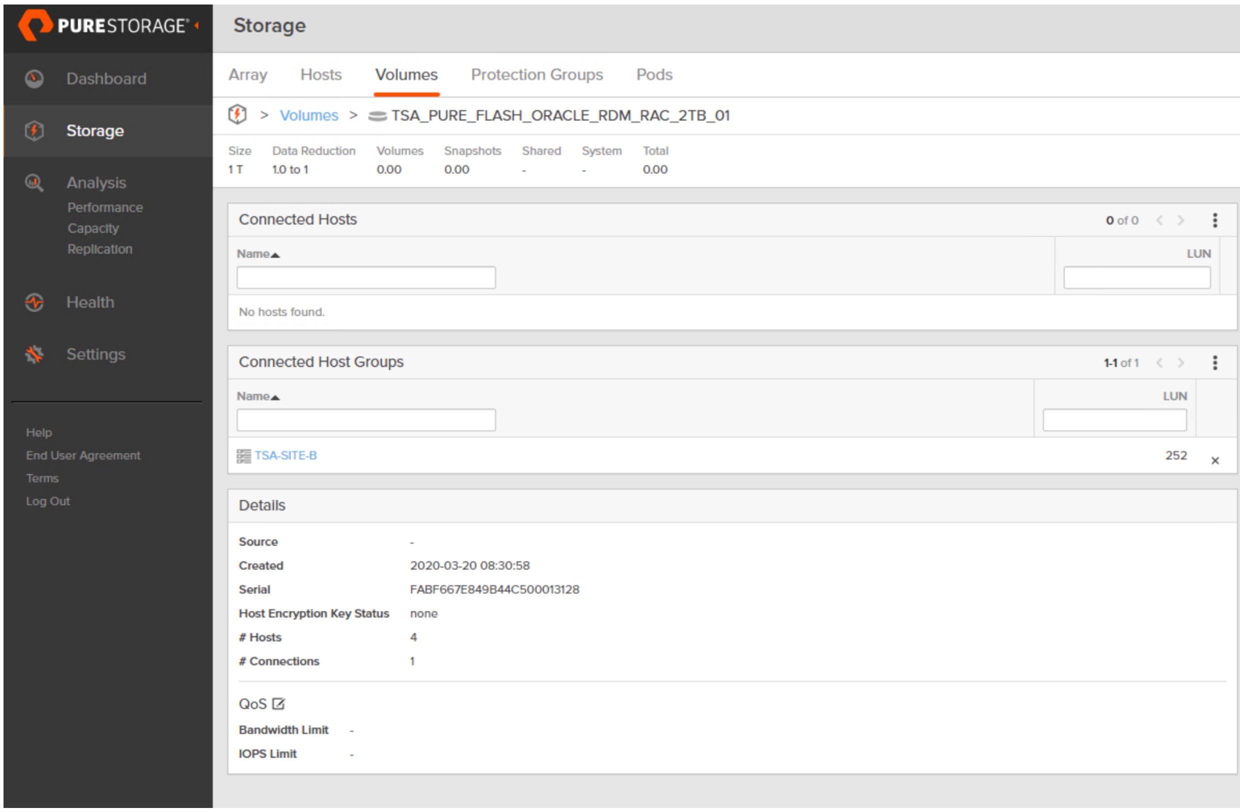


FIGURE 78. PURE STORAGE LUN TSA_PURE_FLASH_ORACLE_RDM_RAC_2TB_01

The Raw Device **PURE Fibre Channel Disk (naa.624a9370fabf667e849b44c500013128)** used in a physical compatibility mode is shown below:

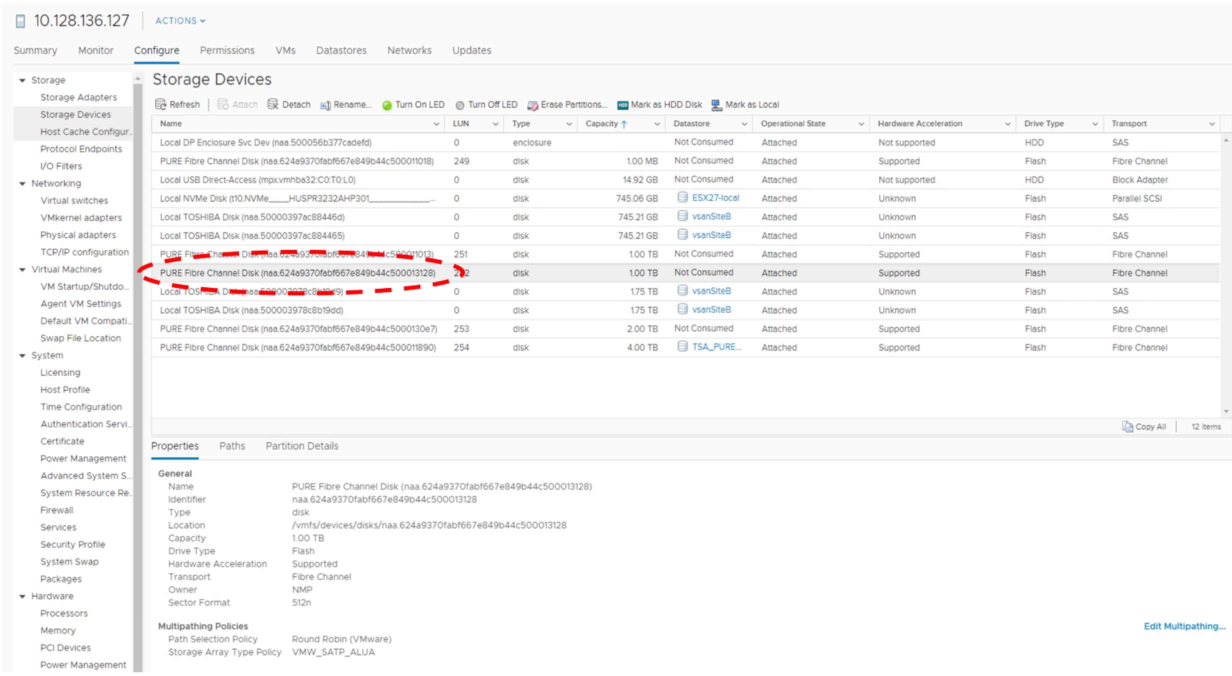
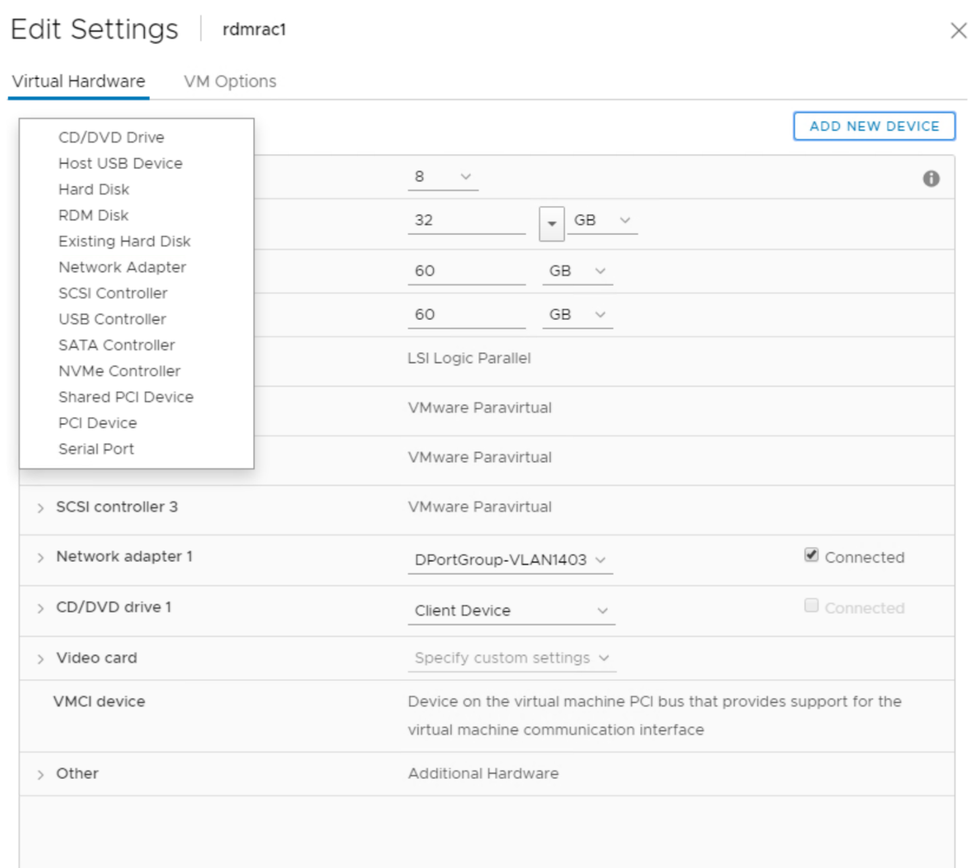


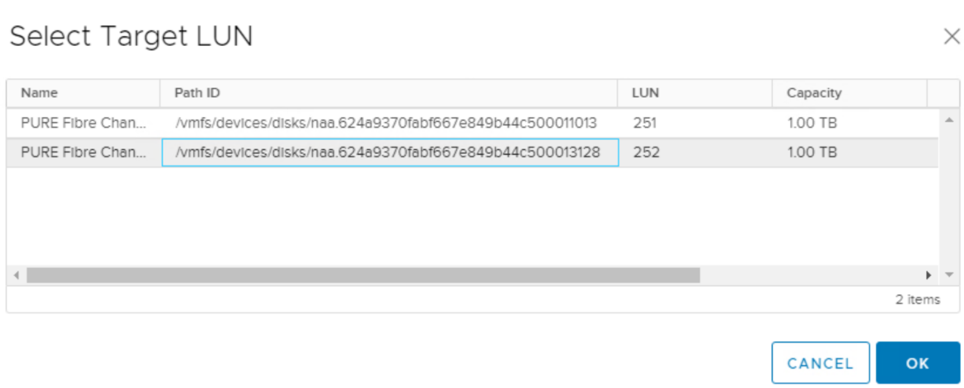
FIGURE 79. RAW DEVICE PURE FIBRE CHANNEL DISK (NAA.624A9370FABF667E849B44C500013128)

The steps required to add shared RDM(s) in physical compatibility mode to Oracle RAC VMs **rdmrac1** and **rdmrac2** are outlined below:

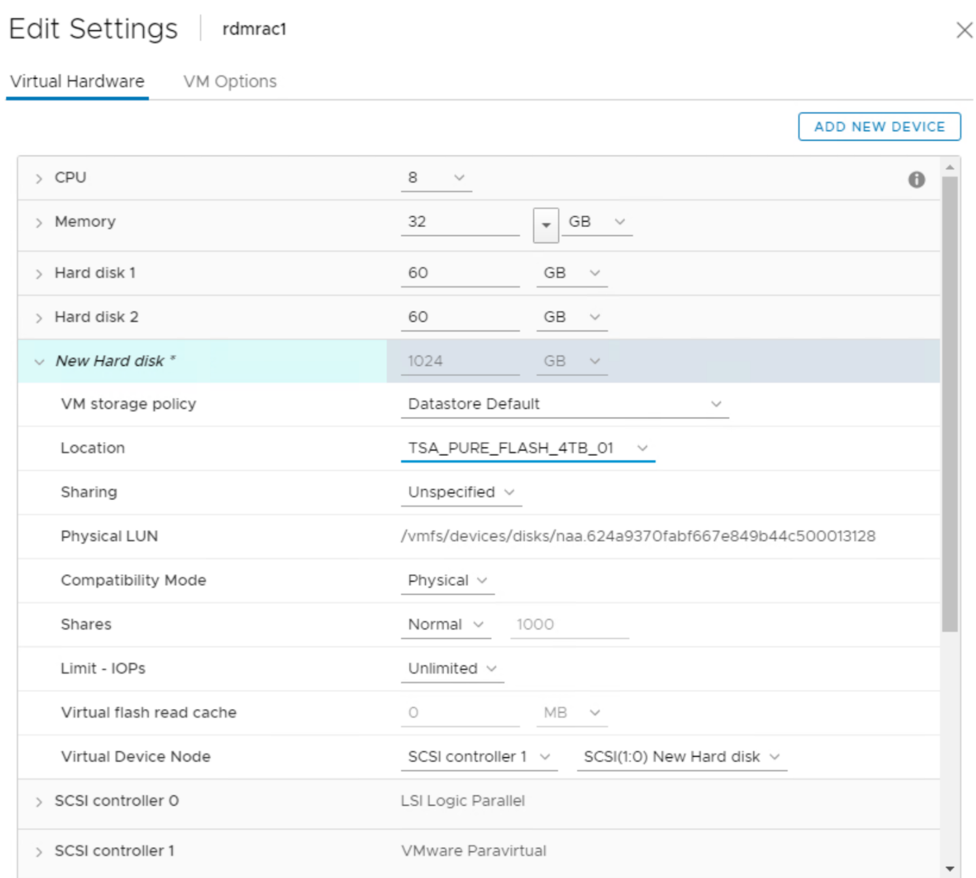
1) Add the RDM disk to the first Oracle RAC VM **rdmrac1** using the add **RDM Disk** option.



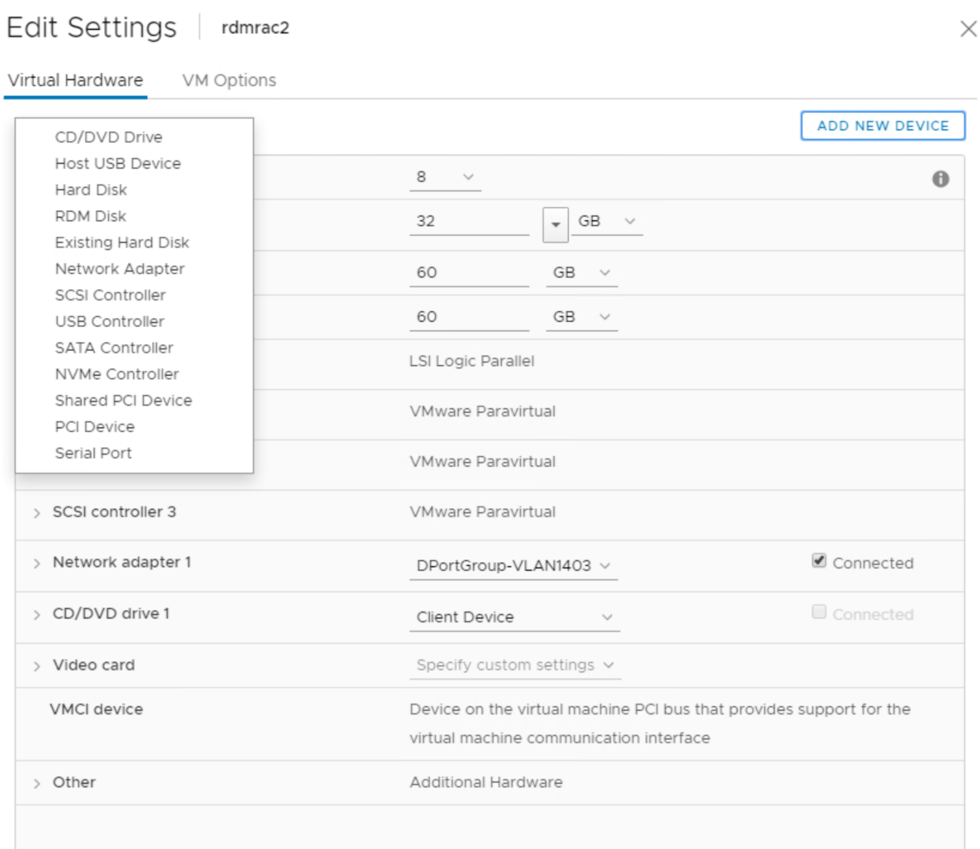
2) Select the correct RDM device **PURE Fibre Channel Disk (naa.624a9370fabf667e849b44c500013128).**

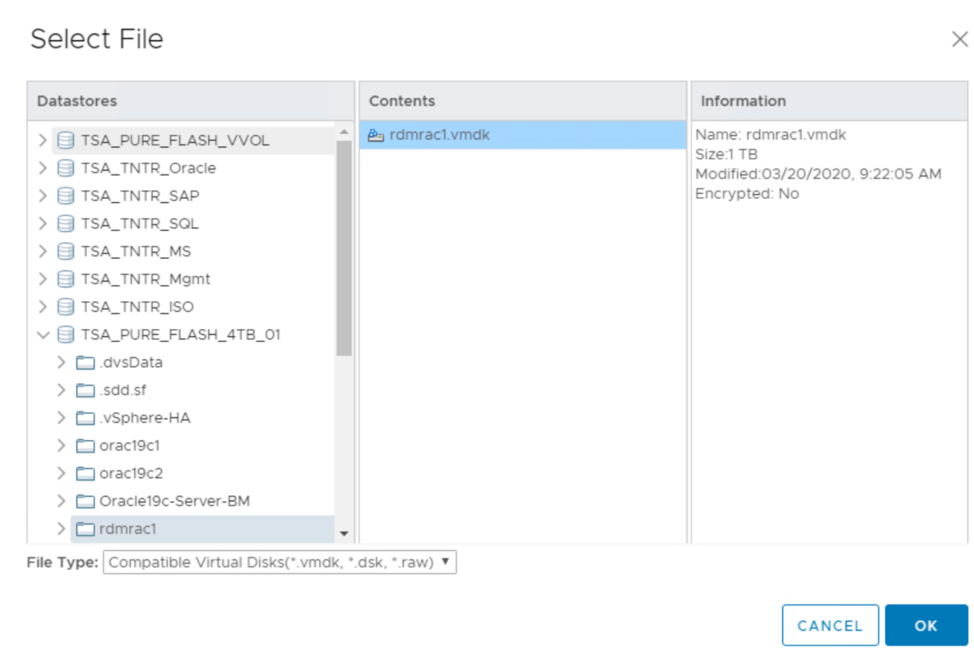


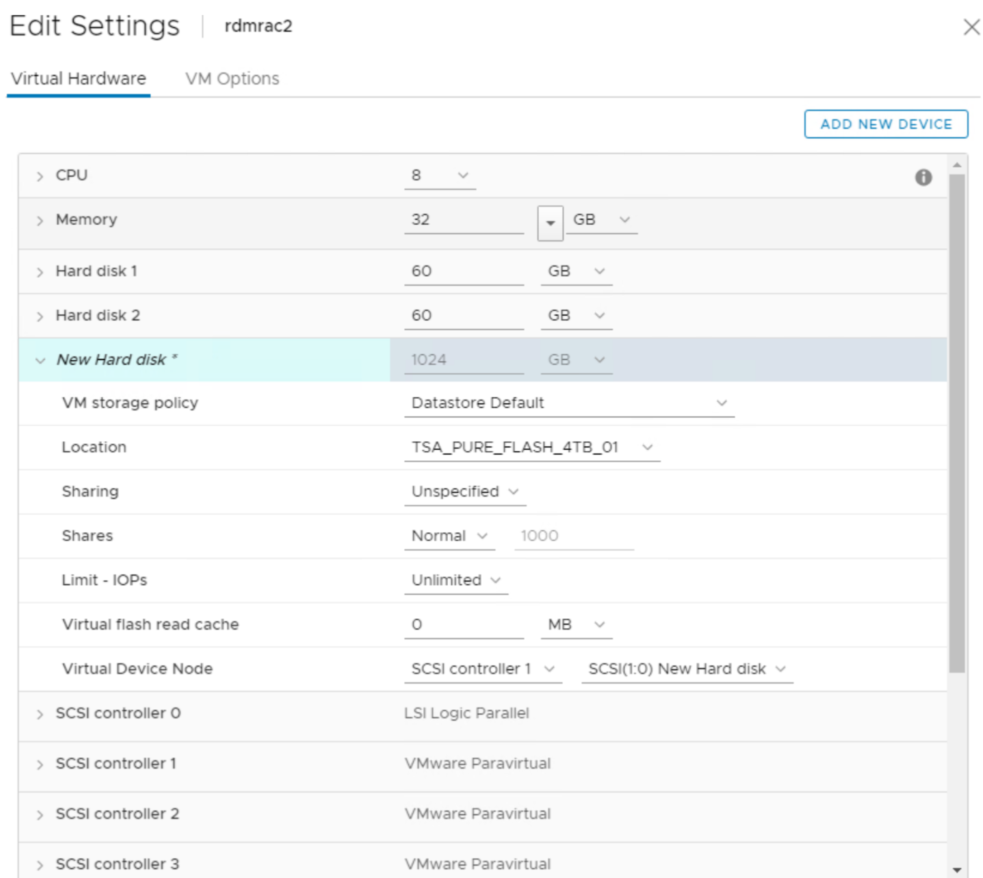
3) Set RDM **Compatibility Mode** to **Physical** as shown. Please make a note of the SCSI ID attached the disk. Use the same ID for this disk when attaching it to the other Oracle RAC VM **rdmrac2** which will be sharing this disk. In this case, SCSI 1:0 is used.



4) Add the same RDM device **PURE Fibre Channel Disk (naa.624a9370fabf667e849b44c500013128)** to Oracle RAC VM **rdmrac2** and to SCSI controller one at the SCSI position 1:0 using the **Existing Hard Disk** option.







5) The shared RDM is now added to both Oracle RAC VMs **rdmrac1** and **rdmrac2**.

In order to use virtual RDMs as shared storage for Oracle RAC, the multi-writer attribute should be used. Follow the same procedure as outlined for shared VMDKs, using the multi-writer attribute as per *KB 1034165* for VMware vSphere (non vSAN Storage).

Using shared physical and virtual RDM(s) for Oracle RAC is clearly explained, using example and screenshots, in *Oracle RAC and VMware Raw Device Mapping (RDM)*.

Learn more about *RDM considerations and limitations*.

Extended Oracle RAC Storage on VMware vSphere Metro Storage Cluster

A VMware vSphere Metro Storage Cluster (vMSC) configuration is a vSphere-certified solution that combines replication with array-based clustering. These solutions are typically deployed in environments where the distance between data centers is limited, often metropolitan or campus environments.

An example of a vMSC using EMC VPLEX is shown as below.

A VMware HA/DRS cluster is created across the two sites using ESXi hosts and managed by vCenter Server. The vSphere Management, vMotion, and VM networks are connected using a redundant network between the two sites. It is assumed that the vCenter Server managing the HA/DRS cluster can connect to the ESXi hosts at both sites.

EMC VPLEX is a federated solution that provides simultaneous access to storage devices at two geographically separate sites. One or more VPLEX Distributed VVols can be provisioned for sharing between the two site's ESXi hosts.

These volumes can be used as RDM disks or as a shared VMFS datastore. The RDM can be used for exclusive access by the VM and the VMFS datastore can be used for provisioning VMs and carving out additional vDisks.

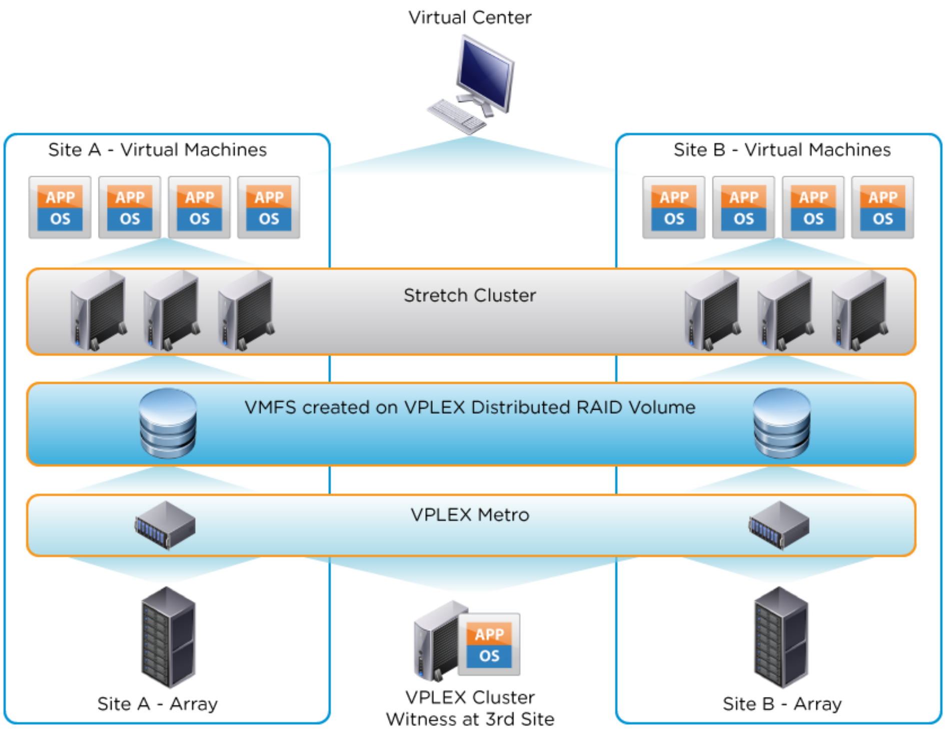


FIGURE 80. VMWARE VSPHERE METRO STORAGE CLUSTER

Learn more about using *vMSC with EMC VPLEX for extended Oracle RAC*.

Further details regarding extended Oracle RAC on vMSC can be found *here*.

This steps required to add a shared Oracle ASM disk with the multi-writer attribute, to an Oracle RAC 19c cluster using a VMFS datastore backed by vMSC, using EMC VPLEX is the same as those outlined in the **Oracle RAC shared storage on VMFS datastore** section of this document.

Extended Oracle RAC Storage on VMware vSAN Stretched Cluster

Stretched clusters extend the vSAN cluster from a single data site to two sites for a higher level of availability and inter-site load-balancing. Stretched clusters are typically deployed in environments where the distance between data centers is limited, such as metropolitan or campus environments.

vSAN Stretched Cluster builds on the foundation of fault domains. The fault domain feature introduced rack awareness in vSAN 6.0. It allows customers to group multiple hosts into failure zones across multiple server racks in order to ensure that replicas of VM objects are not provisioned the same logical failure zones or server racks. Similarly, vSAN Stretched Cluster requires three failure domains and is based on three sites (two active—active sites and witness site). The witness site is only utilized to host a witness virtual appliance that stores witness objects and cluster metadata information and also provide cluster quorum services during failure events.

Learn more about *VMware vSAN Stretched Cluster*.

The following is an example of an extended Oracle RAC on VMware vSAN Stretched Cluster:

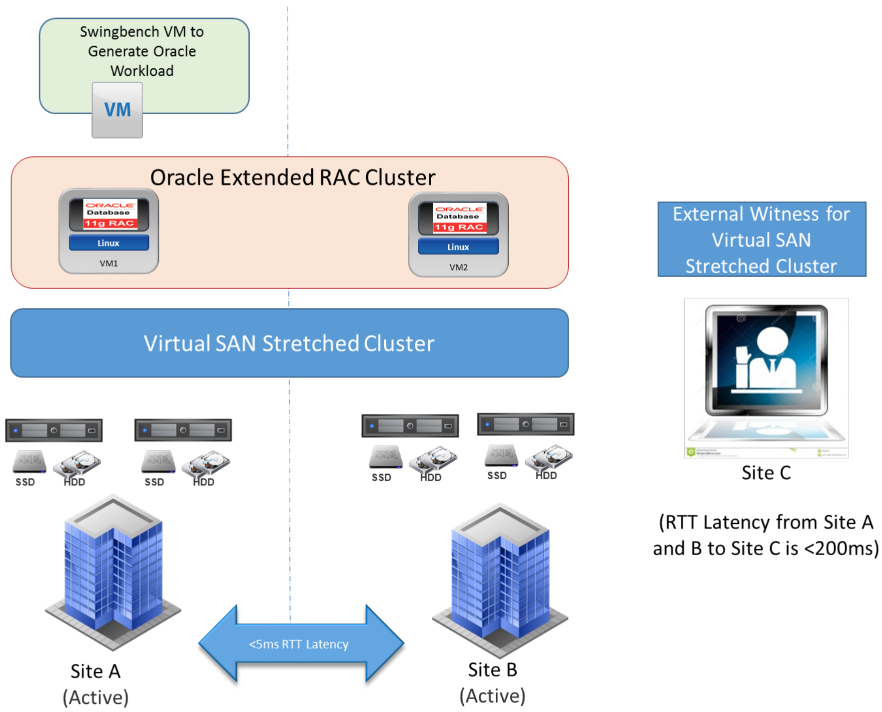


FIGURE 81. EXTENDED ORACLE RAC ON VMWARE VSAN STRETCHED CLUSTER

This steps required to add a shared Oracle ASM disk with the multi-writer attribute to an Oracle RAC 19c cluster, using a stretched vSAN cluster datastore backed by VMware vSAN, are the same as those outlined in the **Oracle RAC shared storage on vSAN datastore** of this document.

Learn more about *Extended Oracle RAC on VMware vSAN Stretched Cluster*.

Oracle RAC Network Deployment Guidelines

The steps required to deploy an Oracle RAC, traditional or extended, on VMware vSphere are essentially the same across all VMware vSphere platforms, with subtle differences in the way Oracle RAC private interconnect networking is set up.

VMware Platform	Distributed Switch	Distributed Port Group	Version	Reference
VMware vSphere, vSAN, vVOls	Regular Distributed Switch	Dedicated Distributed Port Group for Oracle RAC	Version 5.5 and later	*https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.networking.doc/GUID-D21B3241-0AC9-437C-80B1-0C8043CC1D7D.html*

TABLE 27. ORACLE RAC NETWORK DEPLOYMENT GUIDELINES

The sections that follow outline networking deployment for an Oracle RAC cluster:

- Oracle RAC Networking for On-Premises VMware vSphere and vSAN

Oracle RAC Networking for On-Premises VMware vSphere and vSAN

Below is an example of an Oracle RAC public and private network setup with recommendations. These steps are the same for on-premises VMware vSphere and vSAN deployments, including both traditional and stretched cluster deployments.

As previously discussed, referring to the site B architecture shown, distributed-switch VDS **SiteB-10g-dVS** uses (2) 10GbE adapter and (2) 1GbE adapter per host:

3. (2) 10GbE uplinks for VM traffic

4. (2) 1GbE uplinks for VMkernel non-VM traffic

The following distributed-switch port groups were created for Oracle RAC and Oracle VM to balance traffic across the available uplinks:

- Port group **DPortGroup-1403** with VLAN ID 1403 for VM user traffic
- Port group **DPortGroup-VLAN70-NFS** with VLAN ID 70 for NFS traffic
- Port group **DPortGroup-OraclePrivate** with VLAN ID 72 for Oracle RAC interconnect traffic with two active/active uplinks set to **Route based on originating virtual port**
- Port group **DPortGroup-VLAN69-VSAN** with VLAN ID 69 for vSAN traffic

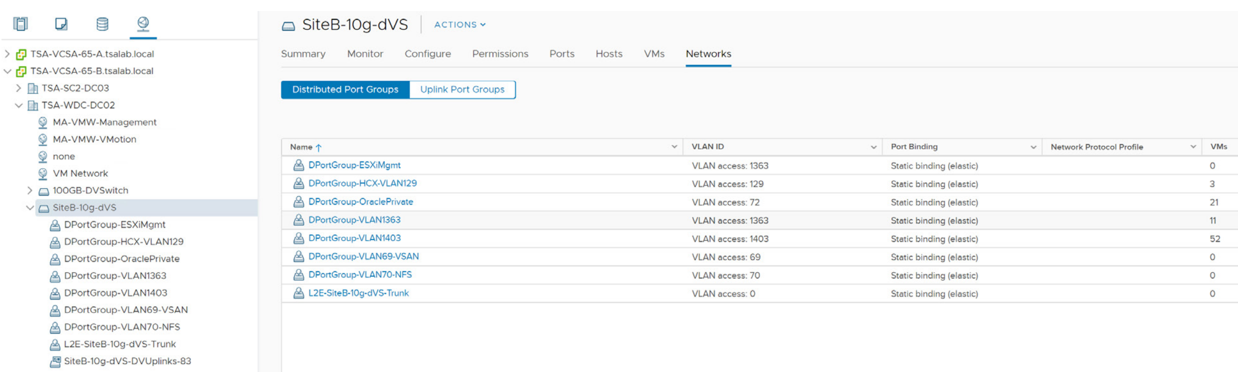


FIGURE 83. SITE B VSPHERE DISTRIBUTED SWITCH PORT GROUP CONFIGURATION

For each Oracle RAC node public adapter, Oracle recommends:

- Each Oracle RAC node must have at least one public network adapter or network interface card (NIC).
- Each public interface must support TCP/IP.

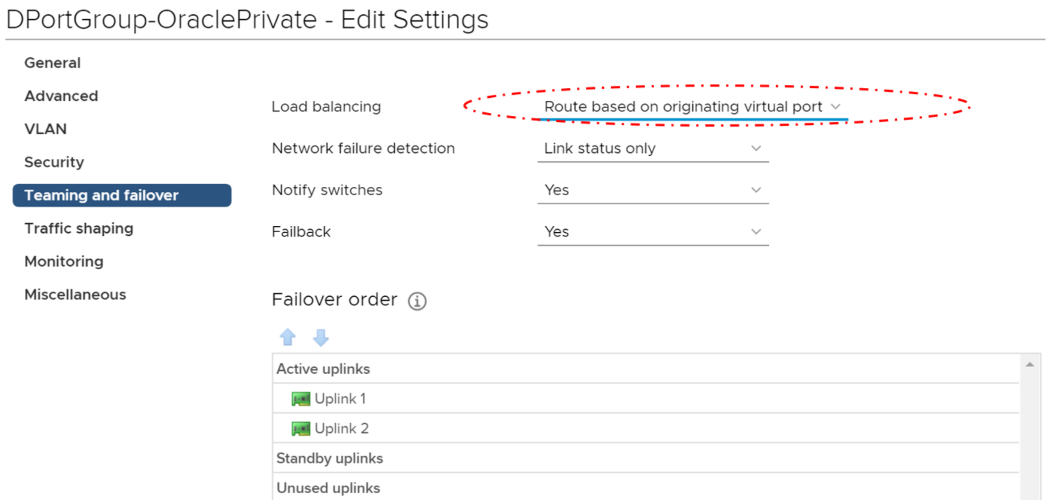
For each Oracle RAC node private adapter, Oracle recommends:

- Each Oracle RAC node must have at least one private network adapter or network interface card (NIC).
- Configuration of interconnects using redundant interconnect usage, in which multiple network adapters are configured with addresses in the link-local range, providing highly available IP (HAIP) addresses for the interconnect.
- Private network must support user datagram protocol (UDP) using high-speed network adapters and switches that support TCP/IP or reliable datagram sockets (RDS) with Infiniband.
- Locate private Interconnect adapter on its own dedicated VLAN with adequate bandwidth for cache fusion traffic.

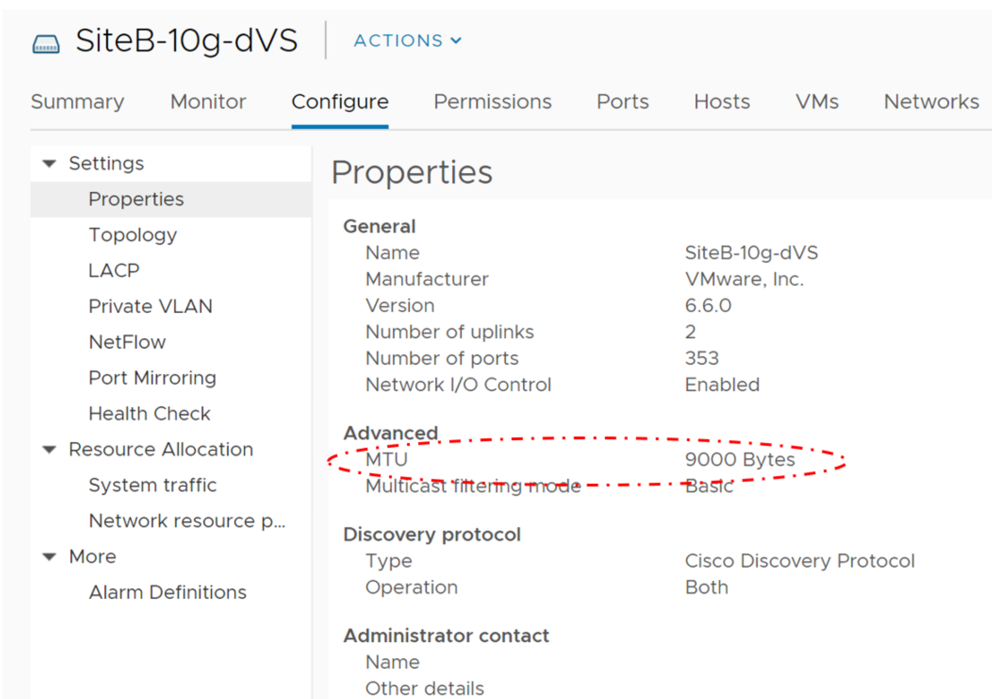
Find the above recommendations in *Network Interface Hardware Minimum Requirements*.

The distributed port group, **DPortGroup-OraclePrivate**, created for Oracle RAC private interconnect has the following attributes set:

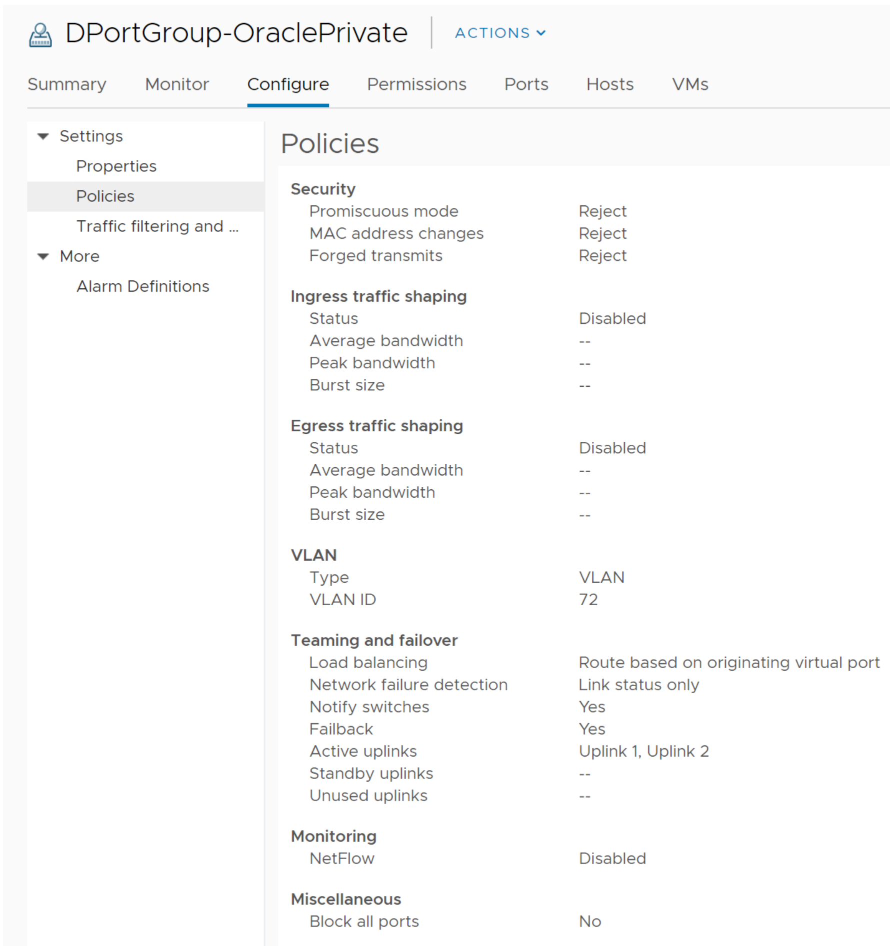
- Set load-balancing on the distributed port group **DPortGroup-OraclePrivate** to **Route based on originating virtual port**.



- Jumbo frames (MTU 9000 Bytes) is set on the distributed switch, Oracle supports and recommends jumbo frames for private interconnect traffic.



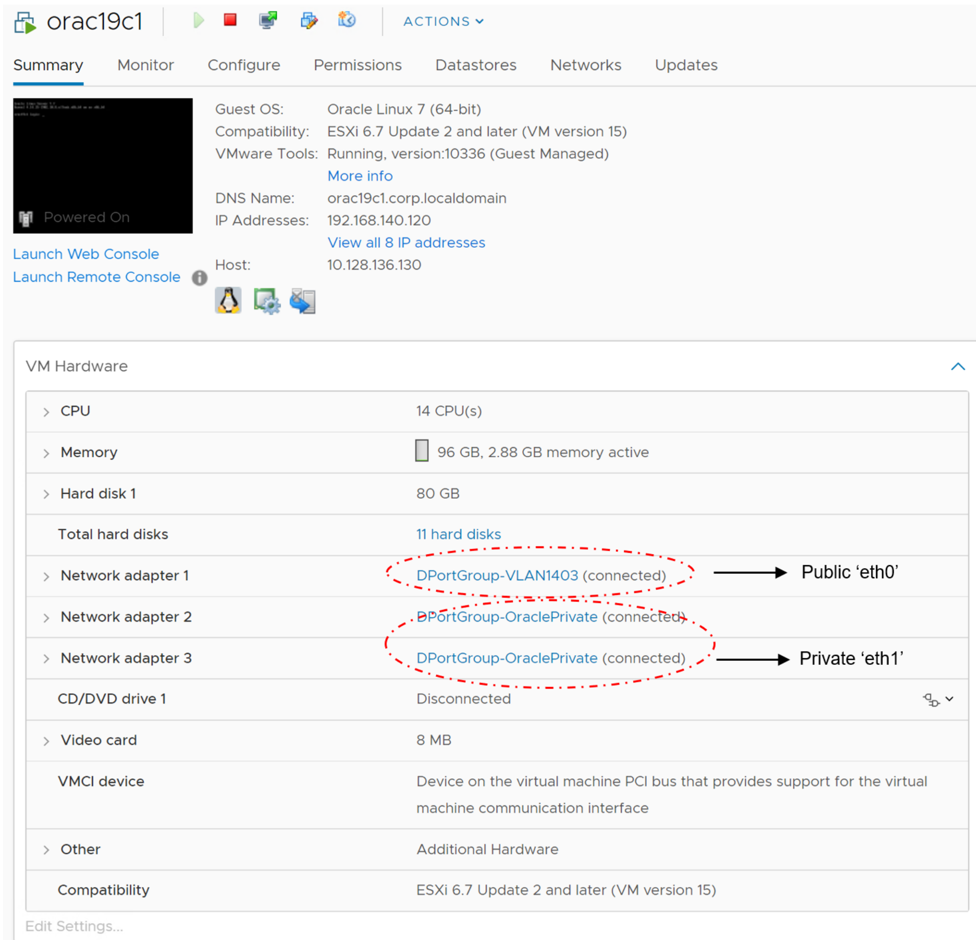
The distributed port group **DPortGroup-OraclePrivate** policies are shown below:

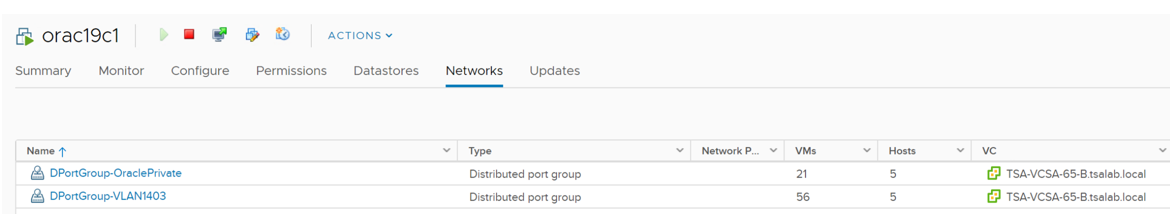


Both Oracle RAC VM **orac19c1** and **orac19c2** have three network adapters each of type **VMXNET3**:

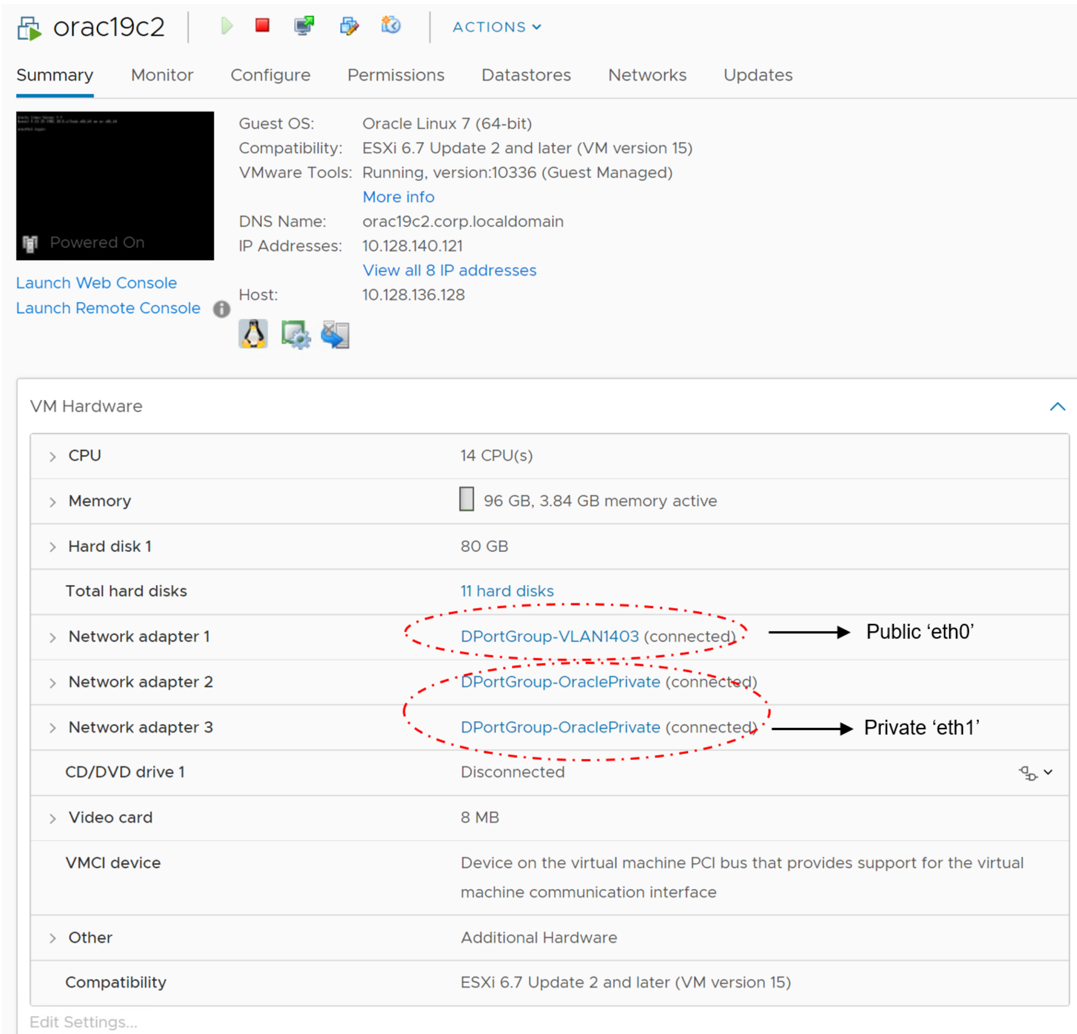
- **One** adapter for public traffic
- **Two** adapters for private interconnect traffic
  - **Two** private network adapters are added to take advantage of Oracle redundant interconnect usage or *Oracle HAIP*

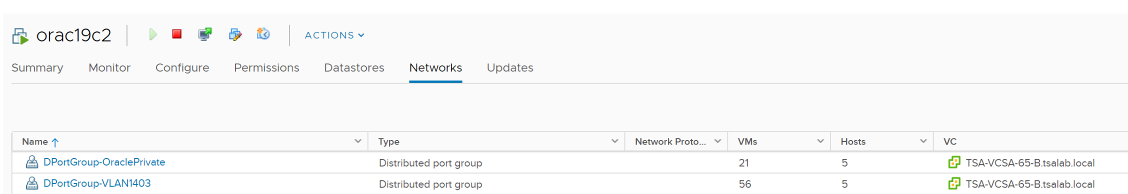
Oracle RAC VM **orac19c1** public and private networking is as shown below:





Oracle RAC VM **orac19c2** public and private networking is as shown below:

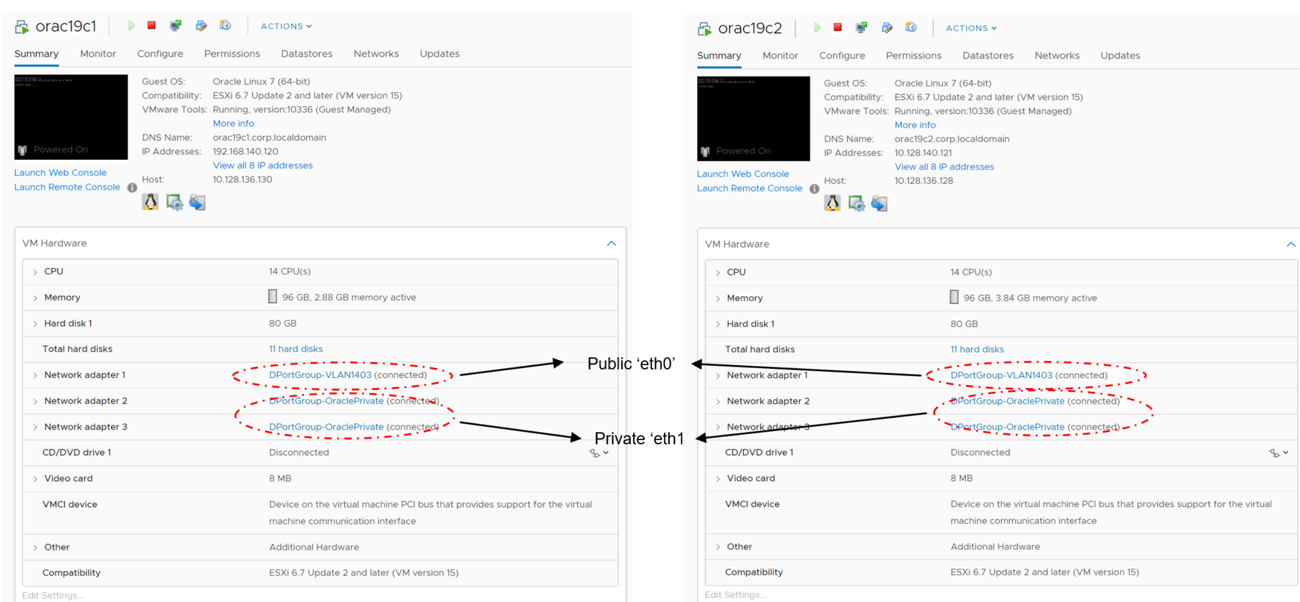




Below shows the hosts file for Oracle RAC VM **orac19c1**

```
[root@orac19c1 ~]# cat /etc/hosts
#127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
#::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
#######################
127.0.0.1 localhost.corp.localdomain localhost
#######################
#RAC Public IPs added to DNS & /etc/hosts
10.128.140.120 orac19c1.corp.localdomain orac19c1
10.128.140.121 orac19c2.corp.localdomain orac19c2
########################
###RAC Private IP's added to /etc/hosts ONLY
##Added to DNS for tracking purpose, Server level firewall to prevent access to priv1 & priv2
##
##RAC Private IPs
192.168.140.120 orac19c1-priv1.corp.localdomain orac19c1-priv1
192.168.140.121 orac19c2-priv1.corp.localdomain orac19c2-priv1
##
##RAC Private IPs
192.168.141.120 orac19c1-priv2.corp.localdomain orac19c1-priv2
192.168.141.121 orac19c2-priv2.corp.localdomain orac19c2-priv2
#
########################
##RAC Virtual IPs added to DNS & /etc/hosts
10.128.140.122 orac19c1-vip.corp.localdomain orac19c1-vip
10.128.140.123 orac19c2-vip.corp.localdomain orac19c2-vip
########################
##RAC SCAN to be added to DNS
##10.128.140.125 orac19c-scan.corp.localdomain orac19c-scan
##10.128.140.126 orac19c-scan.corp.localdomain orac19c-scan
##10.128.140.127 orac19c-scan.corp.localdomain orac19c-scan
#######################
[root@orac19c1 ~]#
```

The network adapters for both Oracle RAC VMs **orac19c1** and **orac19c2** are as shown below:



The following reflects the hosts file for Oracle RAC VM **orac19c2**:

```
[root@orac19c2 ~]# cat /etc/hosts
#127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
#::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
#######################
127.0.0.1 localhost.corp.localdomain localhost
#######################
#RAC Public IPs added to DNS & /etc/hosts
10.128.140.120 orac19c1.corp.localdomain orac19c1
10.128.140.121 orac19c2.corp.localdomain orac19c2
########################
###RAC Private IP's added to /etc/hosts ONLY
##Added to DNS for tracking purpose, Server level firewall to prevent access to priv1 & priv2
##
##RAC Private IPs
192.168.140.120 orac19c1-priv1.corp.localdomain orac19c1-priv1
192.168.140.121 orac19c2-priv1.corp.localdomain orac19c2-priv1
##
##RAC Private IPs
192.168.141.120 orac19c1-priv2.corp.localdomain orac19c1-priv2
192.168.141.121 orac19c2-priv2.corp.localdomain orac19c2-priv2
#
########################
##RAC Virtual IPs added to DNS & /etc/hosts
10.128.140.122 orac19c1-vip.corp.localdomain orac19c1-vip
10.128.140.123 orac19c2-vip.corp.localdomain orac19c2-vip
########################
##RAC SCAN to be added to DNS
##10.128.140.125 orac19c-scan.corp.localdomain orac19c-scan
##10.128.140.126 orac19c-scan.corp.localdomain orac19c-scan
##10.128.140.127 orac19c-scan.corp.localdomain orac19c-scan
#######################
[root@orac19c2 ~]#
```

Oracle RAC Deployment High Level Steps

This section addresses the high-level steps for deploying an Oracle RAC on the VMware vSphere platform.

As part of the Oracle RAC 19c deployment, two VMs **orac19c1** and **orac19c2** were created with Oracle Linux 7.6. These were created with an operating system from a pre-defined template, following all VMware best practices.

Oracle configuration was kept the same on both Oracle RAC VMs:

- 14 vCPU, 96GB RAM with Oracle Enterprise Linux 7.6 operating system
- Oracle 19.3.0.0.0 grid infrastructure and RDBMS binaries installed on both VMs
- Oracle ASMLIB used for device persistence (ASMFD or Linux udev can also be used)
- All ASM disks presented on different PVSCSI controllers for performance and queue depth purposes
- All database related VMDKs partitioned using Linux utilities with proper alignment offset and labelled with Oracle ASMLIB for device persistence
- Oracle ASM disk groups **DATA_DG** and **FRA_DG** created with external redundancy and configured with default allocation unit (AU) size of 4M
- Oracle ASM disk groups **REDO_DG** created with external redundancy and configured with allocation unit (AU) size of 1M
- Oracle ASM disk groups **CRS_DG** and **GIMR_DG** created as part of the grid infrastructure install and configured with default allocation unit (AU) size of 4M
- Oracle ASM disk group **CRS_DG** houses the CRS & VOTE disks and is set to high redundancy
- Oracle ASM disk group **GIMR_DG** houses the grid infrastructure management repository and is set to external redundancy
- Oracle RAC cluster **rac19c** created according to Oracle best practices

The complete list of Oracle initialization parameters can be found in the **Appendix** of this document. All best practices for Oracle on VMware SDDC were followed in accordance with *Oracle Databases on VMware – Best Practices Guide*.

Details of the Oracle RAC VM **orac19c1** are reflected below:

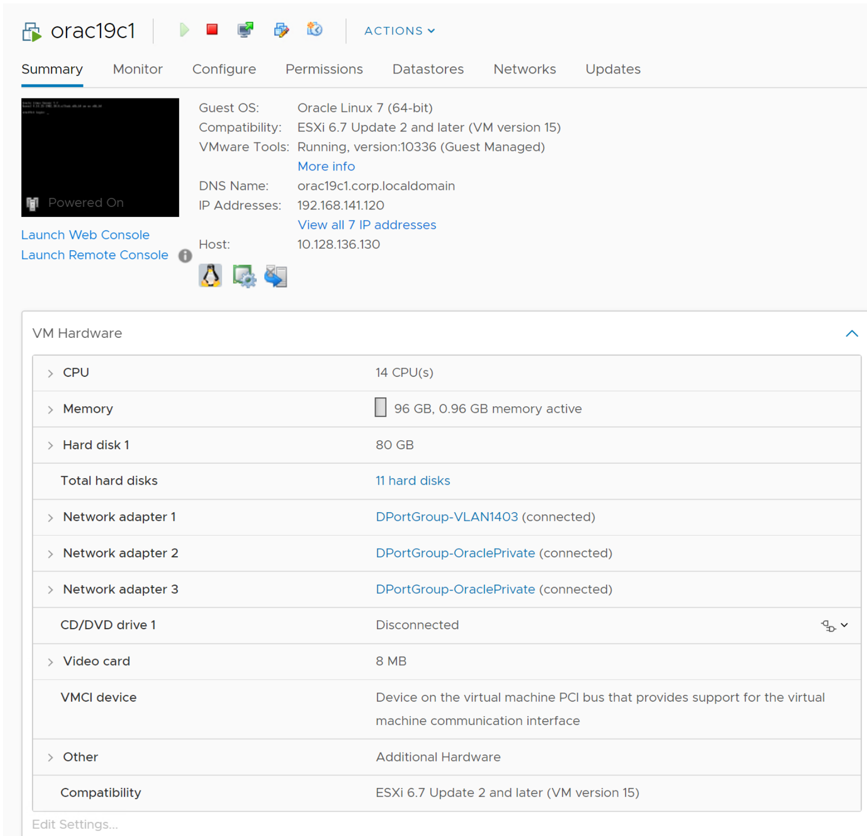


FIGURE 86. ORACLE RAC VM ORAC19C1

Details of the VM **orac19c2** are as follows:

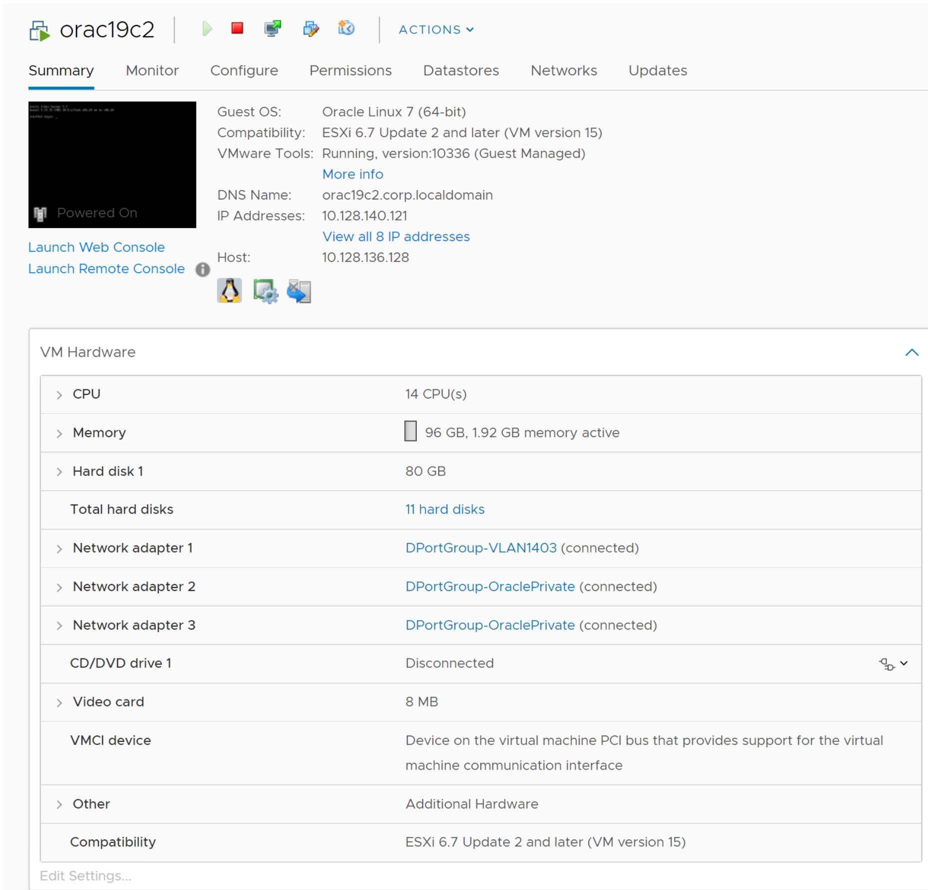


FIGURE 87. ORACLE RAC VM ORAC19C2

For both Oracle RAC VMs **orac19c1** and **orac19c2**, it's recommended to set all four SCSI controllers to **PVSCSI** in accordance with best practices.

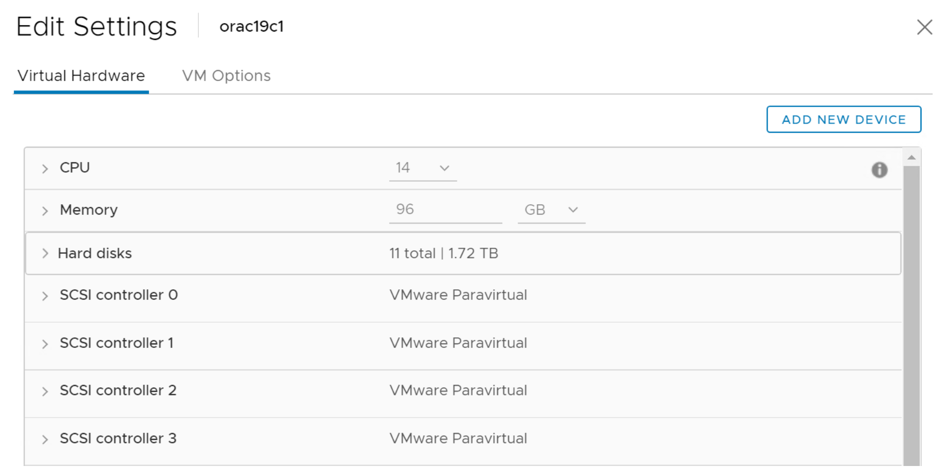


FIGURE 88. VM ORAC19C1 PVSCSI CONTROLLERS

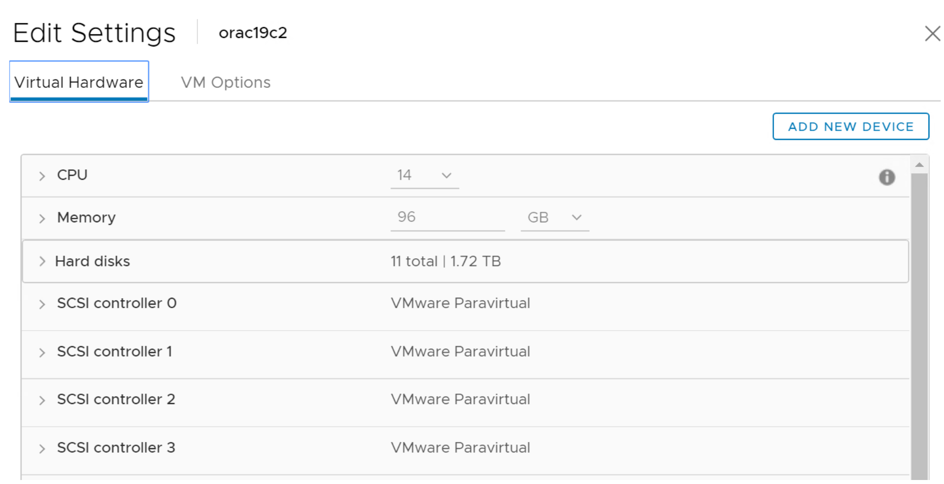
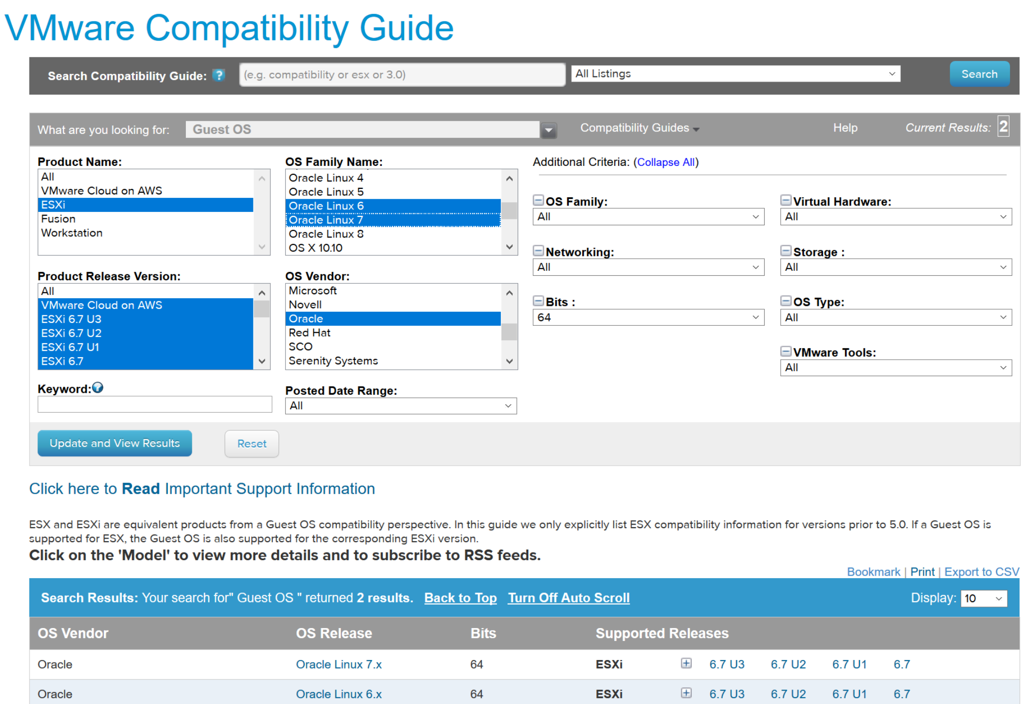


FIGURE 89. VM ORAC19C2 PVSCSI CONTROLLERS

The high-level steps needed to create an Oracle RAC cluster using VMs **orac19c1** and **orac19c2** are outlined below:

1. Deploy two VMs, **orac19c1** and **orac19c2**, from a pre-defined template. The template's guest OS (OEL 7.6 in this deployment) is compatible with VMware vSphere platform version 6.7 (also used in this deployment), as verified by the *VMware compatibility matrix*.



2. Ensure that all Oracle-based OS patches and best practices are adhered to per the *Oracle Databases on VMware – Best Practices Guide*. It's recommended that all patches and best practices be implemented in the template.
3. Ensure Oracle RAC VMs **orac19c1** and **orac19c2** have the latest VMware tools installed. It's recommended that all VMware tools be installed in the template.
4. For both Oracle RAC VMs, **orac19c1** and **orac19c2**, place hard disk one on SCSI0:0 position with 80GB sizing for the OEL 7.6 OS.

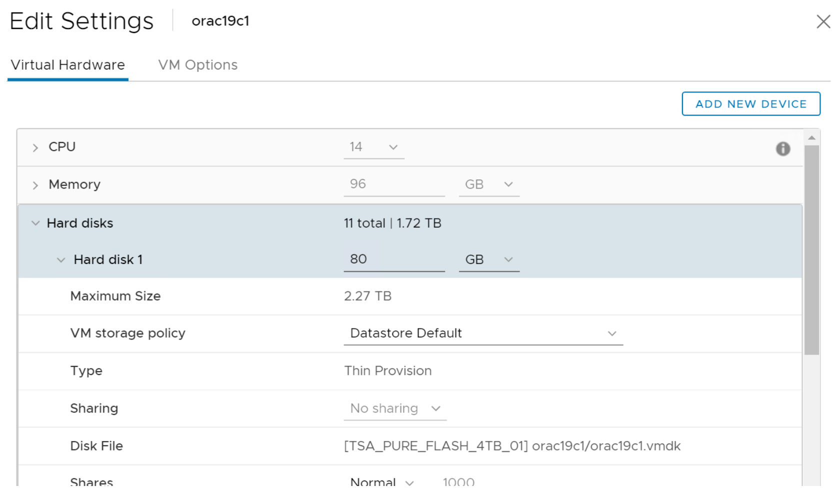


FIGURE 90. ORACLE RAC VM ORAC19C1 OS DISK ON SCSI0:0 Position Set to 80GB

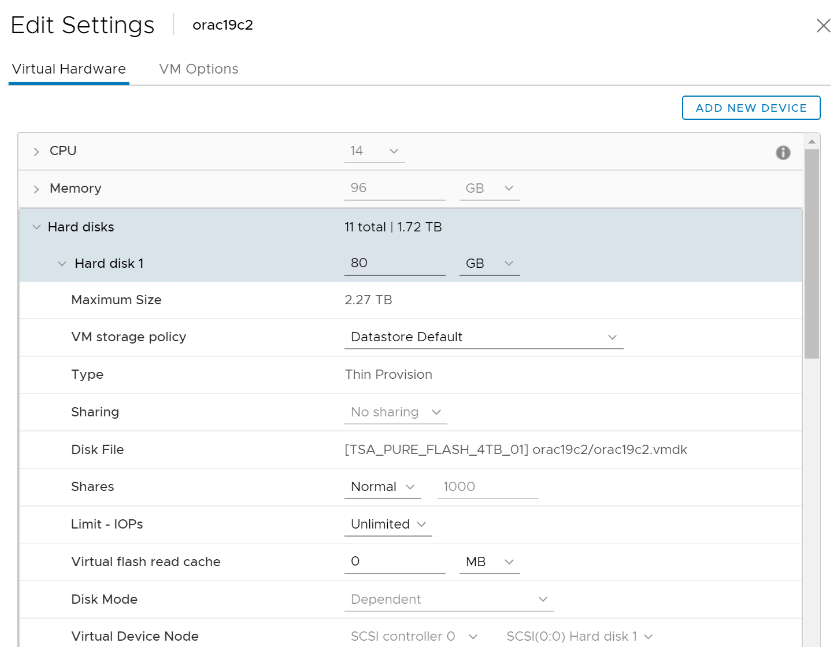


FIGURE 91. ORACLE RAC VM ORAC19C2 OS DISK ON SCSI0:0 Position Set to 80GB

5. For both Oracle RAC VMs **orac19c1** and **orac19c2**, add a second VMDK (Hard Disk 2) on SCSI0:1 position set to 80GB. Create an ext4 (or equivalent file system) (/u01) to hold the Oracle binaries (Grid and RDBMS binaries).

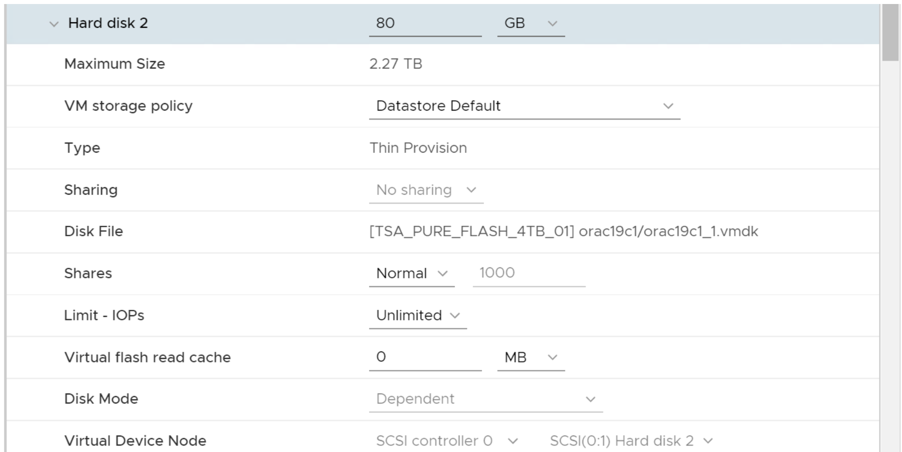


FIGURE 91. ORACLE RAC VM ORAC19C1 ORACLE BINARIES DISK ON SCSI0:1 Position Set to 80GB

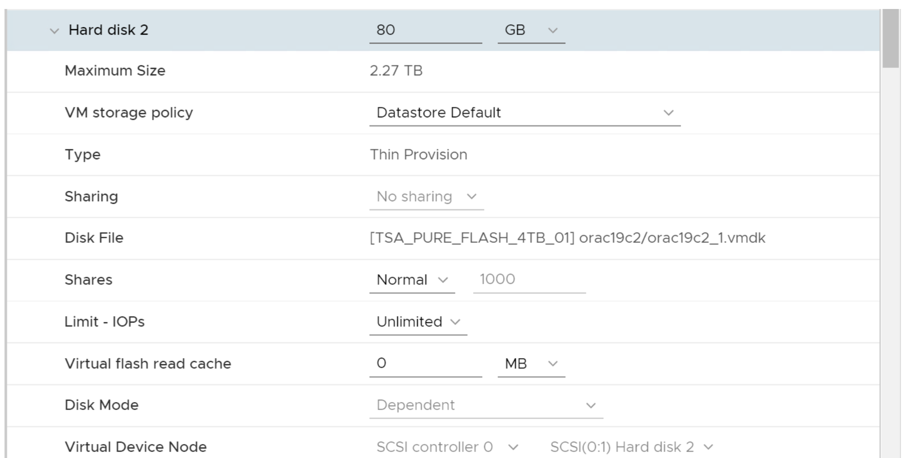


FIGURE 92. ORACLE RAC VM ORAC19C2 ORACLE BINARIES DISK ON SCSI0:1 Position Set to 80GB

Oracle RAC VM **orac19c1** file systems:

```
[root@orac19c1 ~]# df
Filesystem                       1K-blocks     Used Available Use% Mounted on
devtmpfs                          49345468        0  49345468   0% /dev
tmpfs                             49359684  1180156  48179528   3% /dev/shm
tmpfs                             49359684  3048296  46311388   7% /run
tmpfs                             49359684        0  49359684   0% /sys/fs/cgroup
/dev/mapper/ol-root               50000024  3249668  46750356   7% /
/dev/mapper/ol-home               24408428    33232  24375196   1% /home
/dev/mapper/vg2_oracle-LogVol_u01 82434456 23767796  54456180  31% /u01
/dev/sda1                          1038336   251008    787328  25% /boot
tmpfs                              9871940        0   9871940   0% /run/user/501
tmpfs                              9871940        0   9871940   0% /run/user/0
[root@orac19c1 ~]#
```

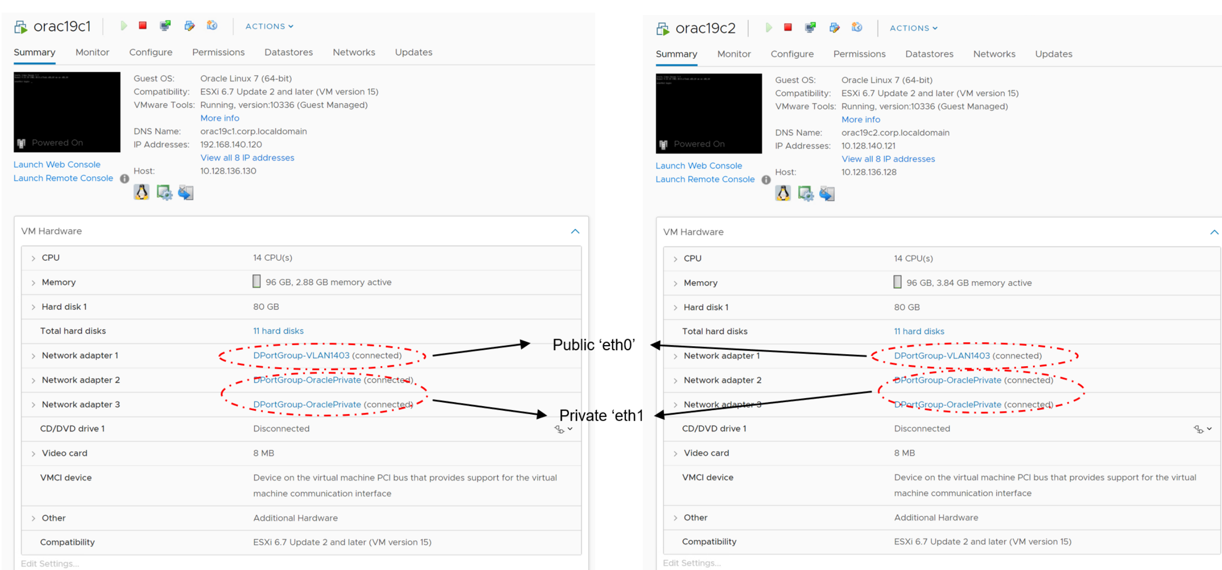
Oracle RAC VM **orac19c2** file systems:

```
[root@orac19c2 ~]# df
Filesystem                       1K-blocks     Used Available Use% Mounted on
devtmpfs                          49345468        0  49345468   0% /dev
tmpfs                             49359684   918012  48441672   2% /dev/shm
tmpfs                             49359684    50016  49309668   1% /run
tmpfs                             49359684        0  49359684   0% /sys/fs/cgroup
/dev/mapper/ol-root               50000024  3151312  46848712   7% /
/dev/mapper/ol-home               24408428    33112  24375316   1% /home
/dev/mapper/vg2_oracle-LogVol_u01 82434456 23198724  55025252  30% /u01
/dev/sda1                          1038336   251008    787328  25% /boot
tmpfs                              9871940        0   9871940   0% /run/user/501
tmpfs                              9871940        0   9871940   0% /run/user/0
[root@orac19c2 ~]#
```

6. Follow *Grid Infrastructure Installation and Upgrade Guide for Linux* for the required Clusterware installation steps.
7. Follow the steps for configuring public, private, VIP and SCAN network for an Oracle cluster, as outlined in *Configuring Networks for Oracle Grid Infrastructure and Oracle RAC*.

For specific guidance regarding Oracle RAC networking, refer to:

– Oracle RAC networking for on-premises VMware vSphere and vSAN



Oracle RAC VM **orac19c1** and **orac19c2** networking IP address details are outlined as follows:

Network	orac19c1	orac19c2
Public FDQN	orac19c1.corp.localdomain	orac19c2.corp.localdomain
Public IP	10.128.140.120	10.128.140.121
Private FDQN	orac19c1-priv1.corp.localdomain	orac19c2-priv1.corp.localdomain
Private IP	192.168.140.120	192.168.140.121
Private FDQN	orac19c1-priv2.corp.localdomain	orac19c2-priv2.corp.localdomain
Private IP	192.168.141.120	192.168.141.121
VIP FDQN	orac19c1-vip.corp.localdomain	orac19c2-vip.corp.localdomain
VIP IP	10.128.140.122	10.128.140.123
SCAN	orac19c-scan.corp.localdomain	
	10.128.140.124	
	10.128.140.125	
	10.128.140.126	

8. Add Oracle RAC shared VMDKs (for ASM disk groups) to Oracle RAC VM **orac19c1** and **orac19c2**.

Refer to specific VMware storage platform guidance for Oracle RAC shared storage:

- Oracle RAC storage on FC VMFS datastore
- Oracle RAC storage on NFS datastore
- Oracle RAC storage on vSAN datastore
- Oracle RAC storage on vVol datastore
- Extended Oracle RAC storage on VMware vSphere Metro Storage Cluster
- Extended Oracle RAC storage on VMware vSAN Stretched Cluster

Oracle RAC VM **orac19c1** disk layout and ASM disk group configuration are outlined in the table below:

Name	SCSI TYPE	SCSI ID (Controller, LUN)	Size (GB)	Type	Source	Hard Disk	Disk Name
Operating System (OS) /	Paravirtual	SCSI (0:0)	80	ext4 Filesystem	Pure AFA	1	/dev/sda1
Oracle binary disk /u01	Paravirtual	SCSI (0:1)	80	ext4 Filesystem	Pure AFA	2	/de/sdb1
DATA disk 1	Paravirtual	SCSI (1:0)	1024	DATA_DG (ASM)	Pure AFA	3	DATA_DISK01
FRA disk 1	Paravirtual	SCSI (2:0)	250	FRA_DG (ASM)	Pure AFA	4	FRA_DISK01
GIMR disk 1	Paravirtual	SCSI (2:1)	250	GIMR_DG (ASM)	Pure AFA	5	GIMR_DISK01
CRS & VOTE disk 1	Paravirtual	SCSI (3:0)	5	CRS_DG (ASM)	Pure AFA	6	CRS_DISK01
CRS & VOTE disk 1	Paravirtual	SCSI (3:1)	5	CRS_DG (ASM)	Pure AFA	7	CRS_DISK02
CRS & VOTE disk 1	Paravirtual	SCSI (3:2)	5	CRS_DG (ASM)	Pure AFA	8	CRS_DISK03
CRS & VOTE disk 1	Paravirtual	SCSI (3:3)	5	CRS_DG (ASM)	Pure AFA	9	CRS_DISK04
CRS & VOTE disk 1	Paravirtual	SCSI (3:4)	5	CRS_DG (ASM)	Pure AFA	10	CRS_DISK05
REDO disk 1	Paravirtual	SCSI (3:5)	50	REDO_DG (ASM)	Pure AFA	11	REDO_DISK01

Oracle RAC VM **orac19c2** disk layout and ASM disk group configuration are as follows:

Name	SCSI TYPE	SCSI ID (Controller, LUN)	Size (GB)	Type	Source	Hard Disk	Disk Name
Operating System (OS) /	Paravirtual	SCSI (0:0)	80	ext4 Filesystem	Pure AFA	1	/dev/sda1
Oracle binary disk /u01	Paravirtual	SCSI (0:1)	80	ext4 Filesystem	Pure AFA	2	/de/sdb1
DATA disk 1	Paravirtual	SCSI (1:0)	1024	DATA_DG (ASM)	Reference to **orac19c1** VMDK	3	DATA_DISK01
					Reference to **orac19c1** VMDK		
FRA disk 1	Paravirtual	SCSI (2:0)	250	FRA_DG (ASM)	Reference to **orac19c1** VMDK	4	FRA_DISK01
GIMR disk 1	Paravirtual	SCSI (2:1)	250	GIMR_DG (ASM)	Reference to **orac19c1** VMDK	5	GIMR_DISK01
					Reference to **orac19c1** VMDK		
CRS & VOTE disk 1	Paravirtual	SCSI (3:0)	5	CRS_DG (ASM)	Reference to **orac19c1** VMDK	6	CRS_DISK01
CRS & VOTE disk 1	Paravirtual	SCSI (3:1)	5	CRS_DG (ASM)	Reference to **orac19c1** VMDK	7	CRS_DISK02
CRS & VOTE disk 1	Paravirtual	SCSI (3:2)	5	CRS_DG (ASM)	Reference to **orac19c1** VMDK	8	CRS_DISK03
CRS & VOTE disk 1	Paravirtual	SCSI (3:3)	5	CRS_DG (ASM)	Reference to **orac19c1** VMDK	9	CRS_DISK04
CRS & VOTE disk 1	Paravirtual	SCSI (3:4)	5	CRS_DG (ASM)	Reference to **orac19c1** VMDK	10	CRS_DISK05
REDO disk 1	Paravirtual	SCSI (3:5)	50	REDO_DG (ASM)	Reference to **orac19c1** VMDK	11	REDO_DISK01

Virtual disks for both Oracle RAC VMs are as shown below:

orac19c1

NAME	SCSI ID (CONTROLLER, LUN)	SIZE (GB)	TYPE	DISK NAME
Operating System (OS) /	SCSI (0:0)	80	ext4 Filesystem	/dev/sda1
Oracle binary disk /u01	SCSI (0:1)	80	ext4 Filesystem	/dev/sdb1
DATA Disk 1	SCSI (1:0)	1024	DATA_DG	DATA_DISK01
FRA Disk 1	SCSI (2:0)	250	FRA_DG	FRA_DISK01
GIMR Disk 1	SCSI (2:1)	250	GIMR_DG	GIMR_DISK01
VOTE Disk 1	SCSI (3:0)	5	CRS_DG	CRS_DISK01
VOTE Disk 2	SCSI (3:1)	5	CRS_DG	CRS_DISK02
VOTE Disk 3	SCSI (3:2)	5	CRS_DG	CRS_DISK03
VOTE Disk 4	SCSI (3:3)	5	CRS_DG	CRS_DISK04
VOTE Disk 5	SCSI (3:4)	5	CRS_DG	CRS_DISK05
REDO Disk 1	SCSI (3:5)	50	REDO_DG	REDO_DISK01

orac19c2

NAME	SCSI ID (CONTROLLER, LUN)	SIZE (GB)	TYPE	DISK NAME
Operating System (OS) /	SCSI (0:0)	50	ext4 Filesystem	/dev/sda1
Oracle binary disk /u01	SCSI (0:1)	50	ext4 Filesystem	/dev/sdb1

VM 'orac19c2' shared vmdk's are a reference to VM 'orac19c1' shared EZT vmdk's

9. this deployment, Oracle ASMLIB is used for device persistence on the OS. Oracle ASMFD or Linux udev may also be used.
10. The disk must be partitioned for Oracle ASMLIB and ASMFD. The disk partitioning offset for all ASM disks are set to 1MB. In the example below, the **fdisk** command output for the DATA_DISK01 is shown. Other disk partitioning tools (e.g., parted) can be used as well. Consult the storage vendor for the appropriate partitioning offset.

```
[root@orac19c1 ~]# fdisk -lu /dev/sdi

Disk /dev/sdi: 1099.5 GB, 1099511627776 bytes, 2147483648 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x17e8fcf8

   Device Boot      Start         End      Blocks   Id  System
/dev/sdi1            2048  2147483647  1073740800   83  Linux
[root@orac19c1 ~]#
```

11. Use the **oracleasm createdisk** command on Oracle RAC VM **orac19c1** to create Oracle ASM disks using Oracle ASMLIB. Use the **oracleasm scandisks** command on Oracle RAC VM **orac19c2** to instantiate the newly created ASM disks.

Below is a listing of Linux disk devices, reflecting the ASM disks on Oracle RAC VM **orac19c1**:

```
[root@orac19c1 ~]# blkid
/dev/sdb1: UUID="XN2HMh-vXiF-xR8w-MEG9-EbBS-uJN4-rdQOX0" TYPE="LVM2_member"
/dev/sdc1: LABEL="CRS_DISK01" TYPE="oracleasm"
/dev/sde1: LABEL="CRS_DISK03" TYPE="oracleasm"
/dev/sdf1: LABEL="CRS_DISK04" TYPE="oracleasm"
/dev/sdi1: LABEL="DATA_DISK01" TYPE="oracleasm"
/dev/sdh1: LABEL="REDO_DISK01" TYPE="oracleasm"
/dev/sdg1: LABEL="CRS_DISK05" TYPE="oracleasm"
/dev/sdj1: LABEL="FRA_DISK01" TYPE="oracleasm"
/dev/sdk1: LABEL="GIMR_DISK01" TYPE="oracleasm"
/dev/sda1: UUID="f6e6f3d4-503f-41b7-9bb2-a07414a21e54" TYPE="xfs"
/dev/sda2: UUID="zavApe-VuLG-9Yef-Ch5A-L0wv-INHH-8ZThrN" TYPE="LVM2_member"
/dev/sdd1: LABEL="CRS_DISK02" TYPE="oracleasm"
/dev/mapper/ol-root: UUID="f34e54b4-9385-40b8-8e45-648541093393" TYPE="xfs"
/dev/mapper/ol-swap: UUID="8cca1839-9676-4e07-8070-5719141f1512" TYPE="swap"
/dev/mapper/ol-home: UUID="ec8f6b35-b1b7-4fb1-b4b8-c7441be60ffb" TYPE="xfs"
/dev/mapper/vg2_oracle-LogVol_u01: UUID="2323b468-3cc6-4d75-a91f-fa1921b5ffd4" TYPE="ext4"
[root@orac19c1 ~]#
```

Below is a listing of Linux disk devices reflecting the AMS disks on Oracle RAC VM **orac19c2**:

```
[root@orac19c2 ~]# blkid
/dev/sda1: UUID="f6e6f3d4-503f-41b7-9bb2-a07414a21e54" TYPE="xfs"
/dev/sda2: UUID="zavApe-VuLG-9Yef-Ch5A-L0wv-INHH-8ZThrN" TYPE="LVM2_member"
/dev/sdc1: LABEL="CRS_DISK01" TYPE="oracleasm"
/dev/sde1: LABEL="CRS_DISK03" TYPE="oracleasm"
/dev/sdh1: LABEL="REDO_DISK01" TYPE="oracleasm"
/dev/sdf1: LABEL="CRS_DISK04" TYPE="oracleasm"
/dev/sdg1: LABEL="CRS_DISK05" TYPE="oracleasm"
/dev/sdb1: UUID="XN2HMh-vXiF-xR8w-MEG9-EbBS-uJN4-rdQOX0" TYPE="LVM2_member"
/dev/sdj1: LABEL="FRA_DISK01" TYPE="oracleasm"
/dev/sdk1: LABEL="GIMR_DISK01" TYPE="oracleasm"
/dev/sdd1: LABEL="CRS_DISK02" TYPE="oracleasm"
/dev/sdi1: LABEL="DATA_DISK01" TYPE="oracleasm"
/dev/mapper/ol-root: UUID="f34e54b4-9385-40b8-8e45-648541093393" TYPE="xfs"
/dev/mapper/ol-swap: UUID="8cca1839-9676-4e07-8070-5719141f1512" TYPE="swap"
/dev/mapper/vg2_oracle-LogVol_u01: UUID="2323b468-3cc6-4d75-a91f-fa1921b5ffd4" TYPE="ext4"
/dev/mapper/ol-home: UUID="ec8f6b35-b1b7-4fb1-b4b8-c7441be60ffb" TYPE="xfs"
[root@orac19c2 ~]#
```

Details for ASM disks on Oracle RAC VM **orac19c1** are outlined below:

```
[root@orac19c1 ~]# oracleasm listdisks
CRS_DISK01
CRS_DISK02
CRS_DISK03
CRS_DISK04
CRS_DISK05
DATA_DISK01
FRA_DISK01
GIMR_DISK01
REDO_DISK01
[root@orac19c1 ~]#
```

Details for ASM disks on Oracle RAC VM **orac19c2** are outlined below:

```
[root@orac19c2 ~]# oracleasm listdisks
CRS_DISK01
CRS_DISK02
CRS_DISK03
CRS_DISK04
CRS_DISK05
DATA_DISK01
FRA_DISK01
GIMR_DISK01
REDO_DISK01
[root@orac19c2 ~]#
```

The following two graphics represent details for ASM disk groups.

ASM disk groups on Oracle RAC VM **orac19c1**:

```
grid@orac19c1:+ASM1:/home/grid> asmcmd
ASMCMD> lsdg
State    Type    Rebal  Sector  Logical_Sector  Block       AU  Total_MB  Free_MB  Req_mir_free_MB  Usable_file_MB  Offline_disks  Voting_files  Name
MOUNTED  HIGH    N         512             512   4096  4194304     25580    24388            10232            4718              0             Y  CRS_DG/
MOUNTED  EXTERN  N         512             512   4096  4194304   1048572  1041080                0         1041080              0             N  DATA_DG/
MOUNTED  EXTERN  N         512             512   4096  4194304    255996   255868                0          255868              0             N  FRA_DG/
MOUNTED  EXTERN  N         512             512   4096  4194304    255996   232292                0          232292              0             N  GIMR_DG/
MOUNTED  EXTERN  N         512             512   4096  1048576     51199    46856                0           46856              0             N  REDO_DG/
ASMCMD>
```

ASM disk groups on Oracle RAC VM **orac19c2**:

```
grid@orac19c2:+ASM2:/home/grid> asmcmd
ASMCMD> lsdg
State    Type    Rebal  Sector  Logical_Sector  Block       AU  Total_MB  Free_MB  Req_mir_free_MB  Usable_file_MB  Offline_disks  Voting_files  Name
MOUNTED  HIGH    N         512             512   4096  4194304     25580    24388            10232            4718              0             Y  CRS_DG/
MOUNTED  EXTERN  N         512             512   4096  4194304   1048572  1041080                0         1041080              0             N  DATA_DG/
MOUNTED  EXTERN  N         512             512   4096  4194304    255996   255868                0          255868              0             N  FRA_DG/
MOUNTED  EXTERN  N         512             512   4096  4194304    255996   232292                0          232292              0             N  GIMR_DG/
MOUNTED  EXTERN  N         512             512   4096  1048576     51199    46856                0           46856              0             N  REDO_DG/
ASMCMD>
```

12. Follow the remaining steps to install an Oracle RAC cluster as outlined in *Grid Infrastructure Installation and Upgrade Guide for Linux*.
13. Follow the steps outline in *Real Application Clusters Installation Guide for Linux and UNIX* to install the Oracle RAC database binaries and create the Oracle RAC database.

Once the Oracle RAC cluster is successfully created, Oracle RAC networking on Oracle RAC VM **orac19c1** will be shown as reflected below:

```
[root@orac19c1 ~]# ifconfig -a
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.128.140.120  netmask 255.255.255.0  broadcast 10.128.140.255
        ether 00:50:56:8e:e5:c6  txqueuelen 1000  (Ethernet)
        RX packets 116859  bytes 27785494 (26.4 MiB)
        RX errors 0  dropped 8  overruns 0  frame 0
        TX packets 126050  bytes 27651449 (26.3 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth0:1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.128.140.122  netmask 255.255.255.0  broadcast 10.128.140.255
        ether 00:50:56:8e:e5:c6  txqueuelen 1000  (Ethernet)

eth0:2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.128.140.125  netmask 255.255.255.0  broadcast 10.128.140.255
        ether 00:50:56:8e:e5:c6  txqueuelen 1000  (Ethernet)

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.140.120  netmask 255.255.255.0  broadcast 192.168.140.255
        ether 00:50:56:8e:b7:7d  txqueuelen 1000  (Ethernet)
        RX packets 421586  bytes 358210154 (341.6 MiB)
        RX errors 0  dropped 25  overruns 0  frame 0
        TX packets 269888  bytes 170308925 (162.4 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth1:1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 169.254.6.8  netmask 255.255.240.0  broadcast 169.254.15.255
        ether 00:50:56:8e:b7:7d  txqueuelen 1000  (Ethernet)

eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.141.120  netmask 255.255.255.0  broadcast 192.168.141.255
        ether 00:50:56:8e:19:6e  txqueuelen 1000  (Ethernet)
        RX packets 166060  bytes 133501066 (127.3 MiB)
        RX errors 0  dropped 19  overruns 0  frame 0
        TX packets 288084  bytes 298404544 (284.5 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth2:1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 169.254.31.1  netmask 255.255.240.0  broadcast 169.254.31.255
        ether 00:50:56:8e:19:6e  txqueuelen 1000  (Ethernet)

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 315789  bytes 107219062 (102.2 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 315789  bytes 107219062 (102.2 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@orac19c1 ~]#
```

Oracle RAC networking on Oracle RAC VM **orac19c2**:

```
[root@orac19c2 ~]# ifconfig -a
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.128.140.121  netmask 255.255.255.0  broadcast 10.128.140.255
        ether 00:50:56:8e:df:0d  txqueuelen 1000  (Ethernet)
        RX packets 4489391  bytes 986938872 (941.2 MiB)
        RX errors 0  dropped 21  overruns 0  frame 0
        TX packets 4003055  bytes 979746584 (934.3 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth0:1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.128.140.123  netmask 255.255.255.0  broadcast 10.128.140.255
        ether 00:50:56:8e:df:0d  txqueuelen 1000  (Ethernet)

eth0:3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.128.140.126  netmask 255.255.255.0  broadcast 10.128.140.255
        ether 00:50:56:8e:df:0d  txqueuelen 1000  (Ethernet)

eth0:4: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.128.140.127  netmask 255.255.255.0  broadcast 10.128.140.255
        ether 00:50:56:8e:df:0d  txqueuelen 1000  (Ethernet)

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.140.121  netmask 255.255.255.0  broadcast 192.168.140.255
        ether 00:50:56:8e:9a:95  txqueuelen 1000  (Ethernet)
        RX packets 3579889  bytes 2118264963 (1.9 GiB)
        RX errors 0  dropped 81  overruns 0  frame 0
        TX packets 6147196  bytes 5851363555 (5.4 GiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth1:1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 169.254.15.39  netmask 255.255.240.0  broadcast 169.254.15.255
        ether 00:50:56:8e:9a:95  txqueuelen 1000  (Ethernet)

eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.141.121  netmask 255.255.255.0  broadcast 192.168.141.255
        ether 00:50:56:8e:f8:8c  txqueuelen 1000  (Ethernet)
        RX packets 4669074  bytes 4568147804 (4.2 GiB)
        RX errors 0  dropped 62  overruns 0  frame 0
        TX packets 2663633  bytes 2106703994 (1.9 GiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth2:1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 169.254.29.28  netmask 255.255.240.0  broadcast 169.254.31.255
        ether 00:50:56:8e:f8:8c  txqueuelen 1000  (Ethernet)

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 1267845  bytes 345097644 (329.1 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1267845  bytes 345097644 (329.1 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@orac19c2 ~]#
```

Oracle RAC resources and status:

```
[root@orac19c1 ~]# /u01/app/19.0.0/grid/bin/crsctl stat res -t
--------------------------------------------------------------------------------
Name           Target  State        Server                   State details
--------------------------------------------------------------------------------
Local Resources
--------------------------------------------------------------------------------
ora.LISTENER.lsnr
               ONLINE  ONLINE       orac19c1                 STABLE
               ONLINE  ONLINE       orac19c2                 STABLE
ora.chad
               ONLINE  ONLINE       orac19c1                 STABLE
               ONLINE  ONLINE       orac19c2                 STABLE
ora.net1.network
               ONLINE  ONLINE       orac19c1                 STABLE
               ONLINE  ONLINE       orac19c2                 STABLE
ora.ons
               ONLINE  ONLINE       orac19c1                 STABLE
               ONLINE  ONLINE       orac19c2                 STABLE
--------------------------------------------------------------------------------
Cluster Resources
--------------------------------------------------------------------------------
ora.ASMNET1LSNR_ASM.lsnr(ora.asmgroup)
      1        ONLINE  ONLINE       orac19c1                 STABLE
      2        ONLINE  ONLINE       orac19c2                 STABLE
      3        ONLINE  OFFLINE                               STABLE
ora.ASMNET2LSNR_ASM.lsnr(ora.asmgroup)
      1        ONLINE  ONLINE       orac19c1                 STABLE
      2        ONLINE  ONLINE       orac19c2                 STABLE
      3        ONLINE  OFFLINE                               STABLE
ora.CRS_DG.dg(ora.asmgroup)
      1        ONLINE  ONLINE       orac19c1                 STABLE
      2        ONLINE  ONLINE       orac19c2                 STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.DATA_DG.dg(ora.asmgroup)
      1        ONLINE  ONLINE       orac19c1                 STABLE
      2        ONLINE  ONLINE       orac19c2                 STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.FRA_DG.dg(ora.asmgroup)
      1        ONLINE  ONLINE       orac19c1                 STABLE
      2        ONLINE  ONLINE       orac19c2                 STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.GIMR_DG.dg(ora.asmgroup)
      1        ONLINE  ONLINE       orac19c1                 STABLE
      2        ONLINE  ONLINE       orac19c2                 STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.LISTENER_SCAN1.lsnr
      1        ONLINE  ONLINE       orac19c1                 STABLE
ora.LISTENER_SCAN2.lsnr
      1        ONLINE  ONLINE       orac19c2                 STABLE
ora.LISTENER_SCAN3.lsnr
      1        ONLINE  ONLINE       orac19c2                 STABLE
ora.MGMTLSNR
      1        ONLINE  ONLINE       orac19c2                 169.254.15.39 192.16
                                                             8.140.121 192.168.14
                                                             1.121,STABLE
ora.REDO_DG.dg(ora.asmgroup)
      1        ONLINE  ONLINE       orac19c1                 STABLE
      2        ONLINE  ONLINE       orac19c2                 STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.asm(ora.asmgroup)
      1        ONLINE  ONLINE       orac19c1                 Started,STABLE
      2        ONLINE  ONLINE       orac19c2                 Started,STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.asmnet1.asmnetwork(ora.asmgroup)
      1        ONLINE  ONLINE       orac19c1                 STABLE
      2        ONLINE  ONLINE       orac19c2                 STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.asmnet2.asmnetwork(ora.asmgroup)
      1        ONLINE  ONLINE       orac19c1                 STABLE
      2        ONLINE  ONLINE       orac19c2                 STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.cvu
      1        ONLINE  ONLINE       orac19c2                 STABLE
ora.mgmtdb
      1        ONLINE  ONLINE       orac19c2                 Open,STABLE
ora.orac19c1.vip
      1        ONLINE  ONLINE       orac19c1                 STABLE
ora.orac19c2.vip
      1        ONLINE  ONLINE       orac19c2                 STABLE
ora.qosmserver
      1        ONLINE  ONLINE       orac19c2                 STABLE
ora.rac19c.db
      1        ONLINE  ONLINE       orac19c1                 Open,HOME=/u01/app/o
                                                             racle/product/19.0.0
                                                             /dbhome_1,STABLE
      2        ONLINE  ONLINE       orac19c2                 Open,HOME=/u01/app/o
                                                             racle/product/19.0.0
                                                             /dbhome_1,STABLE
ora.scan1.vip
      1        ONLINE  ONLINE       orac19c1                 STABLE
ora.scan2.vip
      1        ONLINE  ONLINE       orac19c2                 STABLE
ora.scan3.vip
      1        ONLINE  ONLINE       orac19c2                 STABLE
--------------------------------------------------------------------------------
[root@orac19c1 ~]#
```

Oracle RAC cluster CRS and vote disk:

```
[root@orac19c1 ~]# /u01/app/19.0.0/grid/bin/ocrcheck
Status of Oracle Cluster Registry is as follows :
         Version                  :          4
         Total space (kbytes)     :     491684
         Used space (kbytes)      :      84576
         Available space (kbytes) :     407108
         ID                       : 1989355982
         Device/File Name         :    +CRS_DG
                                    Device/File integrity check succeeded

                                    Device/File not configured

                                    Device/File not configured

                                    Device/File not configured

                                    Device/File not configured

         Cluster registry integrity check succeeded

         Logical corruption check succeeded

[root@orac19c1 ~]# /u01/app/19.0.0/grid/bin/crsctl query css votedisk
##  STATE    File Universal Id                File Name Disk group
--  -----    -----------------                --------- ---------
 1. ONLINE   ca89a1fe7b154f23bf50aebdaceb1fa4 (ORCL:CRS_DISK02) [CRS_DG]
 2. ONLINE   5522f093e3114fe7bf94573e16d0e749 (ORCL:CRS_DISK03) [CRS_DG]
 3. ONLINE   adebaff971df4f06bfacf2132a77b38e (ORCL:CRS_DISK01) [CRS_DG]
 4. ONLINE   d7ad7ce93fff4f10bf1980b47d8b82a7 (ORCL:CRS_DISK04) [CRS_DG]
 5. ONLINE   9de01fc3bd5d4f94bf35149ebd8fe42a (ORCL:CRS_DISK05) [CRS_DG]
Located 5 voting disk(s).
[root@orac19c1 ~]#
```

Oracle RAC cluster mode and GIMR:

```
[root@orac19c1 ~]# /u01/app/19.0.0/grid/bin/crsctl get cluster mode status
Cluster is running in "flex" mode
[root@orac19c1 ~]#
[root@orac19c1 ~]# /u01/app/19.0.0/grid/bin/olsnodes -n
orac19c1        1
orac19c2        2
[root@orac19c1 ~]#
[root@orac19c1 ~]# /u01/app/19.0.0/grid/bin/srvctl status mgmtdb
Database is enabled
Instance -MGMTDB is running on node orac19c1
```

Oracle RAC VM Anti-Affinity Rules

As previously discussed, VMware DRS affinity and anti-affinity rules help distributed resource scheduler (DRS) perform better placements of VMs by understanding the application dependencies and availability.

VM-VM anti-affinity rules can be used to specify anti-affinity between VMs, as DRS attempts to keep the specified VMs apart.

An important aspect of any Oracle RAC configuration on VMware SDDC setup is to ensure that Oracle RAC VMs are not scheduled on the same physical ESXi server, as same-server scheduling negates the HA inherent to VMware SDDCs (i.e., the infrastructure-level HA provided by VMware SDDC as a complement to the application level HA provided by Oracle RAC).

To achieve this, it's necessary to specify VM anti-affinity rules for Oracle RAC VMs, forcing the VMs to remain apart during normal or failover actions.

Learn more about *VMware HA and DRS affinity rules*.

In the example below, Oracle RAC VMs **orac19c1** and **orac19c2** are assigned anti-affinity rule set **RAC_AntiAffinity_orac19c**:

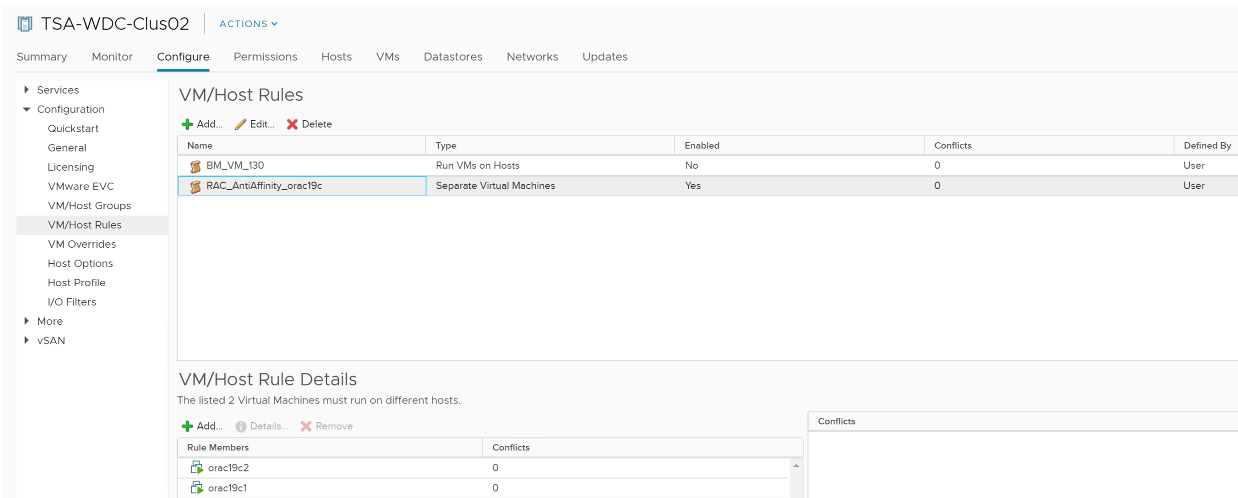


FIGURE 93. ORACLE RAC VM ANTI-AFFINITY RULE SET RAC_ANTIAFFINITY_ORAC19C

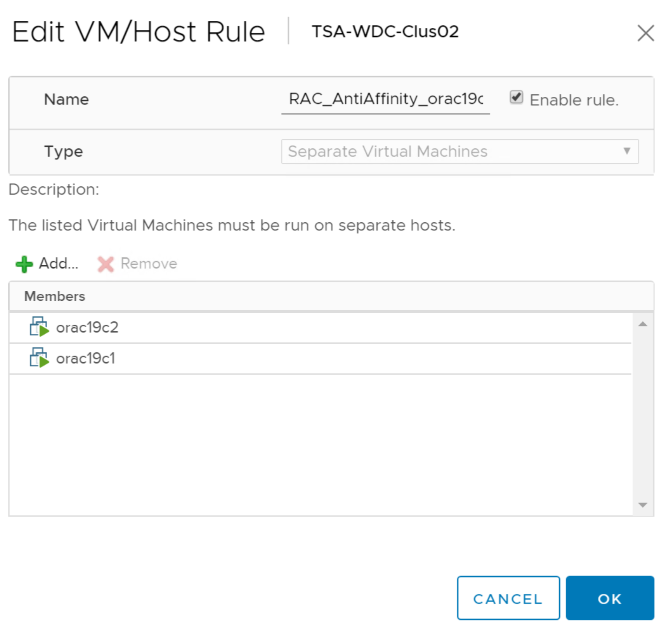


FIGURE 94. ORACLE RAC VM ANTI-AFFINITY RULE SET RAC_ANTIAFFINITY_ORAC19C

For extended Oracle RAC, best practice suggests spreading Oracle RAC VMs across multiple sites to ensure HA across those sites. This is in addition to infrastructure and application-level HA, which ensures all Oracle RAC VMs do not land on the same site at the same time.

This is achieved using compute policies and tags, along with attributes which allow attachment of metadata to objects in the vSphere inventory, simplifies the sorting and searching process.

A tag is a label applied to objects in a vSphere inventory. When creating a tag, a category is assigned to the tag, which allows grouping of related tags together. When defining a category, be sure to specify the object types for its tags and whether more than one tag in the category can be applied to an object.

For vSphere tags and attributes, VMware Cloud on AWS supports the same set of tasks as an on-premises SDDC.

Learn more about *vSphere tags and attributes*.

Oracle Restart with VMware vSphere HA

VMware vSphere HA monitors the ESXi hosts in the cluster and, in the event of a host failure, the VMs on the failed host are restarted on alternate hosts.

Oracle Restart automatically restarts various database components after a hardware or software failure, or whenever the database host computer restarts. Oracle Restart runs periodic check operations to monitor the health of these components. If a check operation fails for a component, the component is shut down and restarted.

Although VMware vSphere High Availability is a cost-effective solution to protect against hardware failures, there will be some database downtime pending Oracle instance recovery. Oracle instance recovery will need to apply pending updates in redo logs and roll back the uncommitted transactions.

The amount of time required for instance recovery depends on three factors, including the amount of redo generated, user configurable parameters, and the database version. It's therefore necessary to factor in time for the Oracle database (using Oracle Restart with VMware HA) to come up. The amount of time needed is determined by the time needed for the VM to restart, the OS to reboot, and the Oracle instance to start and complete instance recovery.

It's also important to note that VMware HA does not protect against any kind of downtime for guest operating systems or Oracle patching.

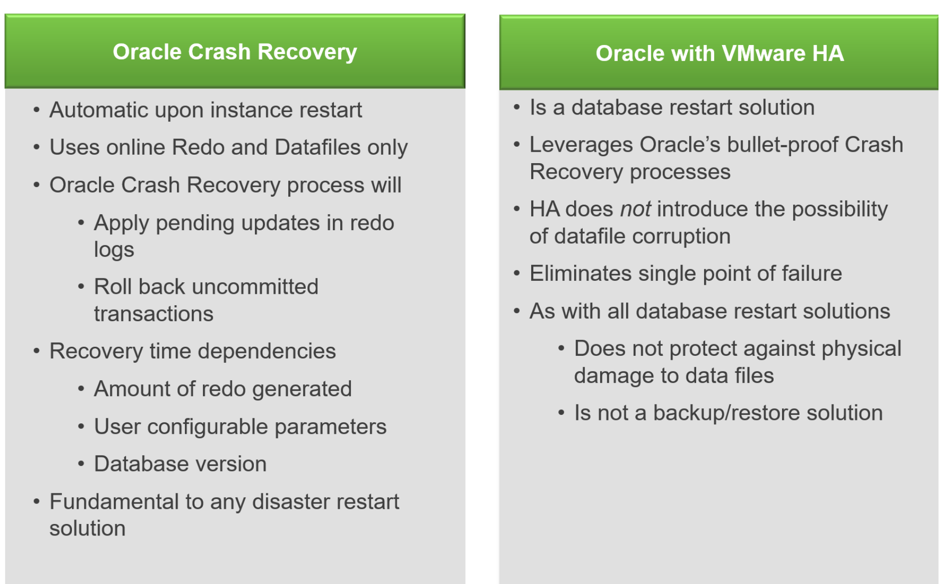


FIGURE 96. ORACLE RESTART WITH VMWARE HA

Summary of Oracle High Availability Options

High availability is the design and implementation of a service to create a system that can be operational during a given measurement period. It is factored into system design to protect business-critical operations from both unplanned downtime (e.g., an outage due to infrastructure failure) and planned downtime (e.g., an outage required for system maintenance).

The Oracle High Availability options to protect a VMware virtualized Oracle database and minimize downtime include:

- Oracle Restart with VMware HA
- Oracle RAC One Node with VMware HA
- Oracle RAC with VMware HA

The following scenarios are compared to help system architects decide which HA solution for Oracle on VMware is best for their environment:

Oracle DB in VM Scenario	ESX Server Protection	Oracle DB Protection	Minimizes Downtime for Guest OS & Oracle Patching	Oracle Available during Failover	Session/Select Persistance through Failover Oracle TAF	Cost/ Complexity
Oracle Restart with VMware HA	YES	NO	NO	NO	MED	LOW
Oracle RAC One Node with VMware HA	YES	YES	YES	NO	MED	HIGH
Oracle RAC with VMware HA	YES	YES	YES	YES	VERY HIGH	VERY HIGH

FIGURE 97. ORACLE HIGH AVAILABILITY OPTIONS AND SCENARIOS

Consider the following points when choosing a high availability solution for Oracle databases on VMware:

- High levels of uptime cannot realistically be achieved with any of the above scenarios alone, as overall system availability also depends on redundancy designed into the other parts of the infrastructure (e.g., network, power, and storage).
- What is the Service Level Agreement (SLA) for the business with respect to uptime/downtime? How much downtime is the business willing to tolerate?
- Oracle RAC with VMware vSphere HA solutions provide the highest degrees of protection and are preferred options for businesses that require near-zero Oracle Database downtime, in addition to the ability to perform rolling patch upgrades, load-balancing for better performance, workload scalability, cost-effective resource management, and continuous application availability using Oracle Transparent Application Failover (TAF).
- For organizations with a tolerance for limited downtime (i.e., those that can accept some loss of availability for software maintenance), Oracle Restart with VMware vSphere HA offers a cost-effective solution to address required service levels.
- What is the business cost or uptime trade-off? Oracle RAC deployments are generally more expensive, yet VMware offers the most cost-efficient solution. Willingness to incur additional costs for increased availability is a key consideration.

The following figure charts the degree of availability in a deployment against the associated cost to deliver each solution. A pattern of diminishing returns exists where, at some point, an organization receives fewer and fewer additional levels of availability for additional investment into the solution.

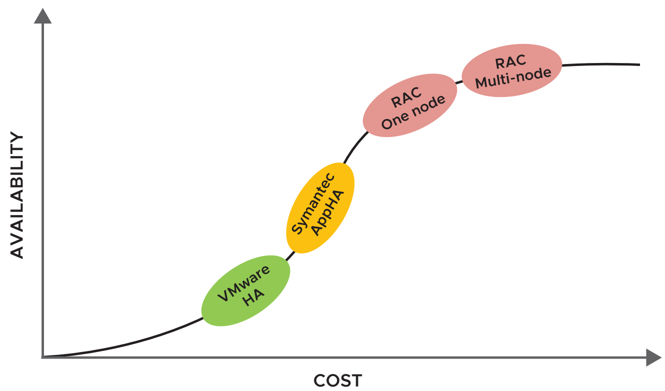


FIGURE 98. AVAILABILITY VERSUS COST TRADE-OFF

The final design choice depends on the amount of downtime a business can tolerate balanced against the cost of additional resources and skills to install and operate software for Oracle RAC. There is a clear tradeoff.

Conclusion

Customers deploying Oracle RACs do so in the face of important requirements such as stringent SLAs, continued high performance, and application availability.

Deploying Oracle RAC on physical architecture is subjected to issues and restrictions similar to those running an Oracle Single Instance on physical architecture, namely hardware failure due to a failed component, power outage, or complete hardware meltdown.

Customers have successfully run their business-critical Oracle workloads with high performance demands on VMware vSphere for many years. VMware vSphere provides high availability natively at the infrastructure level and is completely complementary to the application-level high availability that Oracle RAC provides.

With more and more production servers now virtualized, the demand for highly converged, server-based storage is surging. VMware vSAN provides highly scalable, available, reliable, and high-performance storage using cost-effective hardware (i.e., direct-attached disks in VMware ESXi hosts).

vVols is an integration and management framework that virtualizes SAN/NAS arrays, enabling a more efficient operational model that is optimized for virtualized environments and centered on the application instead of the infrastructure. It simplifies the delivery of storage service levels to individual applications by providing finer control of hardware resources and native array-based data services that can be instantiated with VM granularity.

Extended Oracle RAC provides greater availability than local Oracle RAC. It provides extremely fast recovery from a site failure and enables all servers, in all sites, to actively process transactions as part of a single database cluster. It enables transparent workload sharing, workload balancing, site maintenance without service disruption and high availability across sites.

VMware vSAN Stretched Cluster enables active/active data centers that are separated by metro distance. Extended Oracle RAC with VMware vSAN Stretched Cluster enables transparent workload sharing between two sites accessing a single database while providing the flexibility of migrating or balancing workloads between sites in anticipation of planned events such as hardware maintenance.

vMSC is a specific storage configuration commonly referred to as *stretched storage clusters or metro storage clusters*. These configurations are usually implemented in environments where disaster and downtime avoidance are a key requirement.

VMware Cloud on AWS is an on-demand service that enables customers to run applications across vSphere-based cloud environments, with access to a broad range of AWS services. Powered by VMware Cloud Foundation, this service integrates vSphere, vSAN and NSX along with VMware vCenter management, and is optimized to run on dedicated, elastic, bare-metal AWS infrastructure. ESXi hosts in VMware Cloud on AWS reside in an AWS AZs and are protected by vSphere HA.

Stretched Clusters for VMware Cloud on AWS is designed to protect against an AWS AZ failure. Applications can span multiple AWS AZs within a VMware Cloud on AWS cluster.

An important aspect of any Oracle RAC configuration on VMware SDDC setup is to ensure that Oracle RAC VMs are not scheduled on the same physical ESXi server, as same-server scheduling negates the HA value proposition that VMware SDDC inherently provides (i.e., infrastructure level HA provided by VMware SDDC, complementing the application level HA provided by Oracle RAC).

To achieve this, it's necessary to specify VM anti-affinity rules for Oracle RAC VMs, forcing the VMs to remain apart during normal or even during failover actions.

This solution primarily validated the functional design of various Oracle RAC deployments, both traditional and extended, on various VMware platforms:

- VMFS datastore
- NFS datastore
- vSAN datastore
- vVol datastore
- VMware vSphere Metro Storage Cluster
- VMware vSAN Stretched Cluster

These deployments are summarized in the table below:

Location	Oracle RAC Deployment	DataStore Type	DataStore Name	Storage Vendor
On-premise	Traditional	VMFS FC	TSA_PURE_FLASH_4TB_01	Pure x50 AFA Storage
On-premise	Traditional	VMFS NFS	TSA_TNTR_Oracle	Tintri T880 AFA Storage
On-premise	Traditional	vsanDatastore	vsanSiteB	vSAN 6.7 AFA Hyperconverged
On-premise	Traditional	VMware vVols	TSA_PURE_FLASH_VVOL	Pure x50 AFA Storage
On-premise	Extended	Vendor Specific	Vendor Specific	vSphere Metro Storage Cluster
On-premise	Extended	vsanDatastore	vsanDatastore	vSAN 6.7 AFA Hyperconverged

TABLE 28. ORACLE RAC DEPLOYMENTS

Various test scenarios were conducted, including:

- Abrupt termination of Oracle RAC crsd, ocssd, and evmd processes, while observing if the process is restarted or if the node is restarted by the cluster
- Resiliency testing of Oracle RAC private interconnect, scan listeners, and agent
- vMotion of online Oracle RAC cluster (more details found *here*). The demo of this study can be found *here*.
- In case of Oracle RAC on vSAN storage, disk failure, disk group failure, and storage host failure scenarios were conducted. Details of the test and the test results can be found *here*.

These scenarios are compared to help system architects decide which HA solution for Oracle on VMware is best for their environment.

Oracle DB in VM Scenario	ESX Server Protection	Oracle DB Protection	Minimizes Downtime for Guest OS & Oracle Patching	Oracle Available during Failover	Session/Select Persistance through Failover Oracle TAF	Cost/ Complexity
Oracle Restart with VMware HA	YES	NO	NO	NO	MED	LOW
Oracle RAC One Node with VMware HA	YES	YES	YES	NO	MED	HIGH
Oracle RAC with VMware HA	YES	YES	YES	YES	VERY HIGH	VERY HIGH

FIGURE 99. ORACLE HA OPTIONS AND SCENARIOS

Appendix A Oracle Initialization Parameter Configuration

Oracle initialization parameters

*.audit_trail='db'

*.audit_sys_operations=TRUE

rac19c1.audit_file_dest='/u01/admin/rac19c1/adump'

rac19c2.audit_file_dest='/u01/admin/rac19c2/adump'

*.cluster_database=true

*.compatible=12.1.0.0.0

*.control_files='+DATA_DG/rac19c/control01.ctl','+DATA_DG/rac19c/control02.ctl','+DATA_DG/rac19c/control03.ctl'

*.db_block_size=8192

*.db_cache_advice='ON'

*.db_domain=''

*.db_name='rac19c'

*.db_recovery_file_dest='+FRA_DG'

*.db_recovery_file_dest_size=200G

rac19c1.diagnostic_dest='/u01/admin/rac19c1'

rac19c2.diagnostic_dest='/u01/admin/rac19c2'

*.enable_pluggable_database=true

rac19c1.instance_number=1

rac19c2.instance_number=2

*.log_archive_format='%t_%s_%r.dbf'

*.sga_max_size=48G

*.sga_target=48G

*.open_cursors=1000

*.processes=1000

*.parallel_max_servers=100

*.pga_aggregate_target=4G

*.pga_aggregate_limit=6000M

*.remote_listener='rac19c-scan:1521'

*.remote_login_passwordfile='exclusive'

rac19c1.thread=1

rac19c2.thread=2

rac19c1.undo_tablespace='UNDOTBS01'

rac19c2.undo_tablespace='UNDOTBS02'

*.pre_page_sga=TRUE

*.use_large_pages='only'

Reference

White Paper

For additional information, see the following white papers:

- *Oracle Databases on VMware Best Practices Guide*
- *Enabling or disabling simultaneous write protection provided by VMFS using the multi-writer flag (1034165)*
- *Using Oracle RAC on a vSphere 6.x vSAN Datastore (2121181)*
- *Oracle Real Application Clusters on VMware vSAN*
- *Extended Oracle RAC on VMware vSAN Stretched Cluster*
- *Oracle RAC on vSAN 6.7 P01 – No more Eager Zero Thick requirement for shared VMDKs*
- *Oracle RAC storage migration from non-vSAN to vSAN 6.7 P01 – Through Thick to Thin*
- *Whats New in Virtual Volumes (vVols) 2.0*
- *Oracle Databases on VMware High Availability*

Product Documentation

For additional information, see the following product documentation:

- *Oracle 19c Database Online Documentation*
- *VMware vSAN 6.7*
- *VMware Virtual Volumes (vVols)*
- *VMware vSphere Metro Storage Cluster*
- *VMware vSAN Stretched Cluster*
- *VMware Cloud on AWS*
- *Stretched Clusters for VMware Cloud on AWS*

Other Documentation

For additional information, see the following document:

- *VMware Solutions Lab*
- *SLOB Resources*

Acknowledgements

Author: Sudhir Balasubramanian, Staff Solution Architect, works in the Cloud Platform Business Unit (CPBU). Sudhir specializes in the virtualization of Oracle business-critical applications. Sudhir has more than 25 years' experience in IT infrastructure and database, working as the Principal Oracle DBA and Architect for large enterprises focusing on Oracle, EMC storage, and Unix/Linux technologies. Sudhir holds a master's degree in computer science from San Diego State University. Sudhir is one of the authors of the *Virtualize Oracle Business Critical Databases* book, which is a comprehensive authority for Oracle DBAs on the subject of Oracle and Linux on vSphere. Sudhir is a VMware vExpert, Ex-Member of the VMware CTO Ambassador Program and an Oracle ACE.

Thanks to the following for their reviews and inputs:

- Don Sullivan - Product Line Marketing Manager for Business-Critical Applications
- Oleg Ulyanov - Senior Solution Architect, Cloud Platform Technical Marketing