ORACLE

Oracle Database and Oracle Real Application Cluster on Oracle Linux KVM

Best Practices for Oracle Database and Oracle Real Application Cluster on Oracle Linux KVM

Purpose statement

This document provides an overview of best practices for Oracle Database on Oracle Linux KVM. It is intended solely to help you assess the business benefits of running Oracle Database solution on private cloud architectures based on Oracle Linux KVM and Oracle Linux Virtualization Manager.

Disclaimer

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. This document and information contained herein may not be disclosed, copied, reproduced or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

ORACLE

Table of Contents

Executive summary

Running today's non-critical business applications in virtualized environments has shown to be efficient and cost effective. More sophisticated or highly available applications on the other hand are most likely incompatible with commonly used software-based virtualization solutions.

> ✓ **Oracle Linux KVM and Oracle Real Application Clusters (RAC) enable the benefits of a modern cloud-oriented data center for highly available applications.**

Oracle now offers a highly available, cloud-ready virtualization solution for your data center with Oracle Linux KVM, a modern and cloud-oriented software-based virtualization infrastructure and Oracle Real Application Cluster, the leading high availability solution for Oracle Database.

The combination enables better server consolidation (Oracle RAC databases with underutilized CPU resources or peaky CPU utilization can often benefit from consolidation with other workloads using server virtualization) sub-capacity licensing, and rapid provisioning; this is accomplished through Oracle RAC and Oracle Linux KVM.

Oracle RAC on OL-KVM also support the creation of non-production virtual clusters on a single physical server for product demos, educational settings, and test environments.

This deployment combination permits dynamic changes to pre-configured database resources for agile responses to changing service level requirements common in consolidated environments.

> ✓ **For x86-64 hardware platforms, Oracle Linux KVM is the go-forward virtualization solution that is certified for Oracle Real Application Clusters on Linux.**

This paper discusses various Oracle RAC deployment scenarios and provides best practices for an optimized Oracle RAC deployment on Oracle Linux KVM environments. Version specific information is noted accordingly.

ORACLE

Introduction

> ✓ **Oracle constantly tests additional and advanced Oracle Linux KVM features with Oracle RAC. This paper is updated regularly, as new test results are available.**

Oracle Real Application Clusters (RAC)[1] is an option to the award-winning Oracle Database Enterprise Edition. Oracle RAC is a cluster database with a shared cache architecture that overcomes the limitations of traditional shared-nothing and shared-disk approaches to provide highly scalable and available database solutions for all your business applications. Oracle RAC on top of Oracle Linux KVM, can be used as the foundation of a database cloud system as well as a shared infrastructure that can help ensure high availability, scalability, flexibility, and agility for your data-intensive applications.

The family of Oracle Real Application Clusters solutions includes **Oracle Clusterware,**[2] which provides group membership, high availability, and application resource management. Oracle Clusterware is the technology that transforms a server farm into a cluster. A cluster, in general, is a group of independent servers (virtual or physical) that cooperate as a single system for robust high availability and agile scalability. Oracle Clusterware is the intelligence in this system that allows the required cooperation and is a key component of Oracle Real Application Cluster.

Oracle RAC utilizes **Oracle Automatic Storage Management (ASM)**[3] for efficient shared storage access. Oracle ASM acts as the underlying, clustered volume manager. It provides the database administrator with a simple storage management interface that is consistent across all server and storage platforms. As a vertically integrated file system and volume manager, purpose-built for Oracle Database files, Oracle ASM provides the performance of raw I/O with the easy management of a file system. It also provides the basis for a shared storage pool in Oracle Real Application Cluster.

Leveraged by best-in-class Oracle Exadata and Oracle Cloud Infrastructure, **Oracle Linux KVM**[4] provides a set of modules that enable you to use the *Oracle Linux* kernel as a hypervisor. Oracle Linux KVM enables deployment of operating systems and application software within a supported virtualization environment. Oracle Linux KVM complements the foundation of a database cloud system by providing an Oracle RAC certified virtualization environment.

What is Oracle Real Application Clusters (RAC)?

Oracle Real Application Clusters (RAC) allow customers to run a single Oracle Database across multiple servers in order to maximize availability and enable horizontal scalability, while accessing shared storage. User sessions connecting to Oracle RAC instances can failover and safely replay changes during outages, without any changes to end user applications, hiding the impact of the outages from end users.

Oracle RAC is a clustered database. A cluster is a group of independent servers that cooperate as a single system. Clusters provide improved fault resilience and modular, incremental system growth over single symmetric multiprocessor (SMP) systems.

---

[1] Oracle Real Application Clusters (RAC) homepage

[2] More information on Oracle Clusterware

[3] More information on Oracle Automatic Storage Management (ASM)

[4] Oracle Linux KVM homepage

> ✓ **Oracle RAC enables Oracle Database to run mainstream business applications on clusters, including popular packaged products (such as Oracle E-Business Suite, Oracle's PeopleSoft, Siebel CRM, SAP) and in-house developed applications generating OLTP, DSS, or mixed workloads.**

In the event of a system failure, clustering helps ensure the highest availability to users and access to mission critical data is not lost. Redundant hardware components such as additional nodes, interconnects, and disks, allow the cluster to provide high availability. Such redundant hardware architectures avoid single points-of-failure and provide exceptional fault resilience.

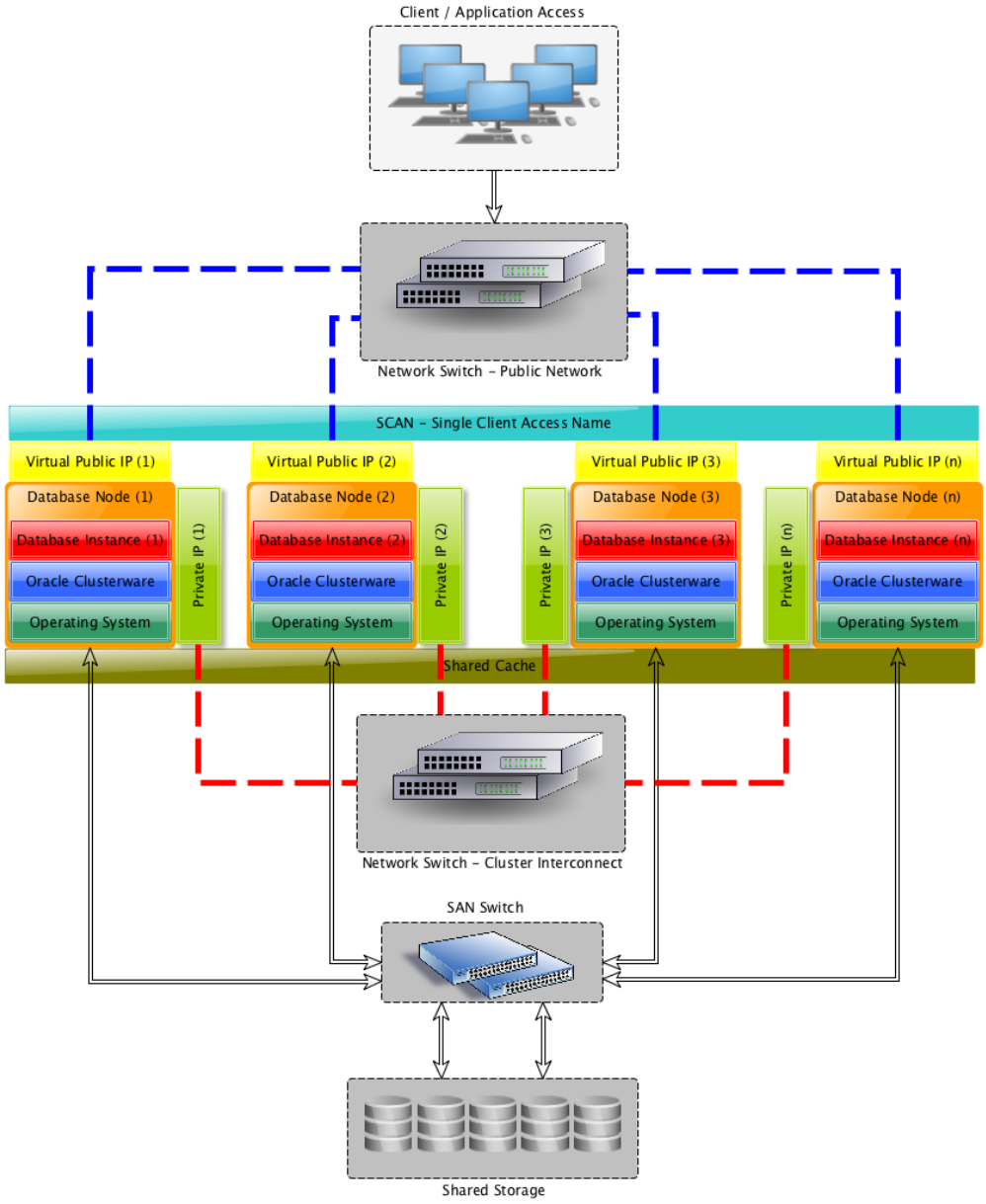


Figure 1: Oracle Real Application Clusters overview

Oracle RAC allows users to access their data, stored on shared storage, from any instance of the cluster.

However, a clustered database (using more than one instance) differs from a single instance database in that the database can be accessed by multiple instances concurrently. Each instance runs on a separate server in the cluster (formed by Oracle Clusterware).

When additional resources are required, additional nodes and instances can easily be added to the cluster with no downtime. Once a new instance has been started, applications using services can immediately take advantage of it, without changes to the application or application server.

What is Oracle RAC One Node?

Oracle Real Application Clusters (RAC) One Node[5] improves upon many of the benefits of server virtualization and extends them to databases running in physical server environments. Oracle RAC One Node enables:

- Better server consolidation
- Enhanced protection from failures
- Greater flexibility and workload management
- Better online maintenance

In addition, it allows customers to virtualize database storage, standardize their database environment, and, should the need arise, upgrade to a full multi-node Oracle RAC database without downtime or disruption. Further, it is compatible with and complementary to Oracle Linux KVM, allowing customers to build environments that leverage the strengths of both Oracle RAC One Node and Oracle Linux KVM virtualization.

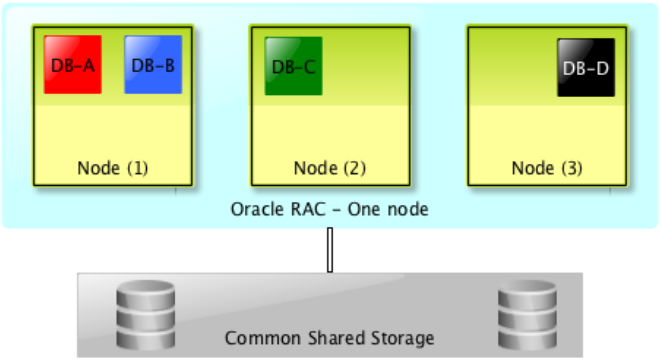


Figure 2: Oracle RAC One Node - typical architecture overview

In the configuration shown in Figure 2, four single-instance Oracle RAC One Node databases are running in a cluster of three servers. Server A is hosting Oracle RAC One Node databases DB-A and DB-B, server B is hosting database DB-C and server C is hosting databases DB-D. Each server runs one operating system (OS). In servers A and C above, multiple databases are consolidated onto a single OS.

What is an Extended Distance Oracle RAC?

Oracle RAC on extended distance (Stretched Cluster) is an architecture that 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.

Oracle RAC on Extended Distance Clusters provides greater high availability than a local Oracle RAC implementation. However, it does not provide full disaster recovery. It offers protection for some disasters (e.g. local power outage or

[5] More information on Oracle RAC One Node

server room flooding) but not all. Disasters such as earthquakes, hurricanes, and regional floods may affect a greater area. Oracle RAC on Extended Distance Clusters does not protect from human errors or corruptions in the shared storage.

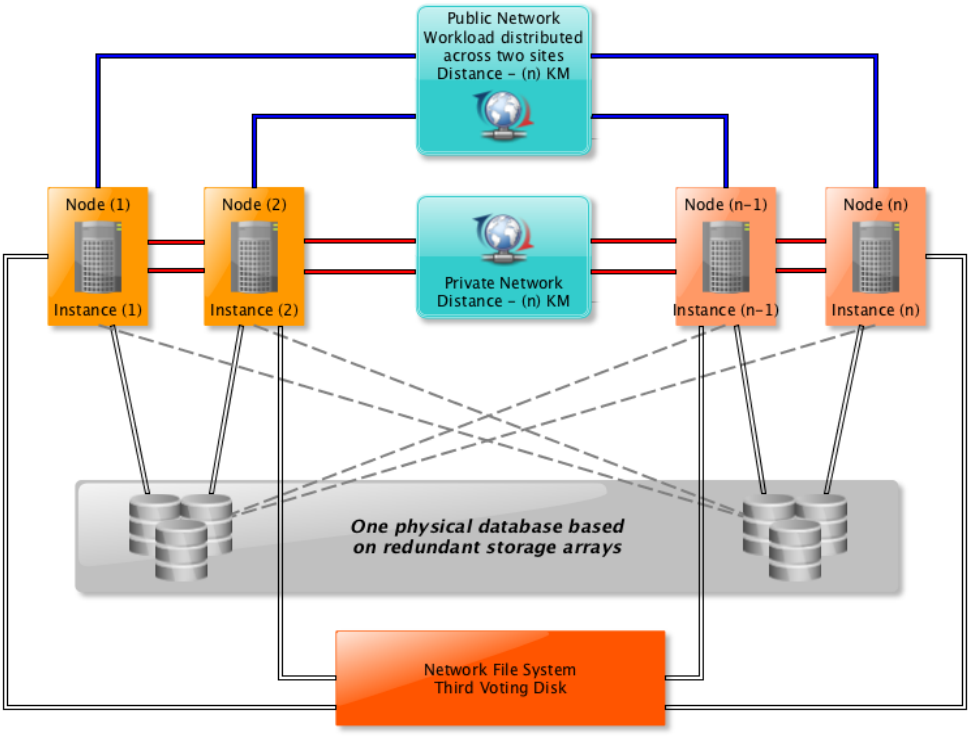


Figure 3: Oracle Extended RAC - architecture overview

For comprehensive data protection, including protection against corruptions and regional disasters, Oracle recommends using Oracle Data Guard together with Oracle RAC. These products are among the building blocks of Oracle's Maximum Availability Architecture (MAA). For active-active databases deployed across geographically separated data centers, the MAA recommendation is to use Oracle GoldenGate. Data Guard and GoldenGate also provide additional benefits such as minimizing downtime for maintenance activities including upgrades and migrations.

Clusters of this kind have been referred to by many names, including campus clusters, metro clusters, geo-clusters, stretched clusters and extended clusters.

Oracle RAC One Node completes the Extended Distance Cluster offering. While using the same Oracle RAC infrastructure (i.e. Oracle Clusterware and Oracle ASM), Oracle RAC One Node is best used in environments or for applications that do not require the degree of HA or scalability that an active / active Oracle RAC database provides, but which benefit from an integrated and fully supported failover solution for Oracle Database.

What is Oracle Linux KVM (OL-KVM)?

Oracle Linux KVM is a feature of Oracle Linux. With the Unbreakable Enterprise Kernel (UEK) Release 5, the Oracle Linux server virtualization solution with KVM has been enhanced. Users can take either a previously deployed version of Oracle Linux and turn the OS into a KVM host, or a KVM configuration can be set up from a base Oracle Linux installation. Oracle Linux KVM is the same hypervisor used in Oracle Cloud Infrastructure, giving users an easy migration path to move workloads into Oracle Cloud.

What is Oracle Linux Virtualization Manager (OLVM)?

To support multiple hosts running Oracle Linux KVM, IT administrators can use Oracle Linux Virtualization Manager, which is built from the open source oVirt project. The heart of this management solution is the ovirt-engine, which is used to discover KVM hosts and configure storage and networking for the virtualized data center. Oracle Linux Virtualization Manager offers a web-based user interface (UI) and a representational state transfer (REST) application programming interface (API) which can be used to manage your Oracle Linux KVM infrastructure. Oracle Linux Virtualization Manager allows enterprise customers to continue supporting their on-premises data center deployments with the KVM hypervisor available in Oracle Linux 8 with Unbreakable Enterprise Kernel Release 7.

For most day-to-day operations, many users will rely on the administration portal or the lighter weight VM portal. These portals can be accessed from the Oracle Linux Virtualization Manager landing page when first connected with a browser.

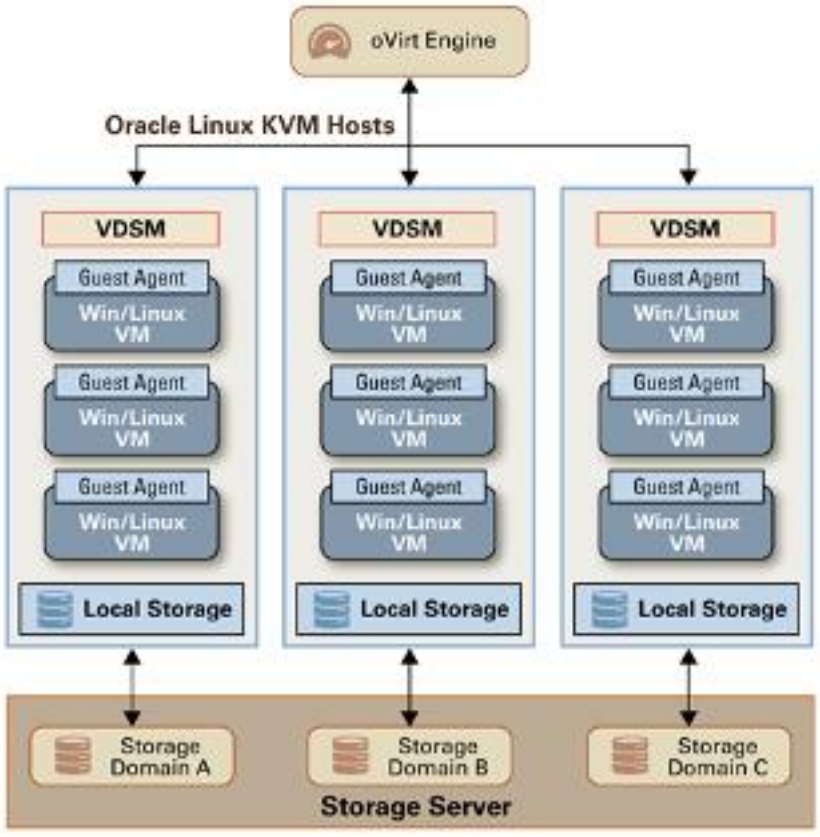


Figure 4: Oracle Linux Virtualization Manager overview

✓ **Oracle Linux KVM combined with Oracle Linux Virtualization Manager supports the stretched/extended configuration and Oracle RAC Extended can be deployed on one stretched Datacenter as well as on top of two different Datacenters.**

ORACLE

Why run Oracle RAC on Oracle Linux KVM?

There are several reasons why customers choose to run Oracle RAC in an Oracle Linux KVM environment. Some of the more common ones are:

- **Server Consolidation:** Oracle RAC, Oracle RAC One Node, or single instance databases with underutilized CPU resources or variable CPU utilization can often benefit from consolidation with other workloads using server virtualization. A typical use case for this scenario would be the consolidation of several Oracle databases into a single Oracle RAC database or multiple Oracle RAC databases where the hosting Oracle Linux KVM guests have pre-defined resource limits configured for each VM guest, e.g. CPU resources. Each Oracle Linux KVM hosting an Oracle RAC instance can be set up with pre-determined, fixed, and limited number of resources maximizing aggregate capacity across all Oracle Linux KVMs and Oracle RAC instances. Each Oracle Linux KVM and hosted Oracle RAC instance are sufficiently isolated so as not to interfere with other Oracle instances sharing the same physical hardware. Oracle Database with Oracle Multitenant, delivers an architecture that allows a multitenant container database to hold many pluggable databases. Oracle Multitenant complements other options, including Oracle Real Application Clusters and Oracle Active Data Guard.
- **Sub-capacity licensing:** The current Oracle licensing model requires the Oracle RAC database to be licensed for all CPUs on each server in the cluster. Sometimes customers wish to use only a subset of the CPUs on the server for a particular Oracle RAC database. Oracle Linux KVM can be configured in such way that it is recognized as a hard partition. Hard partitions allow customers to only license those CPUs used by the partition instead of licensing all CPUs on the physical server. For more information on using hard partitioning with Oracle Linux KVM refer to the "Hard Partitioning with Oracle Linux KVM"[6] technical paper.
- **Create a virtual cluster:** Oracle Linux KVM enables the creation of a virtual cluster on a single physical server. This use case is particularly interesting for product demos, educational settings, and test environments. However, it should never be used to run production Oracle RAC environments. The following are valid deployments for this use case:
    - Test / development cluster
    - Demo cluster
    - Education cluster

✓ **Another option for building Oracle RAC environments for test, demo, or educational purposes is based on "Vagrant Projects," available on GitHub. This solution leverages Vagrant and Oracle VirtualBox and runs on x86 operating systems. Further details are available at:**

https://github.com/oracle/vagrant-projects

- **Rapid Provisioning:** The provisioning time of a new application consists of the server (physical or virtual) deployment time and the software install and configuration time. Oracle Linux KVM can help reduce the deployment time for both of those components. Oracle Linux KVM supports the ability to create deployment templates. These templates can then be used to rapidly provision new (Oracle RAC) systems. Oracle provides a rich portfolio of pre-configured, certified, gold image deployment templates[7] for Oracle Linux KVM for Oracle Real Application Clusters, applications, and operating systems.

[6] Hard Partitioning with Oracle Linux KVM
[7] Oracle Linux Templates for Oracle Linux KVM

Business continuity, high availability, and scalability

Business continuity is a key element in today's business and while Oracle RAC is the ultimate solution when it comes to Oracle databases, there are more options to choose from when considering virtualization technologies.

Two high availability (HA) mechanisms can be distinguished: Oracle Linux KVM HA (external HA) and Oracle Clusterware-based, internal HA (used for Oracle RAC) as illustrated in Figure 5.

When high-availability is granted by Oracle Clusterware the virtual machine(s) part of the Oracle Database cluster have to be configured through Oracle Linux Virtualization Manager, as in the following example:

- High available: "**disabled**"
- Target Storage Domain for VM Lease: "**No VM Lease**"
- Resume Behavior: "**Kill**"

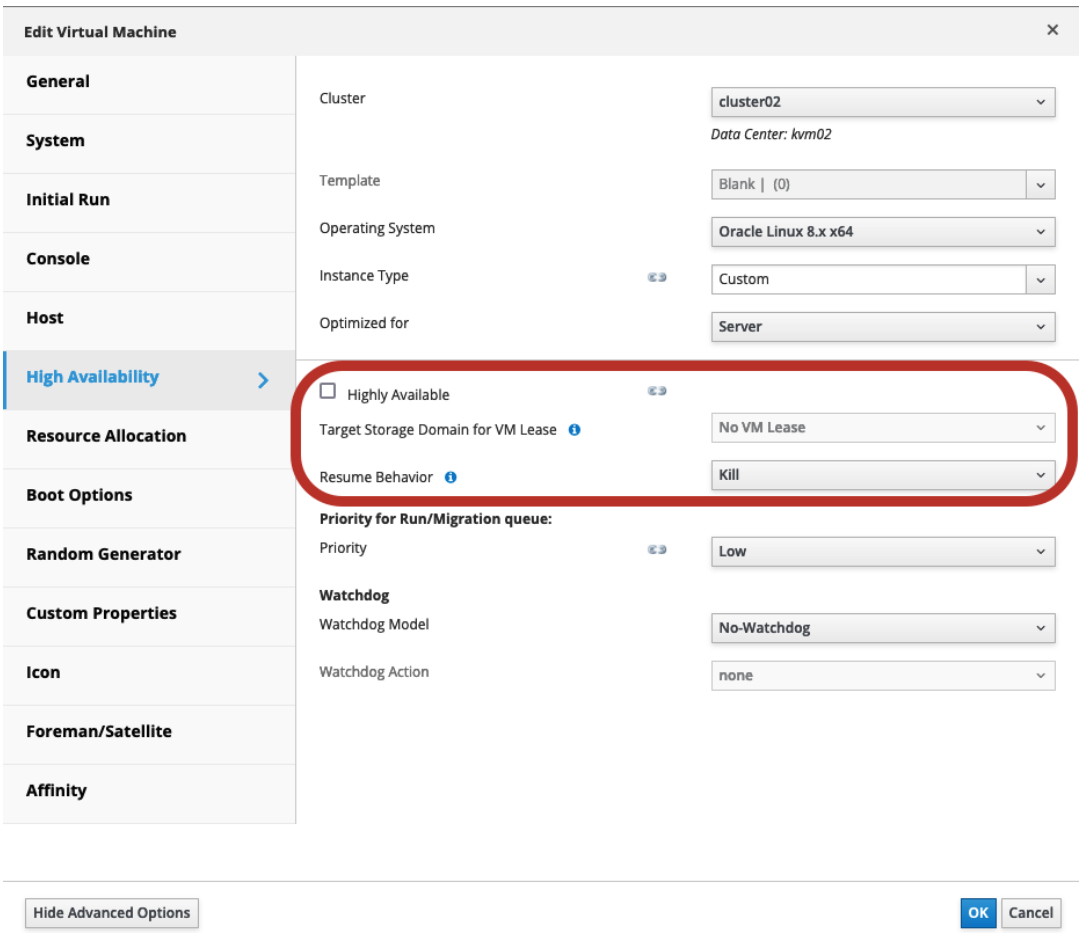


Figure 5: High availability configuration for H/A Oracle Database virtual machines

✓ ***Note: If the high-availability configuration for the virtual machines is not properly configured, any pause or suspend operation, due to possible storage issues, could trigger Oracle Clusterware to fence the suspended virtual machine or, in worst cases, lead to a block corruption.***

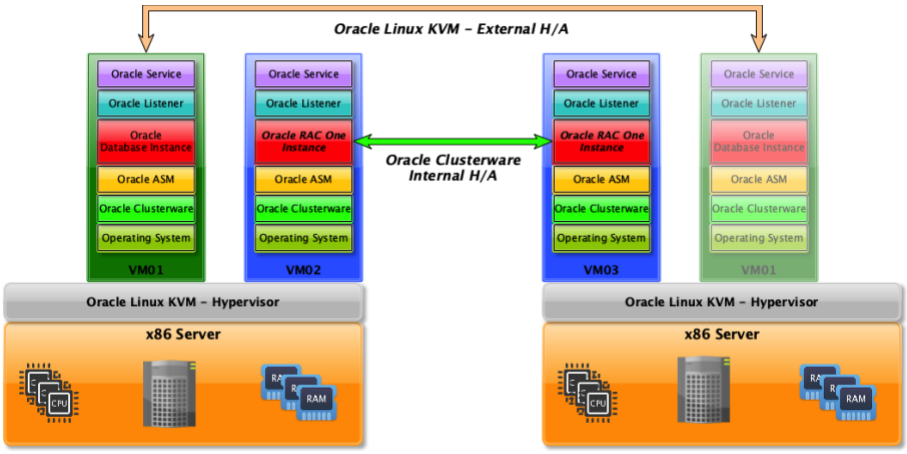


Figure 6: Oracle Linux KVM HA and Oracle RAC

In general, Oracle Linux KVM HA enables the restart of an Oracle Linux KVM guest on either the same physical machine or a different machine, if more than one physical machine is available in the server pool. In any case, Oracle Linux KVM would operate on the virtual machine, restarting whatever is running on it.

The reason is that the Oracle Linux KVM guest is usually unaware of the applications running in the virtualized environment as much as the application is typically unaware of the Oracle Linux KVM guest it is running in.

Oracle Clusterware provides faster, fine-grained recovery from a process/application failure within a guest.

Utilizing application specific agents, these solutions will perform corrective actions particular to the failure without the overhead of restarting the whole Oracle Linux KVM guest.

While a combination of both an internal application HA and the external Oracle Linux KVM HA serves different HA functions. Oracle Linux Virtualization Manager's high availability is policy-based and must not interfere with inherent Oracle RAC instance placement rules. Specifically, for mission-critical, production deployments, co-locating Oracle Linux KVM guests hosting instances of the same Oracle RAC database on a single Oracle Linux KVM server, is not supported. Any Oracle Linux KVM guest failover must respect this fundamental Oracle RAC instance placement rule. For this reason, it's always suggested to configure Oracle Linux Virtualization Manager Affinity Groups. Affinity groups specify that certain virtual machines should never run on the same Oracle Linux KVM server. One affinity group applies to all the Oracle Linux servers in a KVM cluster.

Possible HA combinations include:

- Oracle Real Application Clusters HA and scalability
    - Without additional Oracle Linux KVM guest HA
    - In conjunction with Oracle Linux KVM guest HA, where Oracle RAC instance placement policies are respected
- Oracle Clusterware (failover cluster) provided HA
    - Without additional Oracle Linux KVM guest HA
    - In conjunction with Oracle Linux KVM guest HA, where Oracle RAC instance placement policies are respected
- Standalone Oracle Linux KVM guest HA (single-instance database or Oracle RAC One Node database)

Production environments must be based on the solutions listed above.

✓ **For Oracle Clusterware-based solutions regardless of Oracle Linux KVM, the current support status can be found in Oracle Metalink Note 790189.1 - Oracle Clusterware and Application Failover Management. More information can be found on http://www.oracle.com/goto/clusterware.**

Live migration and online database relocation support

Additional solutions that improve business continuity and availability for Oracle RAC in an Oracle Linux KVM environment include Oracle Linux KVM *Live Migration* and Oracle RAC One Node *Online Database Relocation*.

Oracle Linux KVM *Live Migration* moves a virtual machine from one physical node to another, within the same cluster of servers. Oracle RAC One Node *Online Database Relocation* moves an Oracle Database instance from one server to another within the same cluster. On Oracle Linux KVM environments, these servers are virtual machines, which host an Oracle Clusterware based cluster.

Oracle supports mixed cluster environments, in which some cluster members are based on virtual machines while others are based on physical servers, for migration purposes (going from bare metal to Oracle Linux KVM or vice versa). Oracle RAC One Node Online Database Relocation can therefore be used to migrate between physical and virtual environments based on Oracle Linux KVM. Mixed cluster environments are also supported with Oracle Flex Cluster where "Leaf Nodes" are VMs running on Oracle Linux KVM and "Hub Nodes" are based on bare-metal systems.

However, Oracle Linux KVM *Live Migration* offers a controlled mechanism for manually moving guest virtual machines between physical nodes in the KVM cluster. Since Oracle Linux KVM *Live Migration* allows for the relocation of a guest virtual machine while respecting the placement and CPU allocation rules for Oracle RAC, it can be applied to Oracle Linux KVM guests that host an active Oracle RAC instance or instances.

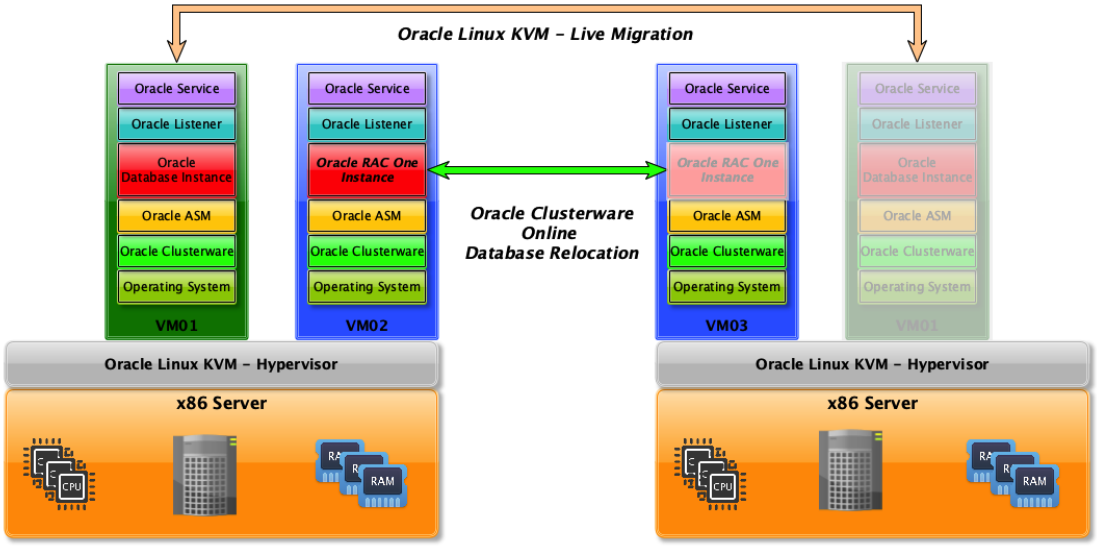


Figure 7: Live Migration and Oracle RAC One Node Database Relocation illustrated

Extended Distance Oracle RAC on Oracle Linux KVM

An Oracle Linux KVM cluster consists of one or more Oracle Linux KVM servers and represents a logical grouping of the servers where a particular set of virtual machines can run. All hosts in a cluster must be of the same CPU type (Intel or AMD). It is recommended that you create your hosts before you create your cluster to help ensure CPU type optimization.

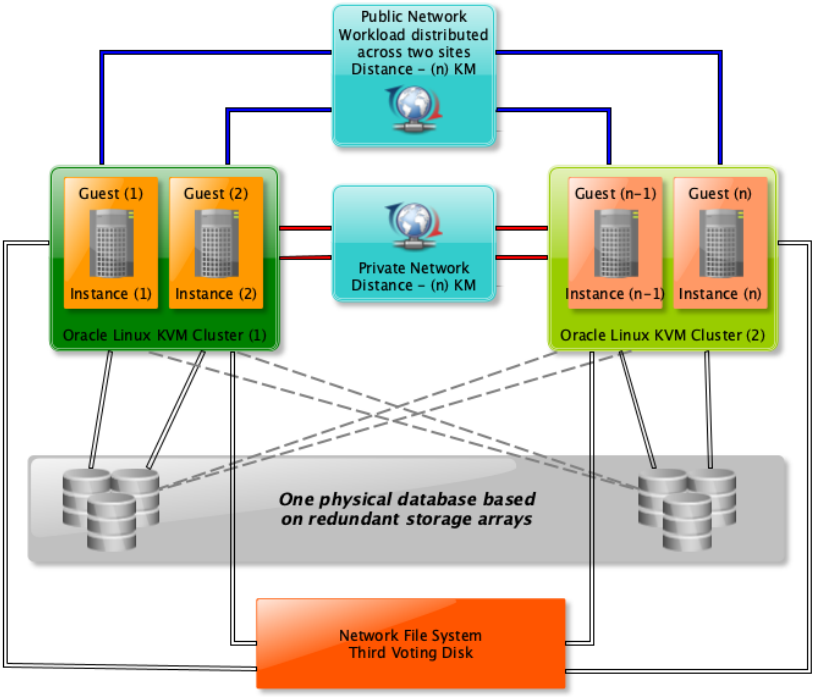


Figure 8: Oracle Extended RAC on Oracle Linux KVM - architecture overview

Stretching an Oracle Linux KVM server cluster across geographical locations is a supported option. Storage and network latencies have to be properly evaluated while implementing this kind of architecture.

Extended Distance Oracle RAC can be deployed on top of one or more Oracle Linux KVM clusters, one cluster stretched on both sites and one cluster for each site (see Figure 7 above).

On the two optional architectures, Oracle Linux Virtualization Manager installation/configuration can have two different options:

Full stretched architecture

In a full stretched architecture, Oracle Linux Virtualization Manager will be installed as a self-hosted engine and configured as a virtual machine running on the stretched Oracle Linux KVM cluster. Both the local and remote Oracle Linux KVM servers, part of the same KVM cluster, will be managed by one Oracle Linux Virtualization Manager that can run as a virtual machine on any Oracle Linux KVM server that is part of the stretched KVM cluster.

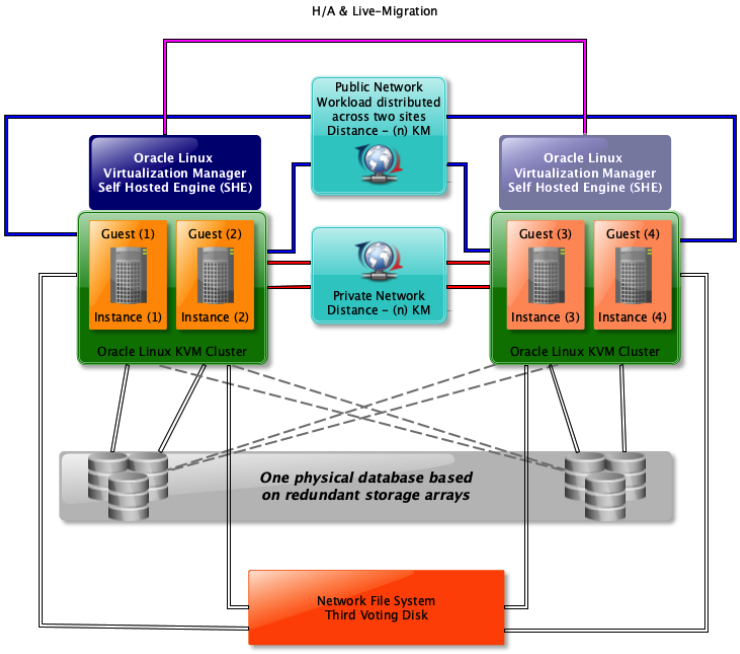


Figure 9: Oracle Extended RAC on Oracle Linux KVM – Single Oracle Linux Virtualization Manager

Stretched database layer

With a stretched database layer, Oracle Linux Virtualization Manager will be installed as a self-hosted engine on each cluster configured in each data center. On top of Oracle Linux Virtualization Manager installations, Oracle Enterprise Manager Cloud Control 13c Release 4 can act as one unique point of monitoring for the entire architecture.

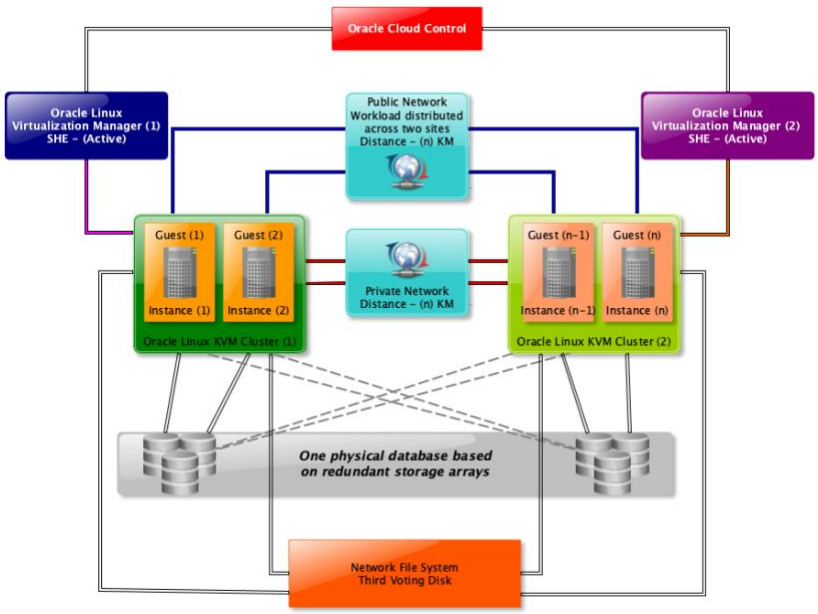


Figure 10: Oracle Extended RAC on Oracle Linux KVM – Oracle Enterprise Manager Cloud Control

When to use *Live Migration* or Oracle RAC One Node *Database Relocation*?

Oracle Linux KVM live migration and Oracle RAC One Node, together, offer complimentary solutions which support continuous, non-interrupted HA.

Oracle Linux KVM live migration offers OS-level migration for virtual machines to facilitate workload management, fault management, and system maintenance operations. For example, live migration permits server level maintenance while not losing access to the Oracle RAC database VM. VM Live Migration consists of iteratively pre-copying the contents of a source virtual machine's memory from one physical host to another target virtual machine on a remote host with no interruption to service.

Oracle RAC One Node and *Online Database Relocation* offer instance-level migrations between properly configured VMs, to facilitate maintenance and workload management within VMs. Using Oracle RAC One Node *Online Database Relocation*, an Oracle RAC One Node instance can be relocated to another server, if the current server is resource constrained or requires maintenance operations such as OS patches.

Online Database Relocation will also migrate Oracle RAC One Node databases to target VMs to facilitate database maintenance on the source VM without interruption of service to other database instances that may be running in the source or target VM. A typical example for such an operation would be a rolling patch upgrade for a database.

Deployment methods

A good example for the universal use and applicability of virtualized solutions, based on Oracle Linux KVM, are Oracle Engineered Systems like Virtualized Exadata Database Machine or Oracle Database Appliance.

> ✓ **Oracle Linux KVM is the ideal server infrastructure when used in virtual clusters for development, test, demo, or educational environments, based on only one Oracle Linux KVM host.**

Easy to re-install by re-deploying a formerly saved base image or template, once the current environment is 'worn out' by various test runs. Oracle Linux KVM provides these benefits as part of the virtual cluster creation and the rapid provisioning approach, in conjunction with Oracle RAC, as described in the previous section.

Figure 10 shows a typical Oracle RAC deployment for a development or test environment based on Oracle Linux KVM. F**or virtual test or development clusters,** Oracle allows the two Oracle Linux KVM virtual machines, each hosting one Oracle RAC database instance, to run on only one Oracle Linux KVM host.

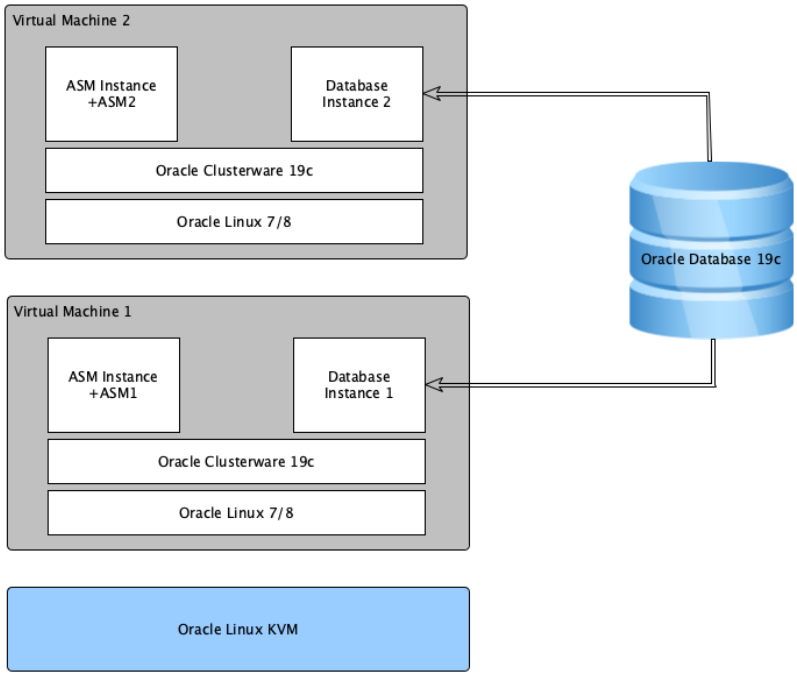


Figure 11: Oracle RAC on Oracle Linux KVM: Development deployment example

Based on certification tests, Oracle has **certified Oracle RAC on Oracle Linux VKM** for **production environments** This is the only certified x86 server virtualization technology for Oracle Real Application Clusters on Linux and thereby enables these technologies to be used beyond the scope of pure development or test environments.

> ✓ **Oracle Linux KVM together with Oracle RAC enable the use of fully virtualized environments as the basis for server consolidation and highly available applications. In order to avoid a single point of failure, a minimum of two Oracle Linux KVM servers should be used.**

Unlike test and development environments, typical production environments use more than one Oracle Linux KVM server for each Oracle Linux KVM virtual machine (minimum two), each, again, hosting an Oracle RAC database instance, as illustrated in Figure 11. These configurations are mostly used in server consolidation environments.

Having more than one Oracle Linux KVM server as the underlying hardware platform to host the virtual machines with the Oracle RAC database instances eliminates the host hardware from becoming the single point of failure.

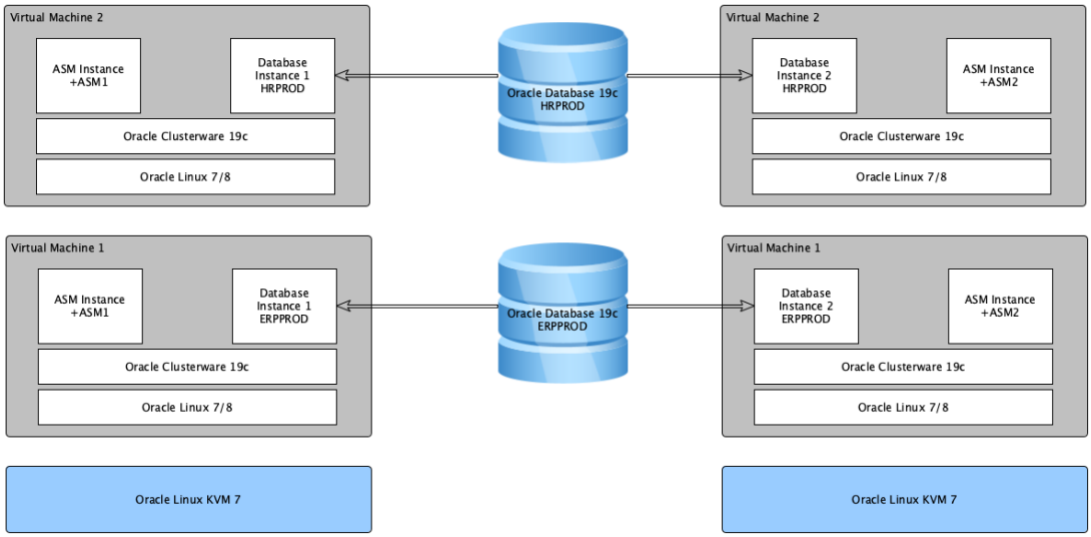


Figure 12: Oracle RAC on Oracle Linux KVM: Production deployment example

Hardware and software requirements

Oracle has certified Oracle RAC on Oracle Linux KVM environments based on the following hardware and software requirements. Configurations that do not fulfill these requirements are currently not supported.

Note: As a minimum, the hardware and software requirements listed in the Oracle RAC documentation must be met to deploy Oracle RAC on Oracle Linux KVM in production environments.[8]

Note: Oracle is constantly in the process of testing additional and advanced Oracle Linux KVM features with Oracle RAC. This paper will be updated as new test results are available.

Hardware requirements for production environments

A minimum of two Oracle Linux KVM servers are strongly recommended

- Each host must provide a minimum of two physical Ethernet NICs for Oracle Linux Virtualization Manager "Management Network," two physical Ethernet NICs for Oracle RAC public communication, and two physical Ethernet NICs for Oracle RAC private (interconnect) communication. A minimum of six physical Ethernet NICs is required, or a minimum of eight physical Ethernet NICs if network based storage connectivity is used.
- Using two physical Ethernet 10GbE NICs (instead of 6/8 suggested above) for all roles ("Management Network – OLVM," "Public Network – RAC," "Private Network – RAC") is a supported option only when an external solution is able to create vNICs (minimum three for each NIC) with dedicated bandwidth and QoS enabled.
- Each pair of Ethernet NICs will be configured in H/A by bond driver on Oracle Linux KVM server unless for Oracle RAC Private Interconnect that can rely on Oracle RAC Redundant Interconnect Usage or HAIP.[9]
- 10GbE or higher network interfaces[10] are supported and it is strongly recommended that these NICs support IEEE Data Center Bridging if server VLAN tagging is to be configured.
- For the Oracle Clusterware interconnect traffic or ASM storage traffic 10GbE is required.

[8] Support questions: How to use Oracle RAC in a Cloud Solution

[9] Oracle® Clusterware Administration and Deployment Guide 19c Release

- For the Oracle Clusterware interconnect network communication, it is suggested that Jumbo Frames (MTU=9000) be implemented on NICs that will be dedicated to Oracle RAC private communication; the same change will also have to be applied and verified within virtual machine configuration (/etc/sysconfig/network-scripts).
- Storage access redundancy requires a minimum of two HBAs/SCSI controllers or the same level of redundancy for iSCSI/NFS storage-based solutions.
- Storage access timeout reacts to given limits on an Oracle Linux KVM cluster by enabling and configuring proper "Fencing Policies."

Hardware requirements for development environments

For development or non-production environments on a single Oracle Linux KVM server, external storage is recommended, but not required.

- Virtual network connectivity can be established within the Oracle Linux KVM server, if no external client connections are required.
- Oracle recommends using identical hardware configurations for development and pre-production QA environments, accurately modeling production environments.

Oracle RAC software

- Oracle RAC and Oracle Clusterware **19c and 21c** in 64-bit software versions are currently supported. Oracle recommends these releases always be kept updated with latest CPU (Critical Patch Update[11]) and PSU (Patch Set Update[12]).

General information about Oracle Linux KVM and Oracle Linux Virtualization Manager

Oracle Linux KVM is based on Oracle Linux 8 Update 9 (or higher update). Maintaining the most up-to-date software is strongly suggested.

Oracle Linux Virtualization Manager 4.5 is required. Oracle Linux KVM running on Oracle Linux 8 is required.

Guest virtual machines running Oracle Database software can be based on Oracle Linux 8 and Oracle Linux 9.

When using Oracle RAC:

- Dynamic vCPU and memory changes are not supported.
- Pause/Unpause Oracle RAC virtual machine is not supported.
- Live migration of an Oracle RAC VM is not supported with Oracle Linux KVM releases older than 8 Update 9

Oracle Linux KVM guest configuration

- The guest must be configured with VirtIO/VirtIO-SCSI controller types, independently if the database disks are physical or virtual
- Oracle Linux 7, Oracle Linux 8 and Oracle Linux 9 are supported
- 64-bit Linux is currently supported for the Oracle Linux KVM guest for Oracle Database

[11] Critical Patch Updates, Security Alerts and Third Party Bulletin
[12] Oracle Recommended Patches - Oracle Database (Doc ID 756671.1)

Oracle Linux KVM vCPU configuration for Oracle RAC

Testing has shown that a small amount of incidental over-committing of physical CPUs will not greatly diminish the overall stability of the system or the cluster stack where the competing workloads for oversubscribed resources are well understood and concurrent demand does not exceed physical capacity. It is still recommended, however, not **to over-commit CPUs**. Multicore CPUs will treat every core as a virtual CPU (vCPU) and multicore CPUs with multithreaded support **will treat each thread** as a vCPU. **vCPU allocation** in Oracle RAC / Oracle Linux KVM environments **must therefore adhere to the following rules**:

- **VCPU =** vCPU calculations are based on physical cores and **not** threads. For CPUs that are not multithread, "1 vCPU=1 core."
- If hyperthreading is enabled on the bare metal system, "1 vCPU = 2 threads."
- The total number of vCPUs allocated to guest domains, should not exceed two times (2x) the number of real cores in the Oracle Linux KVM server.
- The number of vCPUs allocated to a single guest domain should not exceed the number of real cores / threads in the Oracle Linux KVM server.
- CPU pinning is recommended for hard partitioning but also for NUMA awareness.

vCPU allocation examples

For the following examples you can also consider "CPU QoS – Quality of Service" options available with Oracle Linux Virtualization Manager.

CPU quality of service defines the maximum amount of processing capability a virtual machine can access on the host on which it runs, expressed as a percent of the total processing capability available to that host. Assigning CPU quality of service to a virtual machine allows you to prevent the workload on one virtual machine in a cluster from affecting the processing resources available to other virtual machines in that cluster.

Valid configuration examples

- A server with 2 Socket(s), 4 core(s) each (8 cores total), 16 threads with Hyper Threading enabled.

VIRTUAL MACHINE	VCPUS ALLOCATED
Guest1	6 vCPUs (6 threads, 3 cores)
Guest2	6 vCPUs (6 threads, 3 cores)
Guest3	4 vCPUs (6 threads, 2 cores)

✓ This is a valid configuration because the total number of vCPUs allocated to guest domains does not exceed two times (2x) the number of real cores in the Oracle Linux KVM server.

✓ Total Guest vCPU = 16 (two times the number of cores available)

- A server with 4 Sockets, 10 cores each (40 cores total), 80 threads with Hyper Threading enabled.

VIRTUAL MACHINE	VCPUS ALLOCATED
Guest1	20 vCPUs (20 threads, 10 cores)
Guest2	20 vCPUs (20 threads, 10 cores)

Guest3	20 vCPUs (20 threads, 10 cores)

- ✓ This is a valid configuration because the total number of vCPUs allocated to guest domains does not exceed two times (2x) the number of real cores in the Oracle Linux KVM server.
- ✓ Total Guest vCPU = 60 (less than two times the number of cores available)

Invalid configuration examples

- A server with 2 Sockets, 4 cores each (8 cores total), 16 threads with Hyper Threading enabled.

VIRTUAL MACHINE	VCPUS ALLOCATED
Guest1	12 vCPUs (12 threads, 6 cores)
Guest2	12 vCPUs (12 threads, 6 cores)

- ✓ This is an **invalid** configuration because the total number of vCPUs allocated to guest domains exceeds two times (2x) the number of real cores in the Oracle Linux KVM server.
- ✓ Total Guest vCPU = 24 (more than two times the number of cores available)

- A server with 4 Sockets, 10 cores each (40 cores total), 80 threads with Hyper Threading enabled.

VIRTUAL MACHINE	VCPUS ALLOCATED
Guest1	40 vCPUs (20 threads, 10 cores)
Guest2	40 vCPUs (20 threads, 10 cores)
Guest3	20 vCPUs (20 threads, 10 cores)

- ✓ This is an **invalid** configuration because the total number of vCPUs allocated to guest domains exceeds two times (2x) the number of real cores in the Oracle Linux KVM server.
- ✓ Total Guest vCPU = 100 (more than two times the number of cores available)

Oracle Linux KVM vRAM configuration for Oracle RAC

Memory over-commit, a feature supported by Oracle Linux KVM and Oracle Linux Virtualization Manager, is not supported for guest virtual machines running Oracle Database or Oracle Real Application Clusters.

Oracle Linux KVM supported storage configurations for Oracle RAC

With Oracle Linux KVM you can configure storage in different ways. Oracle RAC and Oracle Clusterware are not supported on all possible storage configurations that Oracle Linux KVM offers.

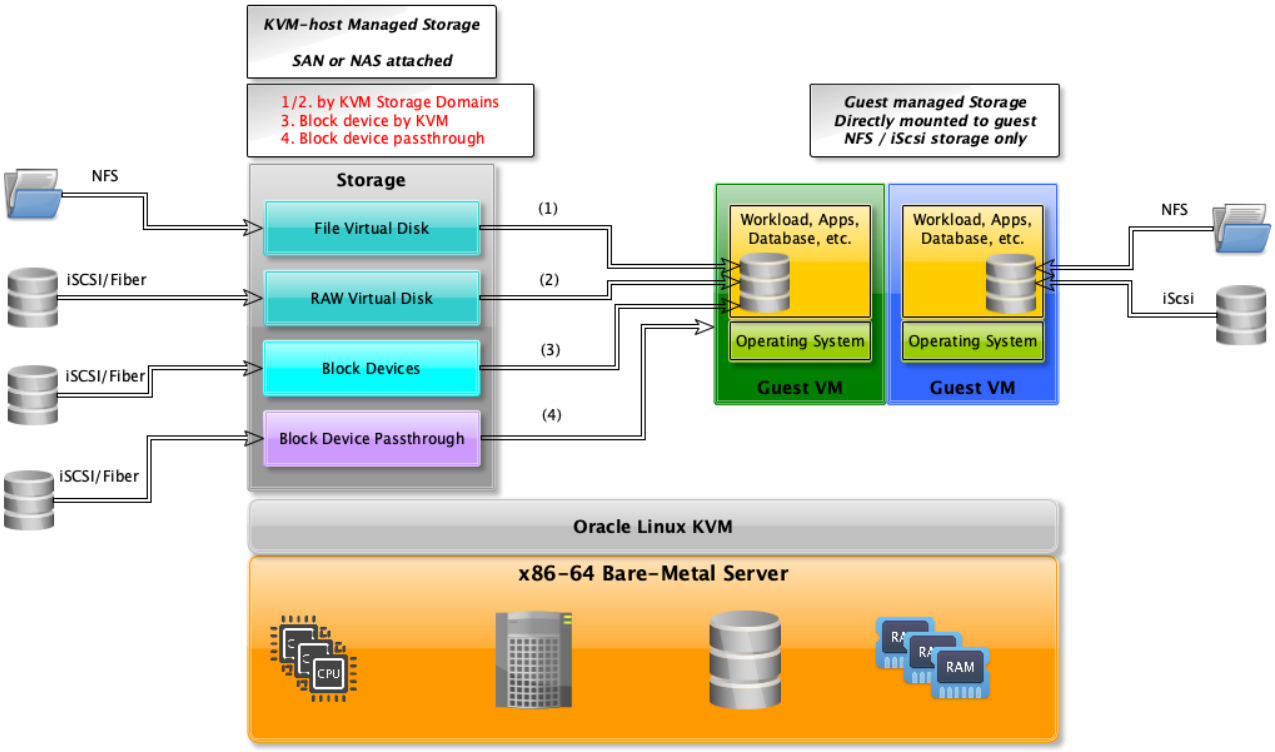


Figure 13: Storage configuration options on Oracle Linux KVM

Generally, storage configurations can be divided into two groups:

- KVM host-managed storage
- Guest-managed storage

KVM host-managed storage

The KVM host-managed method of storage configuration, allows the storage to be made available in the Oracle Linux KVM server and then made available in the guest virtual machine using the guest's configuration file. The storage available in Oracle Linux KVM server can be configured as a "storage domain" or presented by an Oracle Linux Virtualization Manager interface, to the guest virtual machine.

Guest-managed storage

In this method of storage configuration, the storage is made available directly in the guest domain. It is not visible in dom-0. Options for this method are iSCSI and NFS.

STORAGE OPTION	STORAGE OPTION	SUPPORTED WITH ORACLE RAC
KVM Host Managed	FC block device (virtual disk on storage domain)	Yes
	iSCSI block device (virtual disk on storage domain)	Yes
	NFS (virtual disk on storage domain)	Yes
	FC block device - physical LUN mapped to the guest virtual machine	Yes (Passthrough not suggested)
	iSCSI block device - physical disk mapped to the guest	Yes (Passthrough not suggested)
	Gluster block device (virtual disk on storage domain)	No (see Note below)
	Gluster (virtual disk on storage domain)	No (see Note below)
Guest Managed	iSCSI	Yes (Recommended)
	NFS	Yes (Recommended)

Table 1: Storage options supported with Oracle Real Application Clusters

- ✓ **Oracle Supports Oracle RAC for production environments, built on Oracle Linux KVM, with both physical and virtual disks for Oracle Cluster Registry, voting, and database files. Virtual-disk performance should be acceptable if compared to physical disk performance on the same system or architecture.**
- ✓ **Each Oracle Linux KVM server (physical-server) is limited to one Oracle Database Hub-Node or one Oracle Database Node guest and must be part of the same Oracle Real Application Clusters.**
- ✓ **Building Oracle RAC with more virtual machines on a single Oracle Linux KVM server (physical server) is allowed only for demo or educational purposes.**

ORACLE

Oracle RAC on Oracle Linux KVM – Best practices

Based on certification testing of Oracle RAC for Oracle Linux KVM, below are some best practices and recommendations that Oracle has developed for running Oracle RAC in an optimized virtual environment.

Oracle installation requirements and recommendations

Linux Kernel parameters for Oracle Linux KVM guest environment

Set the kernel parameter vm.min_free_kbytes=51200 (not required for Unbreakable Enterprise Kernel for Oracle Linux) which will reserve 512MB to allow the OS to reclaim memory faster and avoid LowMem pressure. See the following My Oracle Support documents for additional information:

- Oracle Linux: Recommended Value of vm.min_free_kbytes Kernel Tuning Parameter (Doc ID 2501269.1)
- Master Note of Linux OS Requirements for Database Server (Doc ID 851598.1)

To setup your Oracle Linux system you can use the "oracle-database-preinstall" RPM, based on the Oracle Database release that will be installed. This RPM will install all the required dependencies and configure all the kernel parameters required to properly guest a single instance Oracle Database, or an Oracle RAC instance.

Time synchronization

Guest virtual machines without accurate time keeping may experience issues with network applications and processes, as session validity, migration, and other network activities rely on timestamps to remain correct.

KVM avoids these issues by providing guest virtual machines with a para-virtualized clock (kvm-clock).

This time synchronization can allow some time drift in the guest domains running Oracle RAC. Therefore, the following recommendations should be implemented when running Oracle RAC on Oracle Linux KVM:

- Cluster Time Synchronization Daemon (CTSSD) can be used in place of NTPD. CTSSD will synchronize time with a reference node in the cluster when an NTPD is not found to be configured. Should you require synchronization from an external time source, you must use NTPD, which will cause CTSSD to run in "observer" mode. However, if NTPD is running, then it must to be configured with the slewing option as documented in Document 551704.1.
- Use an externally configured NTPD as the server for your VMs.

Storage configuration

Oracle Linux KVM guest limits

Oracle RAC deployments on Oracle Linux KVM should also consider maximum-limits[13] imposed by the solution itself.

To design the best solution, you should consider database size, hypothetical growth of the same and, based on this, define a good balance between the number of disks and their size.

If the storage used is guest-managed (NFS shares or iSCSI devices directly presented to the VM), configuration limits for Oracle Linux KVM, mostly related to the number of disk devices, can be safely ignored. Best practices, valid for a bare metal system are also valid for this type of configuration on top of Oracle Linux KVM.

Multipath and device persistency

For Oracle RAC on Oracle Linux KVM environments, multipath access to the (SAN) storage is highly recommended. Multipath is, as a default, already configured in Oracle Linux KVM and not in the guest virtual machines.

[13] Oracle Linux Virtualization Manager: Scalability limits

Oracle Linux KVM server, relying on storage WWID of devices presented, grants device persistency. There is no need to set up device persistency in the guest VMs. The mapping of Oracle Linux KVM disks to guest disks can be dynamically managed using Oracle Linux Virtualization Manager or API (see Appendix A for an example).

Oracle ASM is Oracle's recommended storage management solution that provides an alternative to conventional volume managers, file systems, and raw devices.

The Oracle ASM volume manager functionality provides flexible server-based mirroring options. The Oracle ASM normal and high redundancy disk groups enable two-way and three-way mirroring respectively. You can use external redundancy to enable a Redundant Array of Independent Disks (RAID) storage subsystem to perform the mirroring protection function.

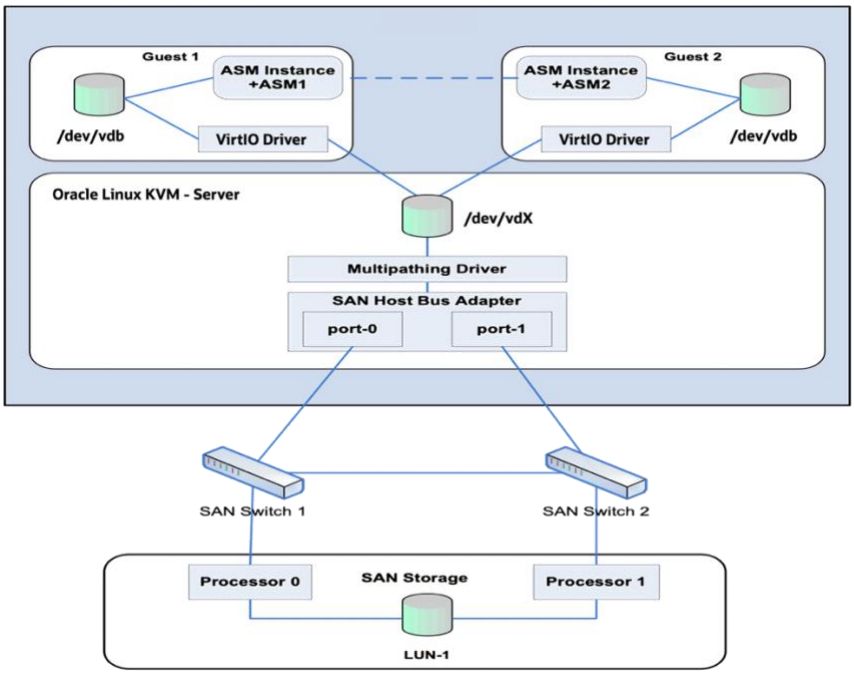
Oracle ASM Filter Driver (Oracle ASMFD) is available as a storage option for guest VMs.

Oracle ASMFD is a kernel module that resides in the I/O path of the Oracle ASM disks. Oracle ASM uses the filter driver to validate write I/O requests to Oracle ASM disks.

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.

Oracle ASM Filter Driver rejects any I/O requests that are invalid. This action eliminates accidental overwrites of Oracle ASM disks that would cause corruption in the disks and files within the disk group. For example, the Oracle ASM Filter Driver filters out all non-Oracle I/Os which could cause accidental overwrites (for example dd if=/dev/zero of=/dev/oracledisk will simply fail when ASMFD is used).

If you have an existing Oracle ASM library driver (Oracle ASMLIB) configuration, then depending on whether you want to use Oracle ASMLIB or Oracle ASMFD, make the necessary changes based on the Oracle Database documentation.[14]



[14] Installing and Configuring Oracle ASMLIB Software

Figure 14: SAN storage on Oracle Linux KVM

Cluster file system requirements

Some applications require running the Oracle RAC database on a cluster file system or relying on some files that need to be available on every node. For these applications, the Oracle ASM Cluster File System (ACFS) should be used in the guest virtual machines.

> ✓ **For additional best practices, read the My Oracle Support KM document:**
>
> ✓ ***Oracle RAC and Oracle Clusterware Best Practices and Starter Kit (Linux) (Doc ID 811306.1)***

Network configuration

Required networks

The Oracle Linux KVM server should have an adequate sufficient number of NICs to provide at least two distinct bridges and virtual NICs to the guest virtual machines. For production systems, the number of NICs per Oracle Linux KVM server is specified in the "Hardware and software requirements" section of this paper. The actual number of NICs depends on the overall configuration and level of redundancy desired.

For other Oracle RAC configurations, a separation for:

- Oracle Linux KVM management network
- Public network
- Private network (cluster interconnect)
- Storage network (if applicable)

> ✓ **In a virtualized environment, like Oracle Linux KVM, network components may be virtualized. As a result, sometimes network failures won't be reflected to guest VMs, even if the hardware network card fails, the public network virtual NIC, available within the guest, remains up. A "Ping Targets"[15] enhancement overcomes this problem.**

is required for production environments. Therefore, the Oracle Linux KVM server must ensure that the respective network communication does not interfere with the others.

These networks, however, can be shared with multiple Oracle RAC VM guests in an Oracle Linux KVM server. Consolidation of network traffic is only supported for 'like' traffic, e.g. private interconnect traffic may be consolidated with private interconnect traffic from other Oracle Linux KVM guests, public traffic may be consolidated with public traffic from other Oracle Linux KVM guests, where both traffic types are arbitrated through their respective virtual bridge. Oracle RAC public and private traffic must be physically and logically separate. When consolidating networks, latency, bandwidth, and capacity must be carefully measured and monitored (see "Sizing the Oracle Linux KVM environment for Oracle RAC" section of this document for recommendations).

The Oracle Linux KVM management network can be shared with Oracle RAC public network. Using the Oracle Linux Virtualization Manager host and guest network QoS can also help to address possible physical NIC sharing.

Network bonding

For production environments, it is strongly recommended to have two Network Interface Controllers (NICs) for each network and use Linux bonding configured in Oracle Linux KVM to make them highly available. This results in an

[15] 12.1.0.2 Ping Target for Oracle Clusterware VIP (Doc ID 1958241.1)

Oracle RAC network HA requirement of a minimum of four NICs (for public and private network) per Oracle Linux KVM server running Oracle RAC, as illustrated in Figure 14.

Appendix B explains setting up bonding devices step-by-step.

For the Oracle Clusterware private interconnect, in place of Linux bonding in Oracle Linux KVM, you may present two to four vNICs for the private interconnect to each of the configured Oracle Linux KVM guest virtual machines. Each vNIC would map to a distinct physical network interface, either on a single subnet or multiple subnets for Oracle Clusterware interconnect high availability. This feature, Redundant Interconnect Usage[16] or HAIP is a failover model with multiple communication endpoints. It is managed by the Oracle Clusterware in each of the Oracle Linux KVM guests and uses automatically assigned, non-routed link local addresses for the Oracle Clusterware private interconnect communication. An acceptable configuration would be a combination of Linux bonding of the clusterware interconnects in Oracle Linux KVM with HAIP configured in the Oracle Linux KVM guest virtual machines.

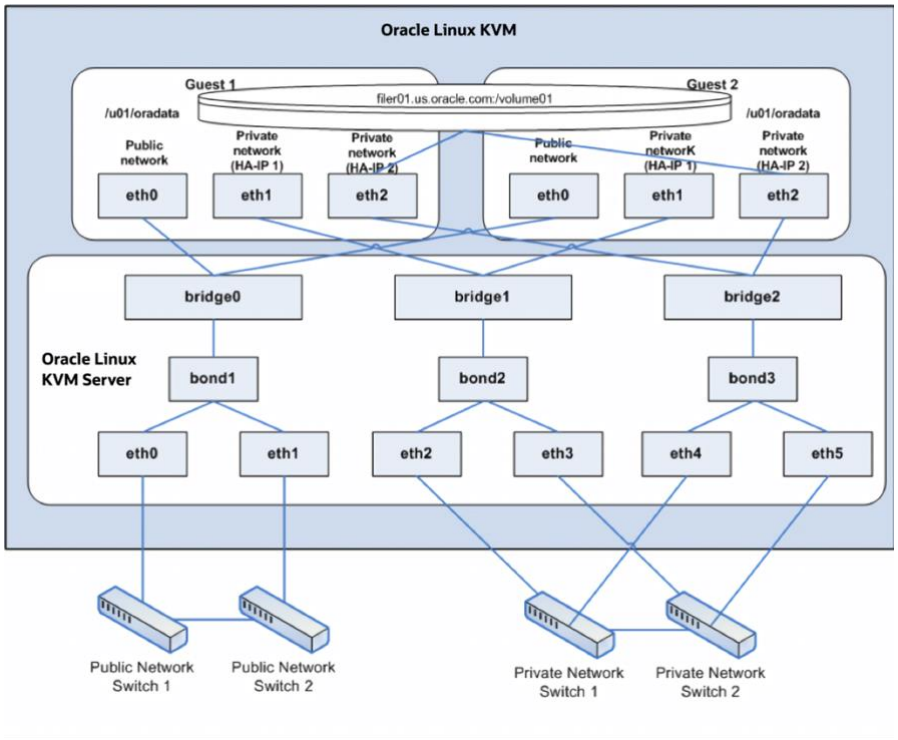


Figure 15: Full network bonding on Oracle Linux KVM

HugePages and High-Performance Virtual Machines

Huge Pages is a feature part of the Linux kernel. Enabling Huge Pages makes it possible for the operating system to support large memory pages. Using Huge Pages can improve system performance by reducing the amount of system CPU and memory resources required to manage Linux page tables, which store the mapping between virtual and physical memory addresses. For Oracle Databases, using Huge Pages can drastically reduce the number of page table entries associated with the System Global Area (SGA).

"High-Performance" optimization, available by OLVM web interface, allows to get the virtual machine configured with a set of automatic, and recommended manual, settings for maximum efficiency.

[16] Oracle® Clusterware Administration and Deployment Guide 19c Release

For further details related to HugePages support and “High-Performance” virtual machines, please refer to Oracle Linux Virtualization Manager Administration Guide.

Orachk utility

Oracle provides the ORAchk tool to audit the configuration of Oracle RAC, Cluster Ready Service, ASM, Grid Infrastructure etc. It is it a useful starting point for most analysis. Further information is available at:

- ORAchk - Health Checks for the Oracle Stack (Doc ID 1268927.2)
- Introduction to Oracle ORAchk and Oracle eXAchk

Sizing the Oracle Linux KVM environment for Oracle RAC

As a general rule of thumb, you should “size your Oracle Linux KVM environment in a similar way as you would size a physical server based environment. Particular attention should be paid to the following parameters:

CPUs and cores

Follow the requirements as stated in this document for assigning vCPUs to guest virtual machines. It is supported, but not recommended, to over-commit by a factor of two.

If multiple databases have been configured into a single Oracle RAC virtual machine, please evaluate Oracle Database Resource Manager configurations and “Instance Caging” best practices.

Memory

Oracle Linux KVM allows memory between virtual machines to be shared. This feature is not supported for virtual machines running Oracle products, such as Oracle Database or Oracle Real Application Clusters.

When sizing memory for your Oracle RAC VM system, make sure your system memory is large enough to accommodate the memory needed for the Oracle Linux KVM host, the operating system running on the virtual machine, Oracle Clusterware, and the Oracle RAC databases in each of the guest virtual machines.

For the Oracle Linux KVM host, a minimum of two logical CPUs and 1 GB of memory per logical CPU is recommended.

Network

For an Oracle RAC on Oracle Linux KVM environment, the general recommendation is to install a minimum of two NICs per KVM server machine. For a production environment, the recommendation is to have at least four NICs (preferably 10G/bit or higher) installed in the Oracle Linux KVM server and use bonding. For demo purposes, the environment can be built on a single physical NIC.

Keep in mind that in a shared network bridge, bandwidth is shared between the different guest virtual machines installed on one Oracle Linux KVM server. In production environments, these network interfaces must be at a minimum 1 Gbit Ethernet, however, 10Gbit Ethernet or higher is fully supported and recommended. As noted earlier, similar network traffic may be consolidated but the physical and logical segregation between public/private/management and storage networks is required. That would mean dedicated 10GbE cards for each traffic type where consolidated traffic on this NIC may be VLAN tagged for logical separation. This configuration should be performed in Oracle Linux KVM through the Oracle Linux Virtualization Manager.

Oracle Linux Virtualization Manager also allows to define QoS for Virtual Machine Networking and this feature can be leveraged to prioritize desired network traffic.

When configuring a larger number of guest VMs on one host, or when deploying guest VMs that require a high amount of network bandwidth, conscientious monitoring for oversubscription events resulting in packet loss, either on the NIC or the switch, and/or increased latencies and bandwidth degradation must be implemented.

At the same time, features like network QoS, available in Oracle Linux Virtualization Manager, can be leveraged to properly setup the architecture and grant higher bandwidth and lower latency to production environments.

Storage

When using SAN storage for your Oracle RAC on Oracle Linux KVM deployment, more than one HBA can be used to obtain a higher bandwidth to the storage in the Oracle Linux KVM host. The multipath driver usually combines the bandwidth of each path automatically.

A dedicated network interface is recommended when using NAS or iSCSI storage.

The dedicated storage network should be configured in the same way as the public or interconnect network. Again, multiple NICs can be used to eliminate a single NIC as a single-point-of-failure and appropriate bonding modes can be used to increase bandwidth.

Recommended installation: Oracle Templates for Oracle Linux KVM

Oracle Database Templates for Oracle Linux KVM provide an easy and fast way of deploying any one of three configurations: Oracle single instance, Oracle Grid Infrastructure for Standalone Server (formerly known as Oracle Restart), and Oracle RAC cluster on Oracle Linux KVM **for test** as well as **production environments**.

Oracle Database Templates for Oracle Linux KVM are built with cloud-init technology to allow automated Oracle Database deployment.
The entire installation is automated, requesting minimal information (e.g. node names, IP addresses, etc.) from the user. The final n-node cluster (or Single Instance Database) configuration adheres to the best practices for production systems as described in this paper.

Oracle Database Templates for Oracle Linux KVM uses an updated Oracle Linux 64-bit system release, which can also be updated in real time at deployment (see Yum@Deploy support). The OS image contains a minimal installation of Oracle Linux, only basic RPM packages are installed with the use of the preinstall RPM described earlier in this paper.

Configuration and benefits at a glance:

- Support for three modes of operation: Single Instance, Grid Infrastructure for Standalone Server (Oracle Restart) and Oracle RAC.
- Templates are available for both Oracle Database Enterprise Edition as well as Standard Edition.
- Templates are released as fully operational “kits” with OS and Oracle Database software disks, however one can “mix and match” thus creating many possible combinations for deployment.
- Full support for the majority of Oracle Database and Oracle RAC features like Oracle Flex Cluster and Flex ASM, pluggable databases, container databases, Admin Managed or Policy Managed databases, automated ACFS Filesystem creation including Database in ACFS; support for dedicated ASM network interface and much more.
- Allows OS real-time update at deployment, see Yum@Deploy support in the Template’s documentation.
- The environment comes loaded with Swingbench, OS Watcher, and more tools.
- The exact same template or VM, created from the template, can be repurposed to become single instance or a member of an Oracle RAC Cluster via a simple cleanup and rebuild of that VM.

To download and use the templates and the Deploycluster tool to configure an Oracle Database on Oracle Linux KVM in **test** or **production** systems see:

https://www.oracle.com/database/technologies/rac/vm-db-templates.html

Oracle RAC on Oracle Linux KVM environments: Manual installation

The following steps describe the complete installation of Oracle RAC in an Oracle Linux KVM environment:

- Plan your installation carefully. Consider the requirements for your new environment and plan accordingly using the guidelines and best practices in this white paper and the official Oracle documentation.

Install Oracle Linux KVM server binaries on the machine(s) on which you want to perform the Oracle RAC installation. Follow the Oracle Linux KVM and Oracle Linux Virtualization Manager Documentation[17] for detailed instructions. Then continue with these steps:

- With Oracle Linux Virtualization Manager, discover the servers and configure storage, networking, and all components related to the Oracle Linux KVM server.
- Create the virtual machines for the Oracle RAC installation using virtual disks for both the operating system and the Oracle Database/Oracle Clusterware software.
- Install and configure the guest operating system (Oracle Linux 7, 8 or 9). Make all necessary configuration changes to the guest operating system, as described in the requirements for installing Oracle Clusterware and Oracle RAC. For Oracle Linux, you also have the option to rely on "preinstall" rpms, dedicated to automate the OS setup for Oracle Database/Oracle Clusterware. See further details:

  Linux OS Installation with Reduced Set of Packages for Running Oracle Database (Doc ID 728346.1)
- Install Oracle Clusterware and Oracle RAC as for non-virtualized environments, following the installation guides for these products.

Live migration and online database relocation best practices

Oracle Linux KVM live migration best practices

Live migration of VMs offers operating system (OS) level migration to facilitate workload management, fault management and system maintenance operations. Oracle Linux KVM live migration consists of iteratively pre-copying the contents of a source VM's memory from one physical host to another target VM on a remote host with minimal interruption to running services.

This iterative pre-copy of the memory continues until it is determined that the dirty page rate on the source machine is consistently higher than the memory transfer rate to the target machine. At this point, the VM is stopped and all remaining dirty pages are copied from the source VM to the target VM.

The stop-and-copy operation to complete the live migration is a suspend time blackout of the VM. For most applications, the dirty page rate will be low and the suspend time will measure in milliseconds to seconds. But for large highly active applications with potential dirty page sets of greater than 2.7GB, the suspend time of the VM could be considerably longer, potentially violating clusterware heartbeat thresholds which would trigger node fencing at the Oracle Clusterware level.

To avoid these potential failures during a live migration of VMs the following best practices should be observed:

- Prior to initiating a live migration in an Oracle RAC production environment, redefine the Oracle Clusterware misscount from the default of 30 to 60 seconds. Issue the following command as root:

  ```
  # crsctl set css misscount 60
  ```

  This will allow a live migration for an application with a potential dirty page set of greater than 2.7GB. Assuming a conservative maximum throughput for a single 1GbE shared link provides 90MB/sec., a 60

[17] Oracle Linux KVM and Oracle Linux Virtualization Manager Documentation

second misscount would allow a suspend period of 5.4GB (90MB/sec * MC) where MC= misscount set to 60 seconds.

- When the live migration is complete, set the Oracle Clusterware misscount back to the default. Issue the following command as root:

  `crsctl unset css misscount`

> ✓ **There is a known issue that live migration may fail when ACFS (ASM Cluster File Systeem) is used, see MOS Note: 2202282.1**

Only one live migration should be executed at a time. Multiple, simultaneous live migrations within the clustered environment may violate the suspend time thresholds described above.

A dedicated, high bandwidth network has to be used for live migrations and the VM guests should be on the same network and IP subnet. The default network used by live migration is the Oracle Linux KVM management network (ovirtmgmt) that may be the public network. If live migration is used as a frequent maintenance operation, it is recommended to deploy bonded interfaces dedicated to this traffic type.

While secure live migration is enabled by default, it should be considered that a secure connection to the remote guest (using –ssl) adds overhead to the live migration operation. It is therefore recommended thatsuch secure connections be avoided, if permitted. In general, a secure connection can be avoided, if the live migration network is inherently secure.

> ✓ **Oracle Linux Virtualization Manager's pause, suspend, and resume management functionality for guest virtual machines must be avoided when the guest virtual machine contains an actively running production Oracle RAC instance. Any pause or suspend operation could trigger Oracle Clusterware to fence the suspended virtual machine or, in worst cases, lead to a block corruption.**

Oracle RAC One Node online database relocation best practices

Best practices for Oracle RAC One Node online database relocation remain the same when performing an online database relocation operation within an Oracle Linux KVM-based environment, as opposed to executing this procedure on physical machines. More information on the Oracle RAC One Node online database relocation can be found in the Oracle RAC One Node User's Guide.[18] When performing an online database relocation between two Oracle Linux KVM guest virtual machines, consider the following:

- The target VM for an online database relocation event must have adequate OS/system resources to allow for an Oracle RAC One Node instance to run.
- Only one online database relocation event should be executed at a time.
- The target VM must be a configured member of the cluster.

[18] 19c Real Application Clusters Installation Guide for Linux and UNIX

Summary

Following the recommendations made in this paper for deploying Oracle RAC on Oracle Linux KVM, and using the standardized technologies described, you can implement configurations such as the one illustrated in fig. 13 or even more sophisticated configurations.

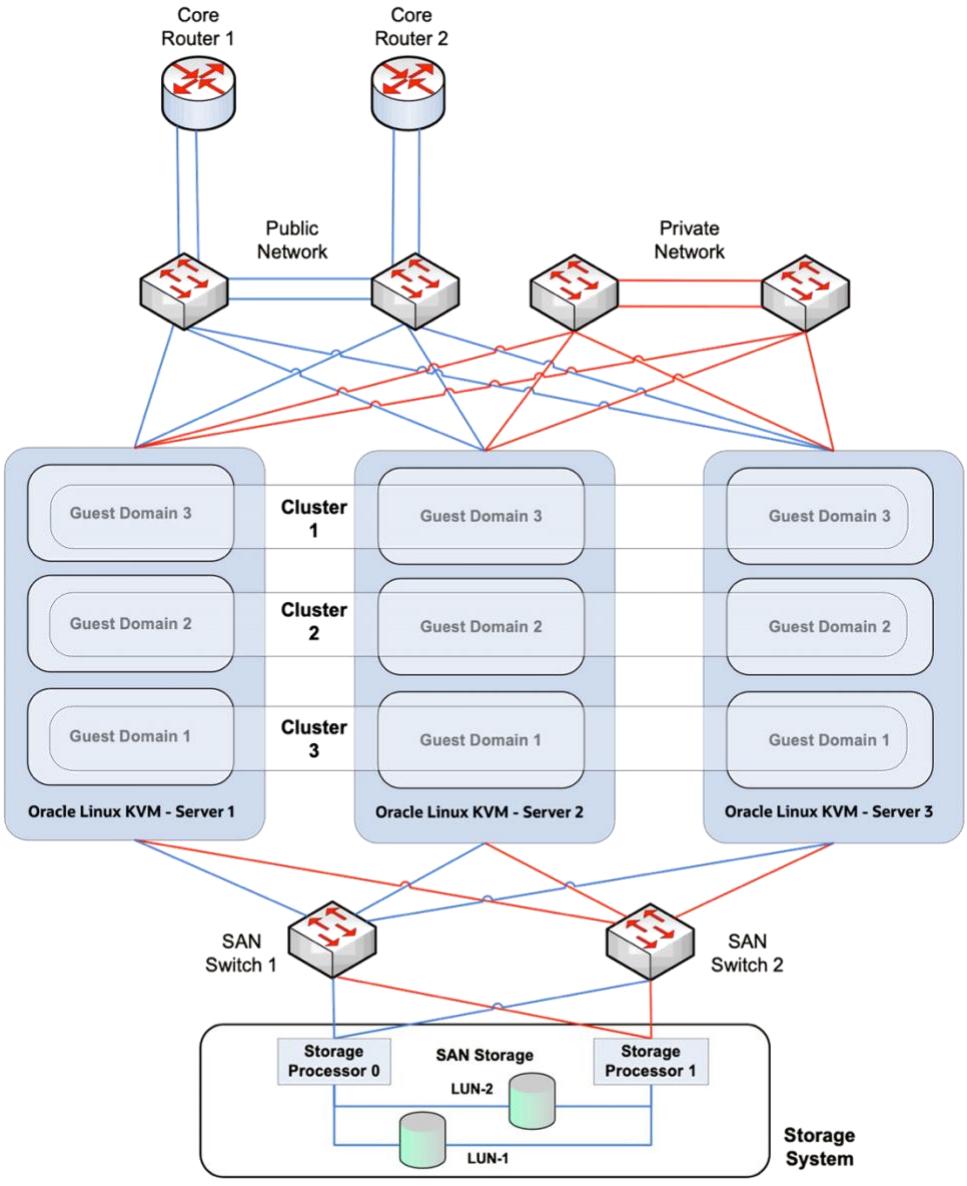
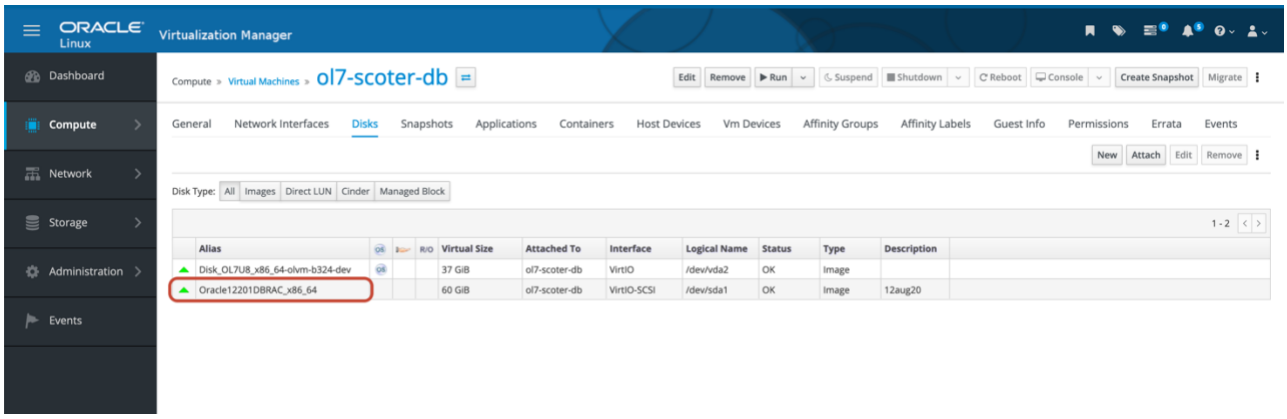


Figure 16: Production architecture example

Appendix A – Disk persistence with Oracle Linux Virtualization Manager

Oracle Linux Virtualization Manager allows editing VMs and to associate the correct virtual or physical disks and obtain a configuration similar to the following:

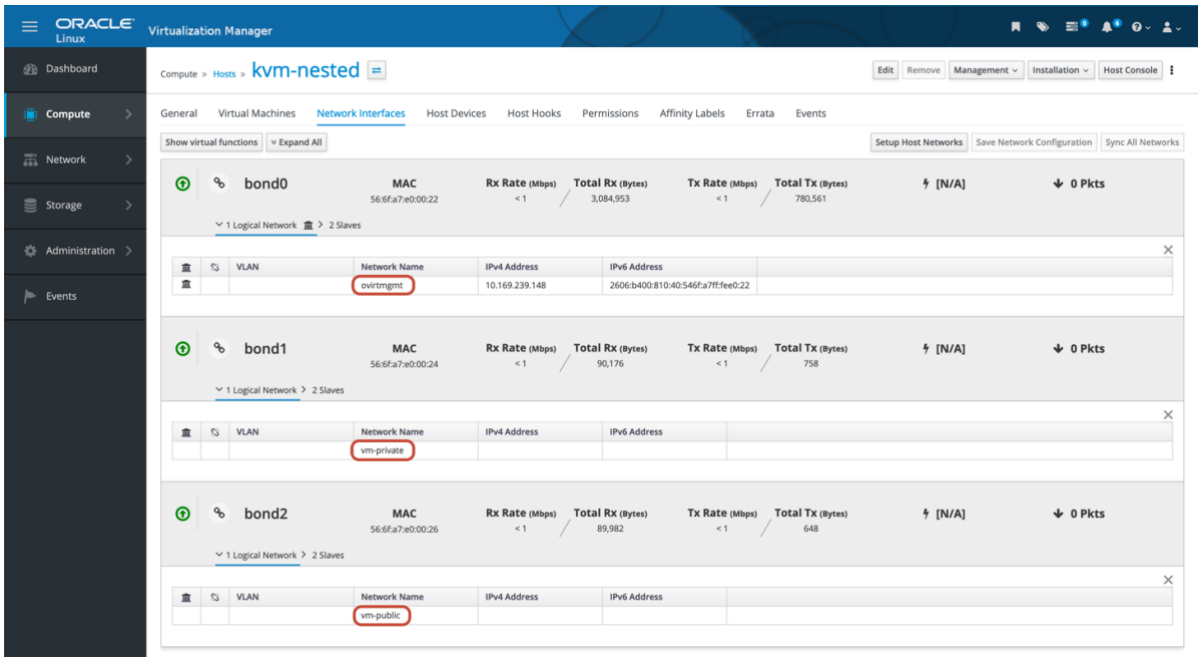


> ✓ **One virtual or physical disk must be edited and configured as "shared" to be presented to more VMs.**

Appendix B – Example for bonding setup

A bonding configuration needs to be applied on all Oracle Linux KVM servers that are part of the KVM Cluster where Oracle RAC virtual machines will be created. All network configuration related to the Oracle Linux KVM server must be executed using Oracle Linux Virtualization Manager's interface or by leveraging WS-API that the same exposes.

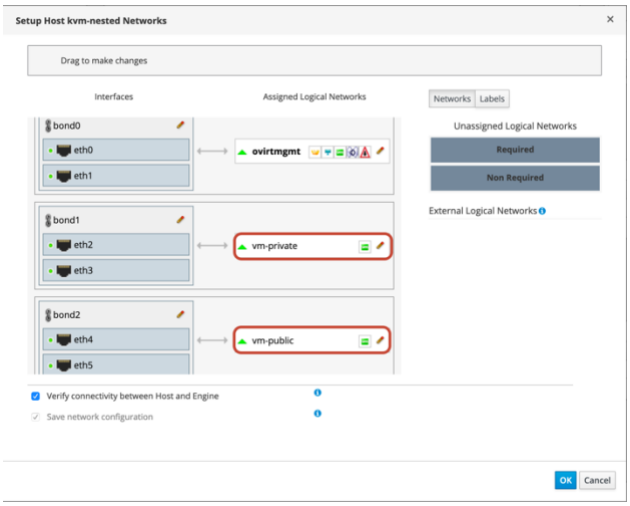
Each bond interface (one for Oracle Linux KVM Management, one for Oracle RAC public network and one for Oracle RAC private network) needs, at least, to rely on two different NICs, as in the following example:



- Bond0 = eth0 and eth1 (dedicated to Management, Live-Migration)
- Bond1 = eth2 and eth3 (dedicated to Oracle RAC Private-Network – VMs)
- Bond 2 = eth4 and eth5 (dedicated to Oracle RAC Public-Network – VMs)

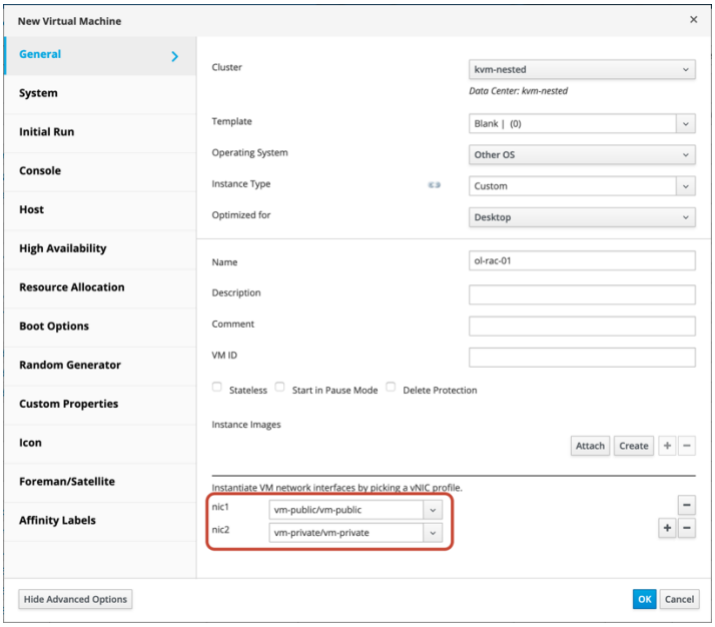
The bonding mode used in the example above is mode 1, active-backup. This is the easiest mode and will work on most switches without problems. Other modes might require specific switches and a certain switch configuration.

Once all bond configurations are defined, you have to correctly associate the interfaces to a network role, using Oracle Linux Virtualization Manager. For each KVM host that is part of the cluster, you have to correctly associate the logical network to the physical bond devices:



With networks created, you can proceed to create your own virtual machines that will be dedicated to Oracle Real Application Clusters.

While creating virtual machines you will have to associate one vNIC for each role needed. In the example below you can see a VM with two different vNICs, one dedicated to the public network and one dedicated to the private:



Starting with Oracle RAC 12c Release 1 you can create two different vNICs and configure Oracle Database with HA-IP as described in this document.

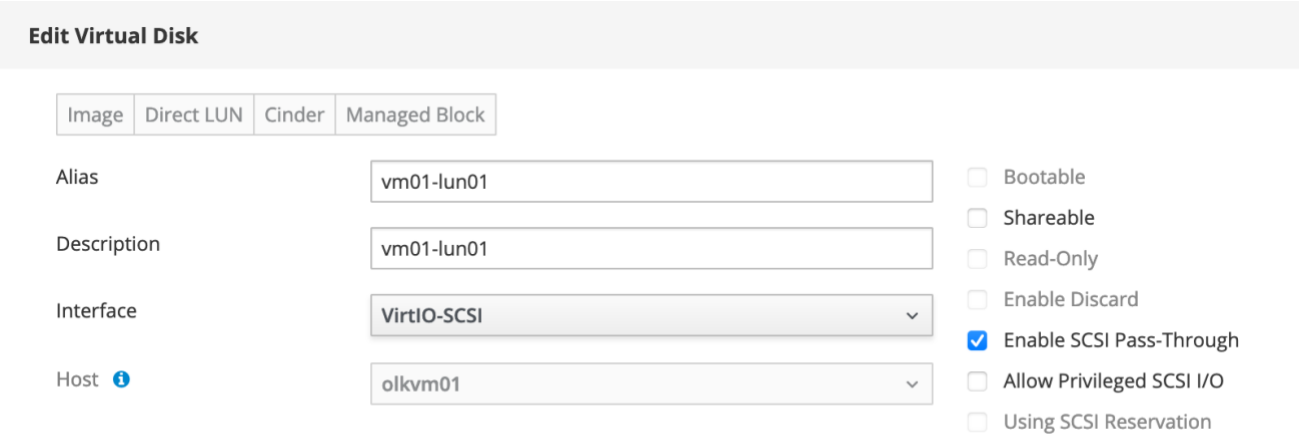
If the intention is to use HA-IP, the bond interface on Oracle Linux KVM server, dedicated to the private network, does not need to be created. Instead, associate each NIC to a specific network role and then associate two different vNICs to the VMs:

- Eth0 ➔ Network Role: "HA-IP 1of2" ➔ vNIC (1) for Private Network
- Eth1 ➔ Network Role: "HA-IP 2of2" ➔ vNIC (2) for Private Network

Appendix C – ASM Volumes migration from other hypervisor(s)

Oracle Linux Virtualization Manager does not support 4k block-storage devices (virtual-disks) and, the same thing is also valid for physical-disks presented to Virtual Machines.

In case of a migration of an Oracle Database from other hypervisor(s) where ASM volumes (*oracleasm*) were on physical disks, now presented to the KVM virtual machine, the same physical disks have to be presented in "Passthrough" mode:



Appendix D – References
Official Oracle Documentation

PRODUCT / REFERENCE	URL
Oracle Linux KVM on oracle.com	http://www.oracle.com/virtualization
Oracle Linux Virtualization Manager Official Documentation	https://docs.oracle.com/en/virtualization/oracle-linux-virtualization-manager/
Oracle Linux KVM User's Guide	https://docs.oracle.com/en/operating-systems/oracle-linux/kvm-user/
Oracle RAC on oracle.com	https://www.oracle.com/database/real-application-clusters/index.html
Oracle Database Templates for Oracle Linux KVM	https://www.oracle.com/database/technologies/rac/vm-db-templates.html
Oracle Clusterware on oracle.com	http://www.oracle.com/technetwork/database/database-technologies/clusterware/overview/index.html
Oracle ASM	https://www.oracle.com/database/technologies/rac/asm.html
Oracle ASMLib	https://www.oracle.com/linux/technologies/asmlib/

ORACLE

Connect with us

Call +**1.800.ORACLE1** or visit **oracle.com/linux**. Outside North America, find your local office at: **oracle.com/contact**.

blogs.oracle.com/linux facebook.com/oraclelinux twitter.com/oraclelinux