ORACLE

Solution-in-a-Box: Deploying Database Systems and Application VMs on Oracle Database Appliance (Bare Metal)

A step-by-step guide

November 5, 2021 | Version 1.2

PURPOSE STATEMENT

This document provides step-by-step guide to utilize the KVM technology on Bare Metal Oracle Database Appliance (ODA) release 19.10.0.0.0 and later. It provides guidelines how to create the Application tier together with the database tier on the same ODA hardware. This concept is also known as 'Solution-in-a-Box'.

TABLE OF CONTENTS

ORACLE DATABASE APPLIANCE (ODA) BASICS

Oracle Database Appliance (ODA) is a pre-built, ready to deploy database platform. ODA is an engineered system, an integrated, full-stack solution developed for Oracle Database. Oracle hardware and software are designed to work perfectly together and are able to run crucial customer workloads faster, at lower costs, and with greater security than multivendor, on-premises 'build your own' solutions. Each ODA system consists of server node(s), networking, and redundant storage. ODA can be a single node model (S for small, M for Medium and in case of X6-2 L for Large) or a two node HA model (HA stands for high availability). With the HA version you get two clustered nodes and you are able to create highly available RAC databases easy and fast via either the browser user interface (BUI) or the command-line interface (CLI).

REASONS FOR VIRTUALIZATION

There are several reasons why virtualization might be considered. Virtual Machines (VMs) can be used to consolidate and to utilize resources more effectively. Rather than disable not licensed cores, you can utilize them for other workloads. This can reduce the number of physical servers, resulting in space, power maintenance reductions, and cost savings. Virtualization is a great technology that provides application and database isolation and enables bundling of applications, middleware, and databases on the same hardware.

ODA VIRTUALIZATION

ODA has provided virtualization since version 2.5. To use virtualization on ODA, it had to be deployed with a special image. Users had to decide before deploying ODA to go with either the bare metal (BM) image or the virtualized platform image (VP). To change from one layout to the other, the ODA had to be reimaged from scratch. ODA VP was only available for ODA High Availability (2 node ODA) and not for single node ODAs.
Virtualization on ODA was introduced to allow the utilization of cores not licensed for Oracle Database. To avoid leaving these cores unused, virtualization allowed to utilize them for virtual machines hosting applications, webservers and so on. To host not only the database tier but as well the application and/or web tier on the same ODA is also known as "Solution in a Box". This concept is widely adopted and not only used for Oracle Applications like EBS, WLS, Peoplesoft, Oracle Hospitality to name a few, but as well for many customer and third party software vendor developed applications.
In the past virtualization on ODA was implemented via Oracle Virtual Machine (OVM) which is a type 1 bare metal hypervisor based on xen technology. This option will still be available for some time, but in the future, virtualization on ODA will only be available via KVM (last OVM version will be 19.13). KVM stands for Kernel-based Virtual Machine and is a type 2 host OS-based hypervisor. KVM provides modules that enable the Linux kernel to be used as a hypervisor. It is available on all ODA hardware models (single node and HA). KVM is a mature technology that has been around since 2007 and offers several advantages. To name one, ODA will not need separate BM and VP images anymore.
Starting with ODA 19.9, ODA integrated KVM support for applications. A user can use ODA's built-in BUI or the command line to manage the Application VM life cycle. The user is responsible for managing the application inside the KVM, for the ODA software tooling it is a 'black box'. Starting with ODA 19.10, ODA integrates KVM support for Oracle databases. A user can use ODA's built-in BUI or the command line to manage both DB System (ODA's database KVM system) life cycle and databases inside DB Systems. DB Systems allow Oracle hard partitioning licensing, thus a user only needs to have Oracle Database licenses for the CPU cores assigned to the DB System.

TERMINOLOGY

To understand the terminology used in this document or when reading the Oracle documentation, here a list of the often used terms:

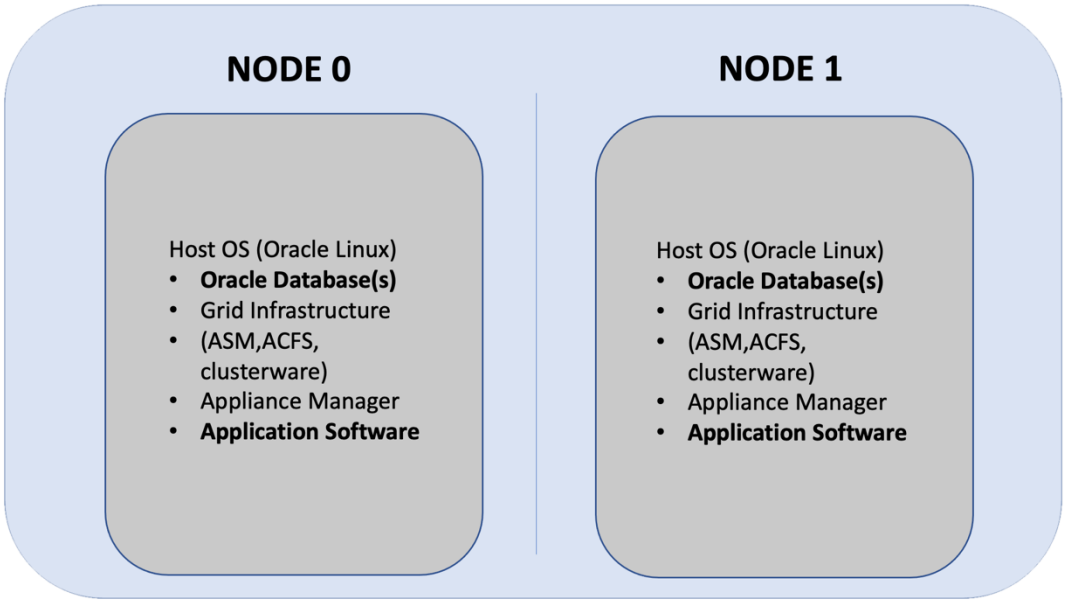
- Virtual Machine (VM): image and virtual resources that behave like an actual server within the host server
- KVM guest : the software part of an application VM with an independent OS instance
- DB System (KVM Database System): A database VM which runs an Oracle Database on ODA. Created and managed by ODA tooling
- VM Storage: central location for storing resources needed to create virtual machines
- Virtual Network: separate virtual network for the VM
- Virtual Disk: additional storage option for VMs to add additional block storage
- KVM Database System Networks: default networks created on ODA, which are the 'public network' (vnetwork pubnet) and the internal managed 'cluster interconnect' and ASM network

SOLUTION IN A BOX DESIGNS

There are three basic scenarios on how to install the application along with the database in the same ODA. For an ODA HA there are additional designs since your database can be a RAC or a single instance database on one node only and depending on your application you might be able to run it on one or both nodes. The basic scenarios are:

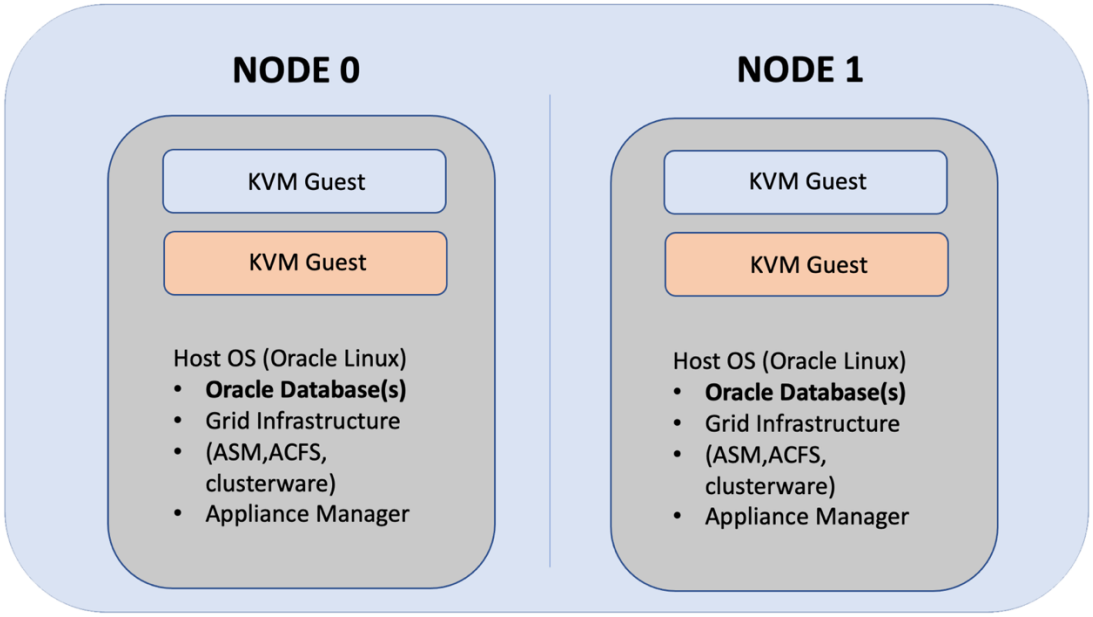
Application and Database installed in BM

This option is only applicable for a few applications which do not interfere with the database software and OS installation. One supported example is the Oracle Enterprise Manager software.



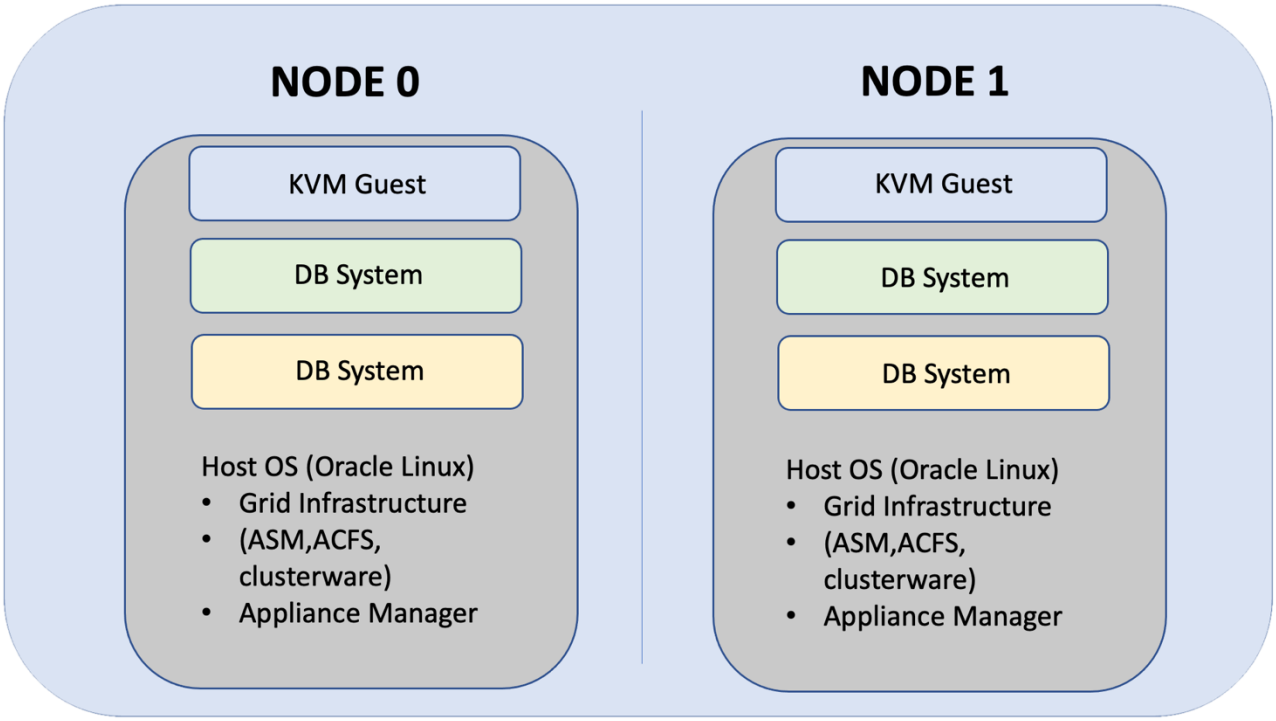
Database installed in BM, Application installed in KVM guest

This option doesn't allow hard partitioning, means there is no separation of cores for database and application possible.



Database and Application in separate KVM guests

This third option is ideal to separate resources for databases and applications and license them independently. This step-by-step guide covers an example of the third case since this will be the best solution for most use cases.



HA CONSIDERATIONS

With and ODA HA the best High Availability option is to use a RAC database. A RAC One Node design might be sufficient for your availability requirements as well. These Options are only available with an Enterprise Edition (EE) licence. Standard Edition (SE) Oracle databases are protected by Standard Edition High Availability (SEHA) feature.
The KVM guests you create for your applications are setup by default to auto-restart and auto-failover.
You can confirm the setting with the command: 'odacli describe-vm'. This will return:

- Auto start: YES
- Fail over: YES

NETWORKING CONSIDERATIONS

ODA KVM vnetwork supports two type of networks, bridged and bridged-vlan. For bridged network a bridge is created and a NIC or a bond is attached to it. On the public network (selected during appliance deployment) a bridge with the name pubnet is attached. If a VM needs access to pubnet it can be attached to this network. This pubnet vnetwork cannot be modified or deleted.
If additional bridged vnetwork are required they can be created on other interfaces that aren't used for pubnet. For DB Systems it isn't possible with 19.10 to use a Vnetwork other than pubnet. This restriction has been removed with the later versions.
Vlans can be created on all available public interfaces, including the interface where public network is already configured.
When planning to run one or more DB Systems, be aware that every DB System requires it's own set of VIPs and SCAN IPs.

STEP BY STEP INSTRUCTIONS TO CREATE APPLICATION GUEST KVMS AND DB SYTEMS

Create an Application VM with Linux OS

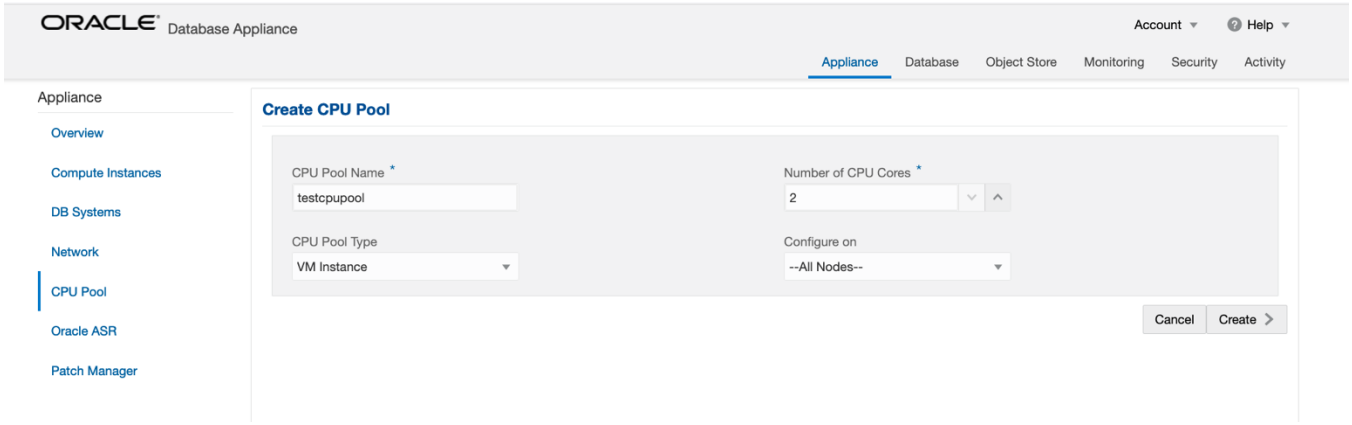
Below an example how to setup an “Application VM” or “Compute Instance” with Linux Operating System. While some of the steps are mandatory others are optional:

1.) create a VM CPU Pool (optional)
2.) Create VM Storage
3.) Create a Virtual Network (optional)
4.) Create Virtual Disk (optional)
5.) Create the VM Instance
6.) Finish the OS installation
7.) Test the application VM

If you execute the optional steps depends on how you designed your Solution-in-a-Box. If your application VM should only use a defined number of CPUs you should for example create a VM CPU Pool as part of the process. To create VM Storage is mandatory, several Application VMs can share a single VM Storage. If you have specific network requirements you can create additional Virtual Networks for these purpose. Another optional step is the creating of Virtual Disks. One “VM disk” is created for each application guest KVM as part of the VM instance creation, additional Virtual Disks can be created if your Application VM requires this, they can be shared between different Application VMs if required.

1.) create a VM CPU Pool (optional)

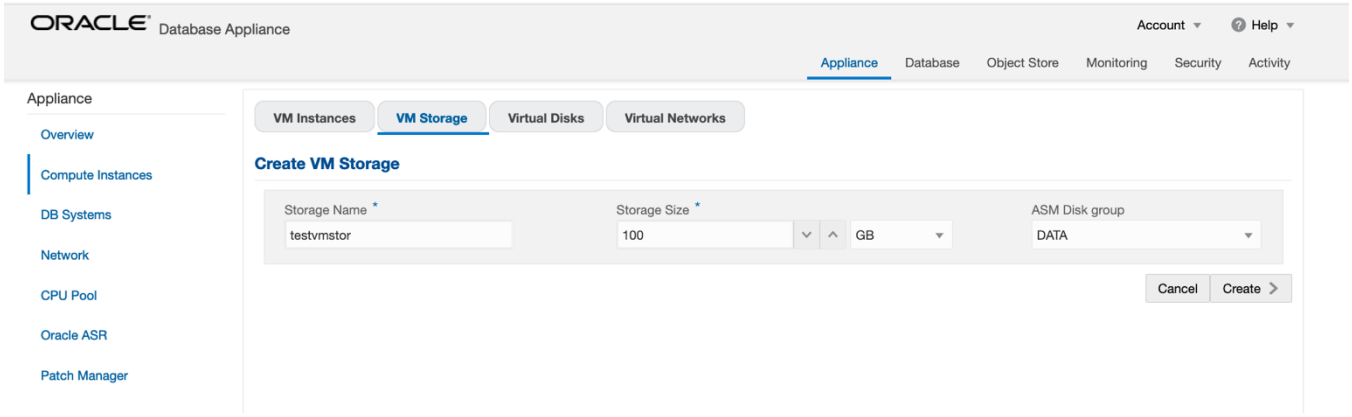
Create a CPU Pool to specify how many CPU cores your KVM should use, this CPU Pool will be later assigned to the KVM Guest. Under Appliance go to CPU Pool and define the name the number of CPU cores and as CPU Pool Type select ‘VM Instance’.




```
Alternative cli command:
# odacli create-cpupool -n testcpupool -c 2 -vm
```

2.) Create VM Storage

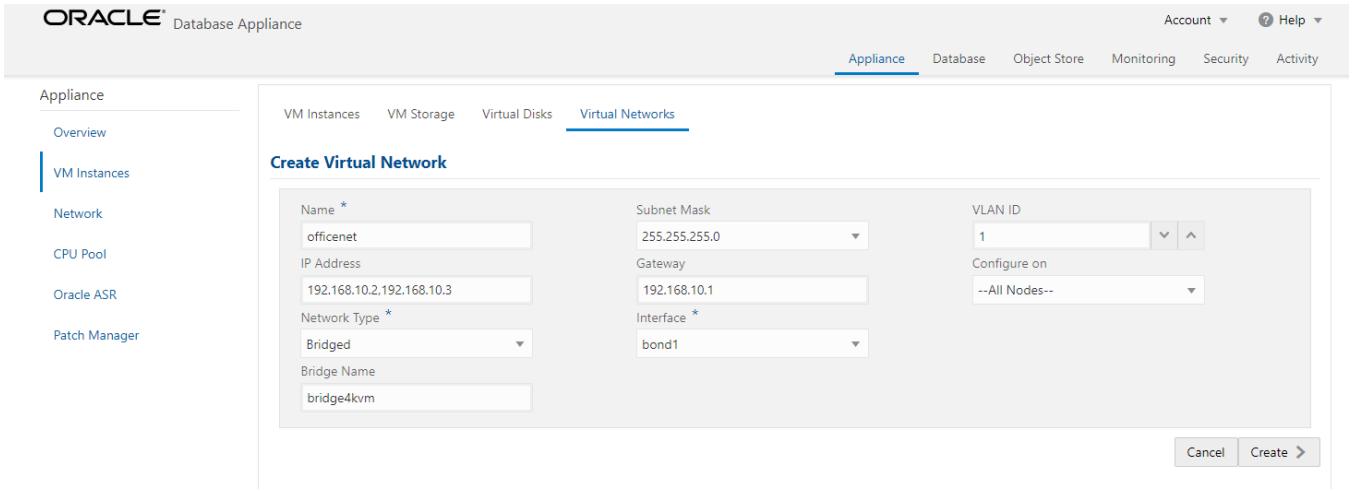
Under Appliance go to VM Storage and define name, size and ASM disk group.




```
Alternative cli command:
# odacli create-vmstorage -n testvmstor -s 100G -dg DATA
Note: the -dg default value is DATA
```

3.) Create a Virtual Network (if required)

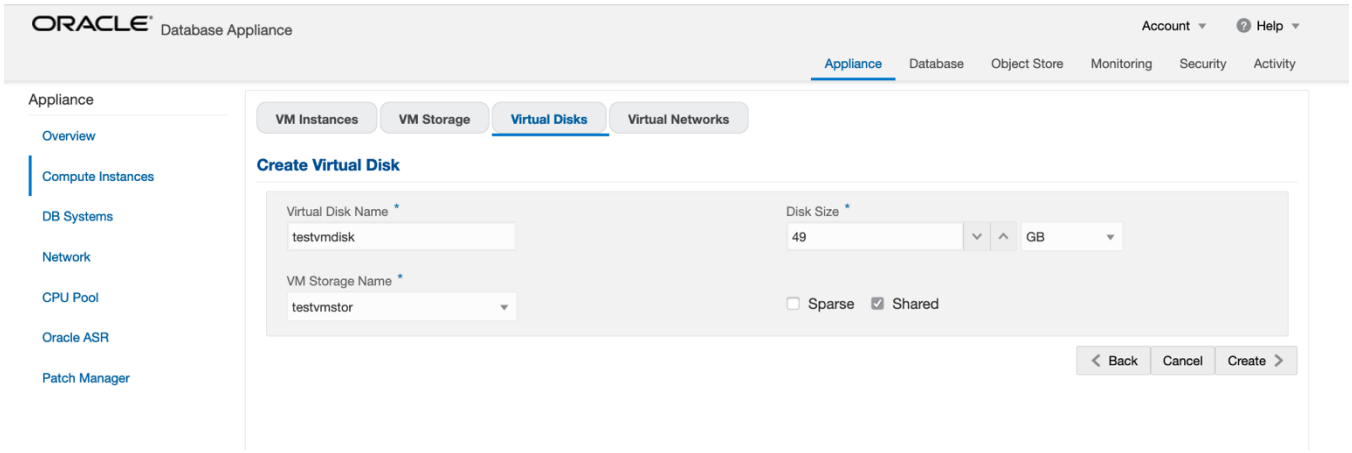
Any VM that needs to access the public network has to be attached to the pubnet Vnetwork.
No other bridged network than the default pubnet bridge is allowed to be created on the public network interface.
Additional bridged vnetwork (public network without tagged VLAN configuration) can be created on other interfaces that are not used for the public network.
It isn't mandatory to assign IPs to the bridge. This is only required if communication between the BM host and the KVM guest is desired. When 2 IP addresses are assigned to a HA machine, the first will be assigned to the bridge on node0 and the second IP to node1.




```
Alternative cli command:
# odacli create-vnetwork –name officenet –bridge bridge4kvm –type bridged –interface bond1 –
ip xx.xx.xx.xx –gateway xx.xx.xx.xx –netmask xx.xx.xx.xx
```

4.) Create Virtual Disk (if required)

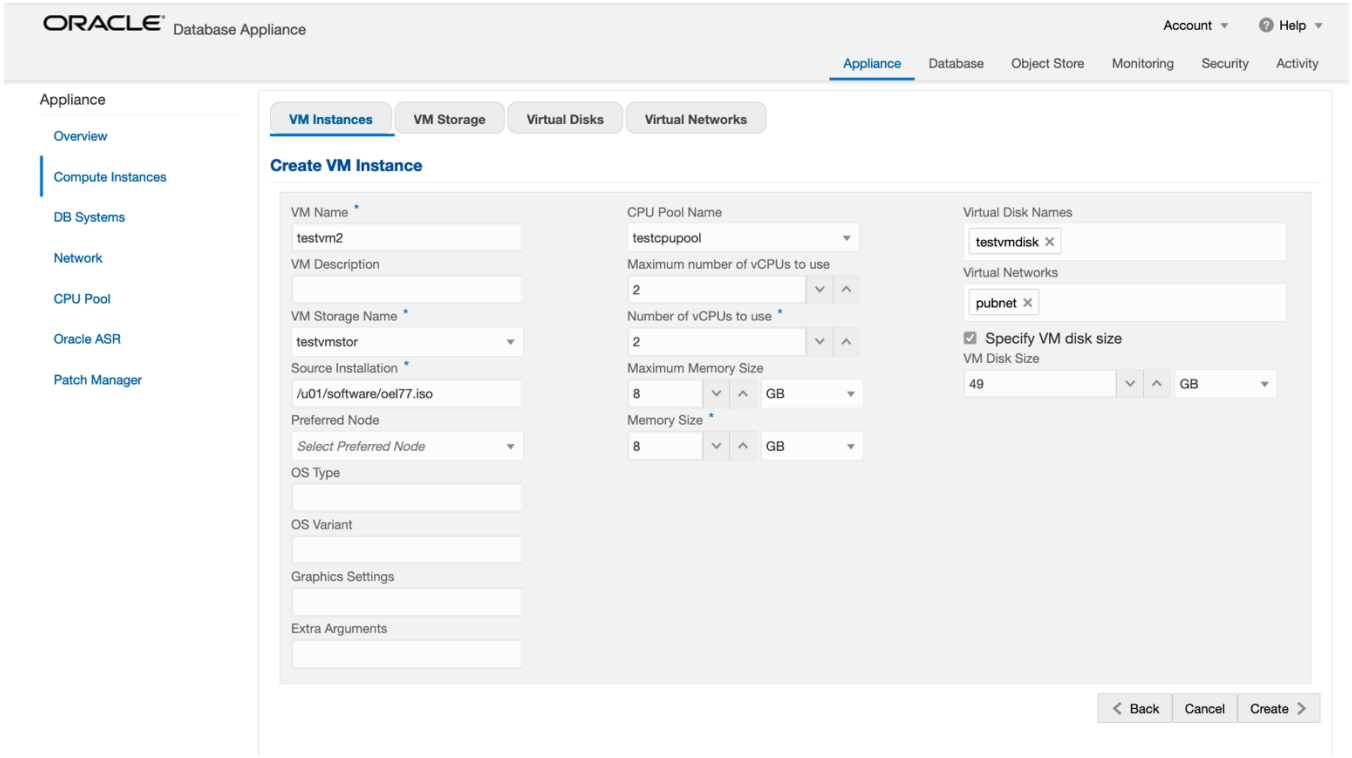
During the VM creation a VM disk will be created to host the OS. In case additional disks are required you should create them in advance. Using virtual disks gives you more control over the storage layout of your VM.




```
Alternative cli command:
# odacli create-vdisk -n testvmdisk -vms testvmstor -s 49G -sh
```

5.) Create the VM Instance

To create the VM go to 'VM Instances' and 'Create VM'. Type in the mandatory information for your VM plus the additional configuration you would like to use for your VM guest. Mandatory information is the 'VM name', 'VM Storage Name', the path to the application to install, under ‚Source installation'. You have as well to define how many CPUs and how much memory to use for the VM. If you have created a CPU Pool a Virtual Disk or a Virtual Network, select them into the respective fields. Specify the size of the OS disk to be created for your VM guest in "VM Disk Size". If you need your VM to access the DB System via the pubnet, make sure 'pubnet' is selected in the 'Virtual Networks'.




```
Alternative cli command:
# odacli create-vm -n testvm2 -vc 2 -m 8G -vms testvmstor -vd testvmdisk -s 49G -cp
testcpupool -vn pubnet -src /u01/software/oel77.iso
```

6.) Finish the OS installation

If the creation of the VM was successful, Use the odacli describe-vm command to check the VNC port.

```
# odacli describe-vm -n testvm2
VM details
--------------------------------------------------------------------------------
                        ID:  4956584d-7409-4d38-8a22-60653a6c2ae1
                      Name:  testvm2
                   Created:  2021-05-10 02:24:01 HDT
                   Updated:  2021-05-10 02:24:01 HDT
                VM Storage:  testvmstor
               Description:  NONE
                   VM size:  49.00 GB
                    Source:  oel77.iso
                   OS Type:  NONE
                OS Variant:  NONE
         Graphics settings:  vnc,listen=127.0.0.1
              Display Port:  127.0.0.1:2     <-VNC port

 Status
--------------------------
               Current node:  node1
              Current state:  ONLINE
               Target state:  ONLINE

 Parameters
--------------------------
             Preferred node:  NONE
                Boot option:  NONE
                 Auto start:  YES
                  Fail over:  YES

                             Config                      Live
                             --------------------------  --------------------------
                    Memory:  8.00 GB                     8.00 GB
                Max Memory:  8.00 GB                     8.00 GB
                vCPU count:  2                           2
             Max vCPU count:  2                           2
                  CPU Pool:  testcpupool                 testcpupool
         Effective CPU set:  2-3,38-39                   2-3,38-39
                     vCPUs:  0:2-3,38-39                 0:2-3,38-39
                             1:2-3,38-39                 1:2-3,38-39
                    vDisks:  testvmdisk:vdb              testvmdisk:vdb
                 vNetworks:  pubnet:52:54:00:7d:7b:6b    pubnet:52:54:00:7d:7b:6b
```

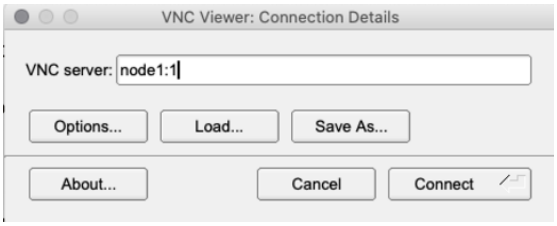
7.) Connect to the ODA BM node via VNC

To find out the VNC port the BM (not the KVM guest) uses run the vncserver command.

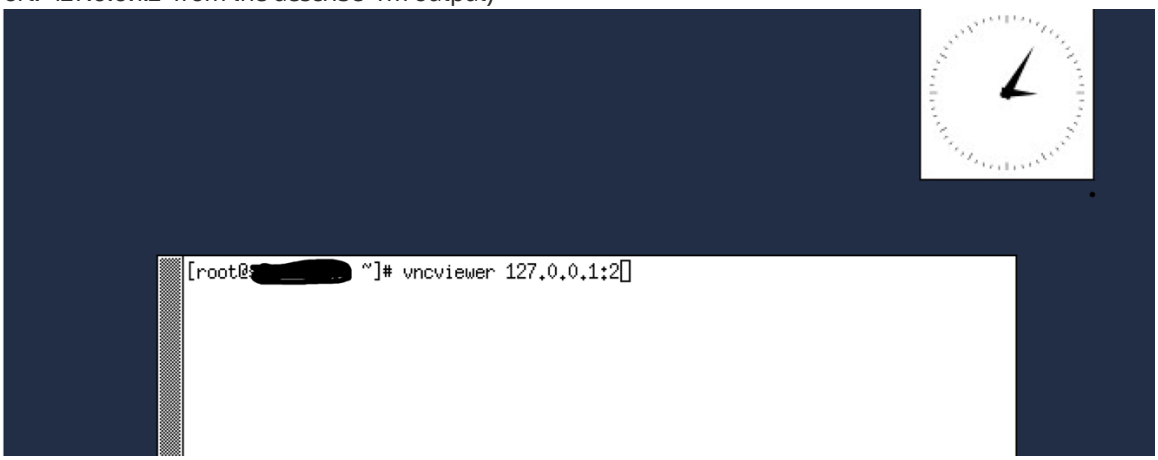
```
# vncserver
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
  LANGUAGE = (unset),
  LC_ALL = (unset),
  LC_CTYPE = "UTF-8",
  LANG = "en_US.UTF-8"
    are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").

You will require a password to access your desktops.

Password:
Verify:
Would you like to enter a view-only password (y/n)? n
A view-only password is not used
xauth:  file /root/.Xauthority does not exist
```

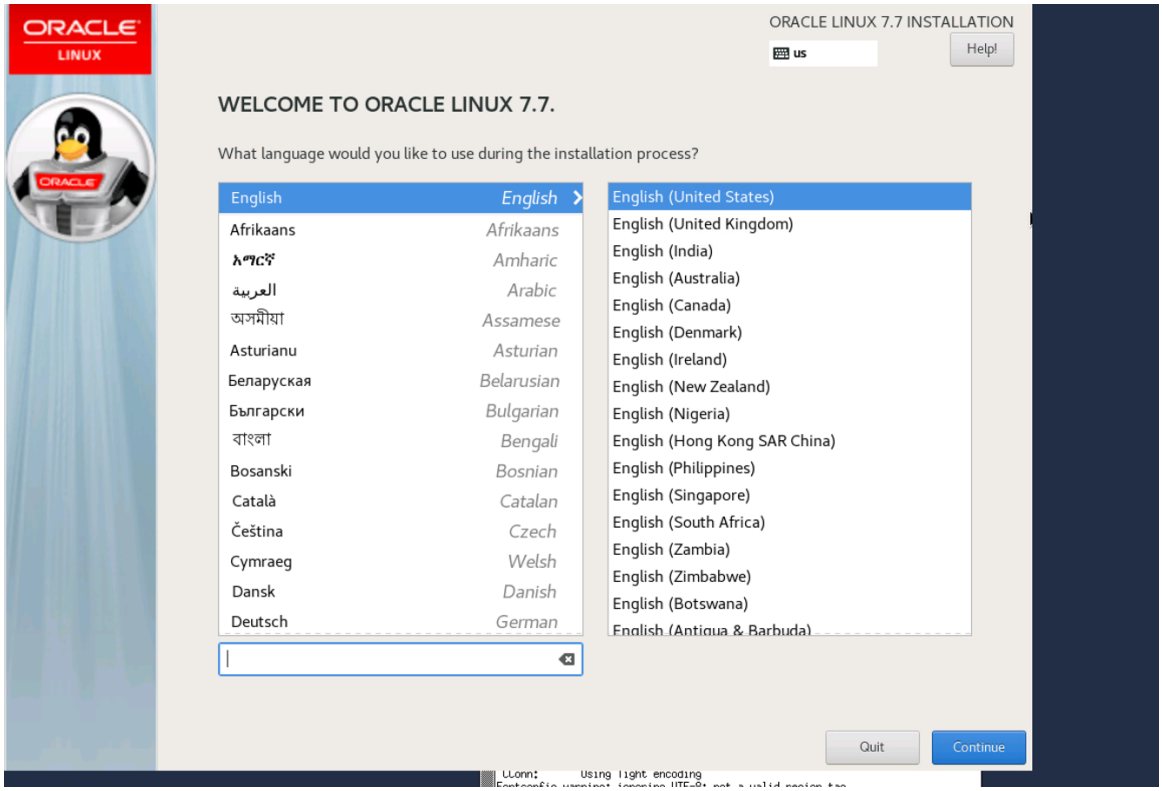
```
New 'node1:1 (root)' desktop is node1:1  ← VNC port

Creating default startup script /root/.vnc/xstartup
Creating default config /root/.vnc/config
Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/node1:1.log
```

Now use a VNC client of your choice and connect to host:vncport (the host is your ODA BM, the VNC port is from vncserver command).

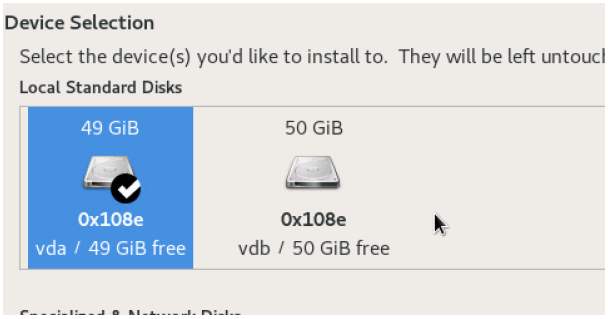


When you see the command prompt in the VNC viewer type (from the example output) : vncviewer 127.0.0.1:2 (this is the Display Port: 127.0.0.1:2 from the describe-vm output)





8.) Complete the installation process as you would do with any other OS installation. In the 'Device Selection', you see the disk created by 'create-vm' and the additional one from 'create-vdisk'. Make sure to use the right one for the OS install.



After you are asked and you confirmed to reboot, you have to wait a short time and VNC back in order to complete the setup (e.g., accept the license agreement).

9.) Configure the Application VM network

After the Linux OS installation is completed for the Application VM, the VM might have several interfaces. In my example I want to assign the IP address to the pubnet and have to find the MAC address for this interface.

```
# odacli describe-vm -n testvm2 | grep vNetworks
                vNetworks:   pubnet:52:54:00:7d:7b:6b    pubnet:52:54:00:7d:7b:6b
```

Login to the Application VM via VNC client and run the command "ip link show" to identify the interface with the MAC address found with the command above.

```
# ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group
default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT
group default qlen 1000
    link/ether 52:54:00:7d:7b:6b brd ff:ff:ff:ff:ff:ff
3: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN mode DEFAULT
group default qlen 1000
    link/ether 52:54:00:42:63:d7 brd ff:ff:ff:ff:ff:ff
4: virbr0-nic: <BROADCAST,MULTICAST> mtu 1500 qdisc pfifo_fast master virbr0 state DOWN mode
DEFAULT group default qlen 1000
    link/ether 52:54:00:42:63:d7 brd ff:ff:ff:ff:ff:ff
```

Edit the /etc/sysconfig/network-scripts/ifcfg-eth0 and /etc/resolv.conf for DNS inside the VM. Refer to your operating system specific document to configure the network interface. The following is just an example.

```
# cat /etc/sysconfig/network-scripts/ifcfg-eth0
TYPE=Ethernet
BOOTPROTO=none
DEFROUTE=yes
NAME=eth0
DEVICE=eth0
NM_CONTROLED=no
ONBOOT=yes
IPADDR=<your IP address>
PREFIX=<your subnet prefix>
GATEWAY=<your gateway>
```

```
# cat /etc/resolv.conf
search us.oracle.com
nameserver <your first DNS server>
nameserver <your second DNS server>
```

10.) Test if the desired connectivity to the VM guest

Test that you are able to connect from an to all machines and clients (database, external machines etc.). To connect as root via ssh you might have to change SELINUX to permissive mode in /etc/selinux/config.
If it is desired, test this connectivity as well after you migrated the VM guest to the other node. To migrate the node execute the below commands.

```
# odacli describe-vm -n testvm2 | grep 'Current node'
 Current node: node1

# odacli migrate-vm -n testvm2 -to node2

# odacli describe-vm -n testvm2 | grep 'Current node'
 Current node: node2
```

Create an Application VM with Windows OS

NOTE: The procedure below works from software ODA version 19.12 on. If you aren't able to upgrade, please follow the instructions from MOS note 2748946.1 instead.

Before you start, get the 'Oracle VirtIO Drivers Version for Microsoft Windows' which you can download Oracle Software Delivery Cloud or from My Oracle Support (MOS).
For details and the links please refer to https://docs.oracle.com/en/operating-systems/oracle-linux/kvm-user/kvm-virtio.html#kvm-virtio-download .

BUI Versions 19.11 and 19.12 don't have the option to define an extra source yet (CLI flag -esrc). Thus we can only create the VM via the odacli interface.

1.) create a VM CPU Pool (optional, for cases you need dedicated CPU resources for the Windows VM)

```
# odacli create-cpupool -n winvmpool -c 2 -vm
```

2.) Create VM Storage

```
# odacli create-vmstorage -n winvmstor -s 100G -dg DATA
```

3.) Create Virtual Disk (optional, for cases you need additional disks for the Windows VM)

```
# odacli create-vdisk -n winvmdisk -vms winvmstor -s 49G -sh
```

4.) Create the VM Instance

```
# odacli create-vm -n winvm -vc 2 -m 8G -vms winvmstor -s 49G -vd winvmdisk -cp winvmpool -vn pubnet -src
/tmp/win19.iso -esrc /tmp/winvirtio.iso -ost windows
```

5.) Use the odacli describe-vm command to identify the VNC port

```
# odacli describe-vm -n winvm
VM details
--------------------------------------------------------------------------------
                 ID:  7b741c6a-7073-4d39-9270-24698b887135
               Name:  winvm
            Created:  2021-08-03 02:59:42 HDT
            Updated:  2021-08-03 02:59:42 HDT
          VM Storage:  winvmstor
        Description:  NONE
            VM size:  49.00 GB
             Source:  win19.iso
            OS Type:  windows
         OS Variant:  NONE
   Graphics settings:  vnc,listen=127.0.0.1
      Display Port:  127.0.0.1:0

 Status
--------------------------
        Current node:  scaoda8181
       Current state:  ONLINE
        Target state:  ONLINE

 Parameters
-------------------------
      Preferred node:  NONE
        Boot option:  NONE
        Auto start:  YES
         Fail over:  YES

                  Config                 Live
                  ---------------------  ---------------------
           Memory:  8.00 GB                8.00 GB
        Max Memory:  8.00 GB                8.00 GB
        vCPU count:  2                   2
     Max vCPU count:  2                   2
         CPU Pool:  winvmpool               winvmpool
  Effective CPU set:  0-1,32-33             0-1,32-33
            vCPUs:  0:0-1,32-33            0:0-1,32-33
                 1:0-1,32-33            1:0-1,32-33
           vDisks:  winvmdisk:vdb           winvmdisk:vdb
```

```
vNetworks:  pubnet:52:54:00:84:8f:cd   pubnet:52:54:00:84:8f:cd
```

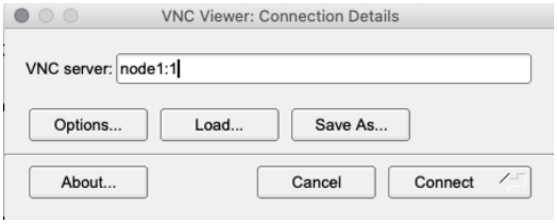
6.) Connect to the ODA BM node via VNC

To identify the VNC port of the BM (not the KVM guest) run the vncserver command.

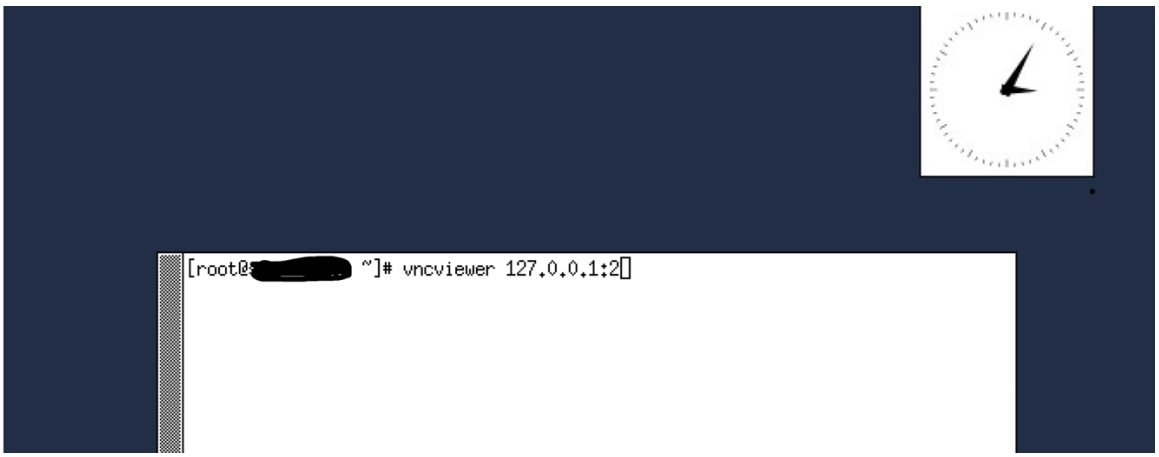
```
# vncserver
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
  LANGUAGE = (unset),
  LC_ALL = (unset),
  LC_CTYPE = "UTF-8",
  LANG = "en_US.UTF-8"
    are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").

You will require a password to access your desktops.

Password:
Verify:
Would you like to enter a view-only password (y/n)? n
A view-only password is not used
xauth:  file /root/.Xauthority does not exist

New 'node1:1 (root)' desktop is node1:1  ← VNC port

Creating default startup script /root/.vnc/xstartup
Creating default config /root/.vnc/config
Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/node1:1.log
```

7.) Connect to the VM guest via VNC

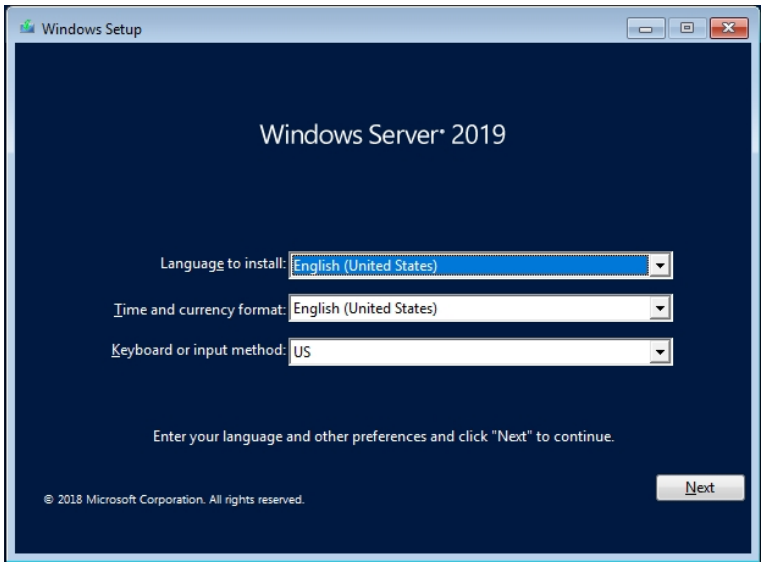
Use a VNC client of your choice and connect to host:vncport (the host is your ODA BM, the VNC port is from vncserver command).

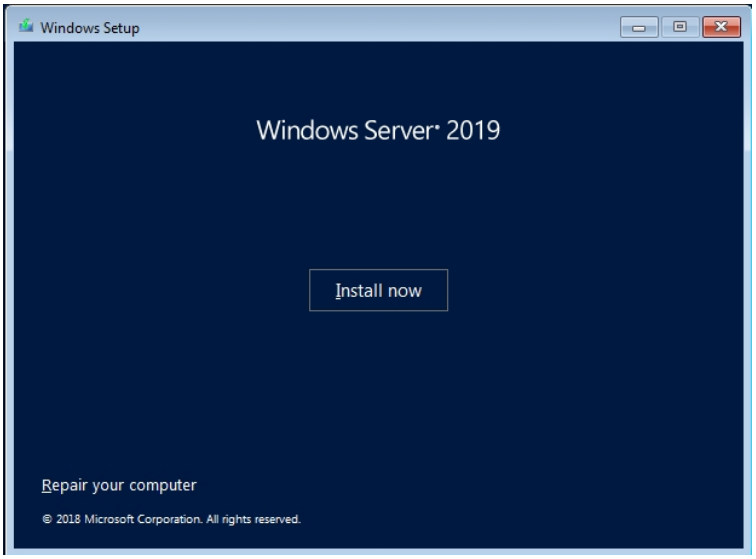


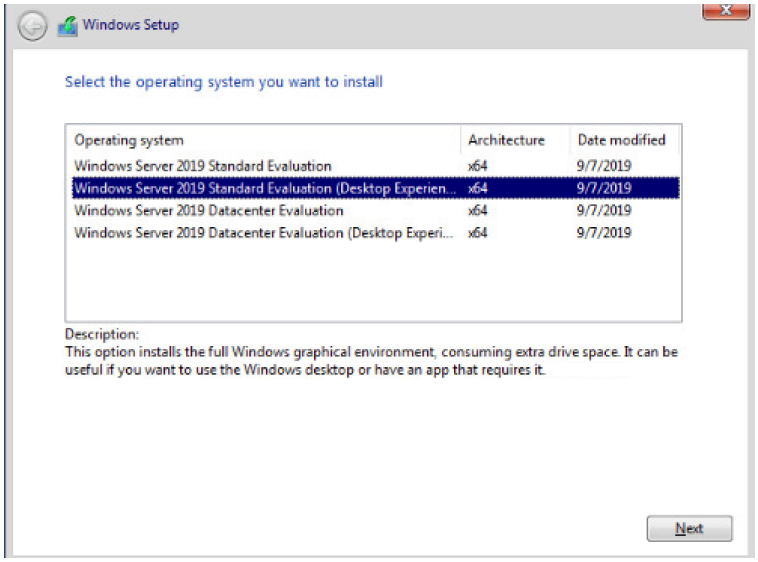
When you see the command prompt in the VNC viewer type (from the example output) : vncviewer 127.0.0.1:2 (this is the Display Port: 127.0.0.1:2 from the describe-vm output in step 5.))



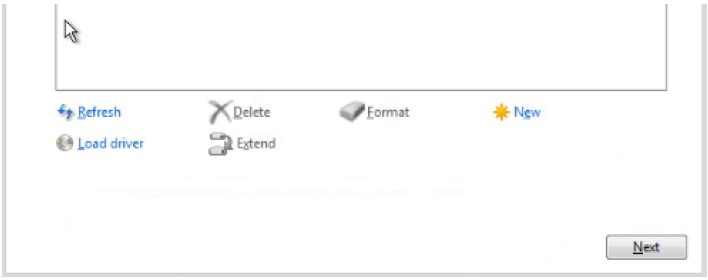
8.) Complete the Windows OS installation



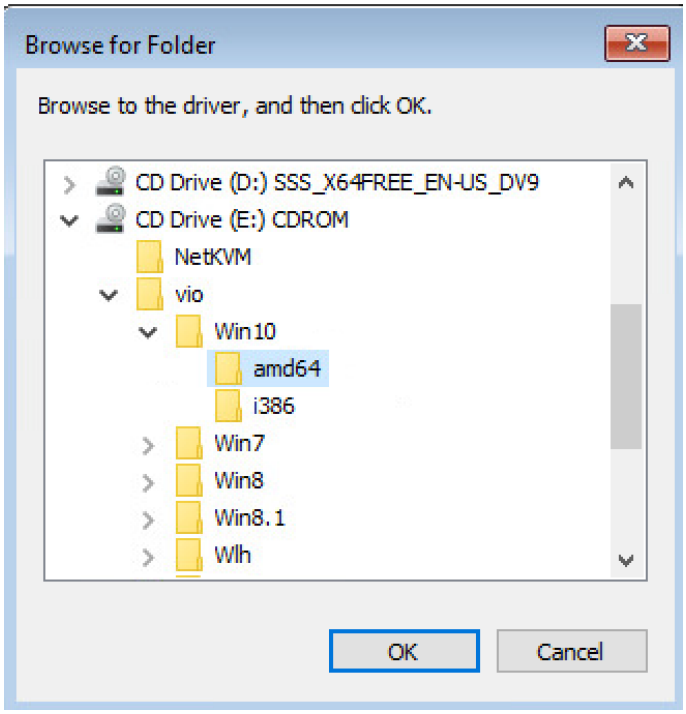


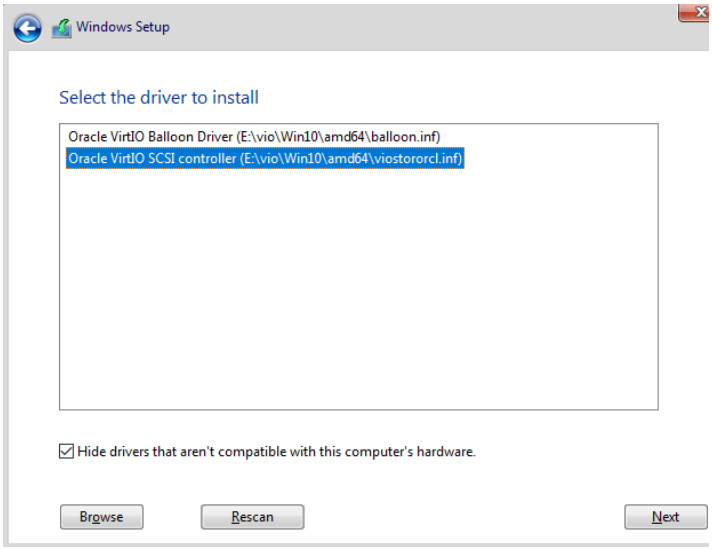


After a few confirmation windows the 'Where do you want to install Windows?' will appear. Select 'load driver'.

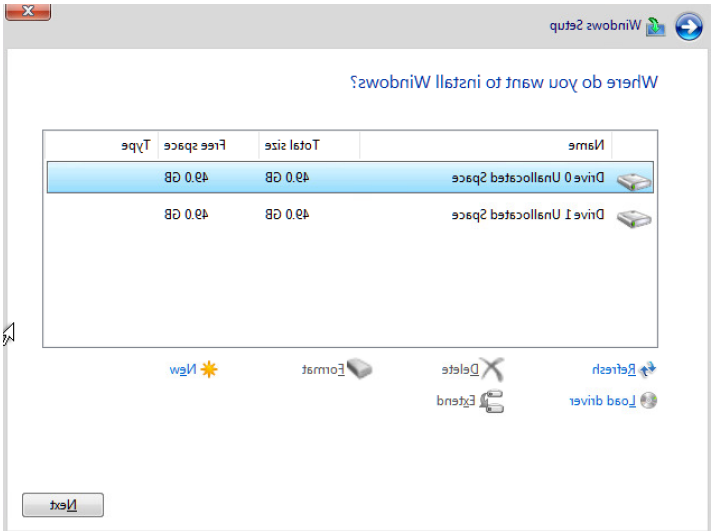


Navigate to the VirtIO driver on E: CDROM drive. Go to vio/Win10/amd64 and choose the viostorcl.inf from the list.

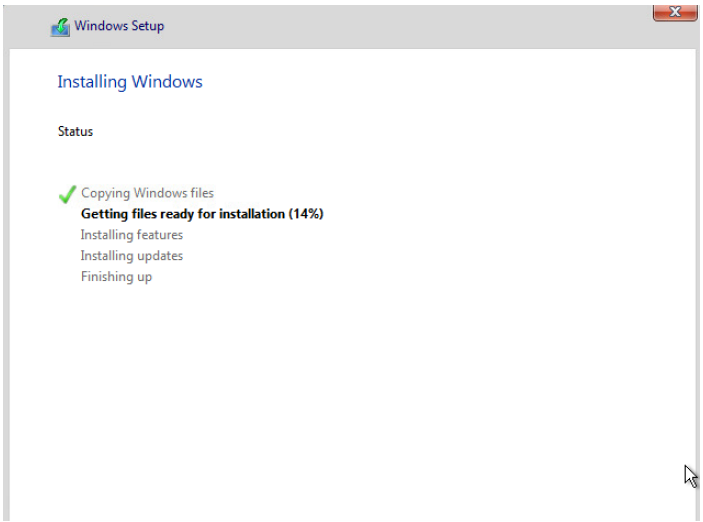




After loading the driver the pre-created Vdisk and the extra Vdisk (winvmdisk) will be displayed. Highlight the first and continue with the installation.



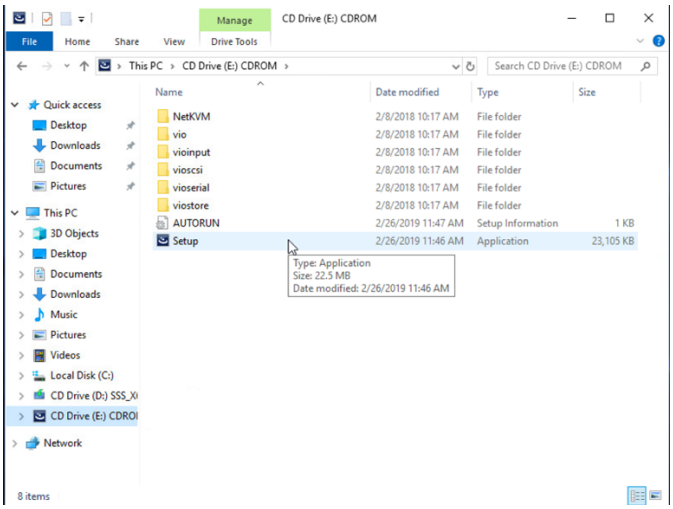
The installation will start.

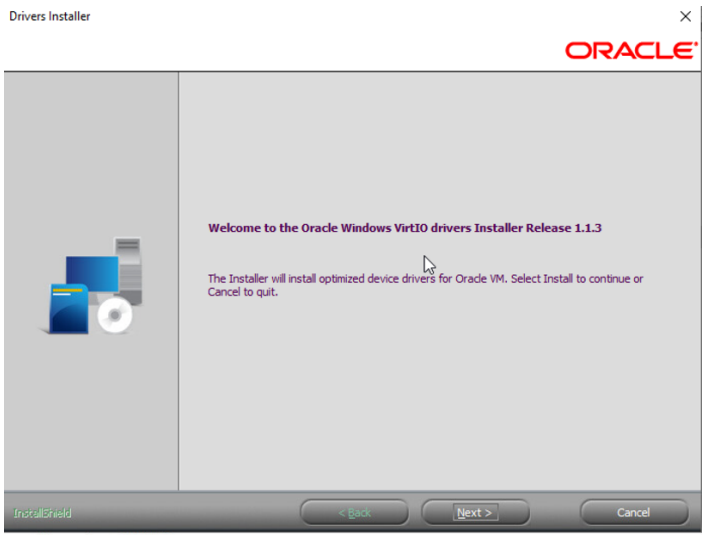


If the VNC connection terminates, wait a few seconds and reconnect. If asked to set the password for the Windows VM, provide the one of your choice.

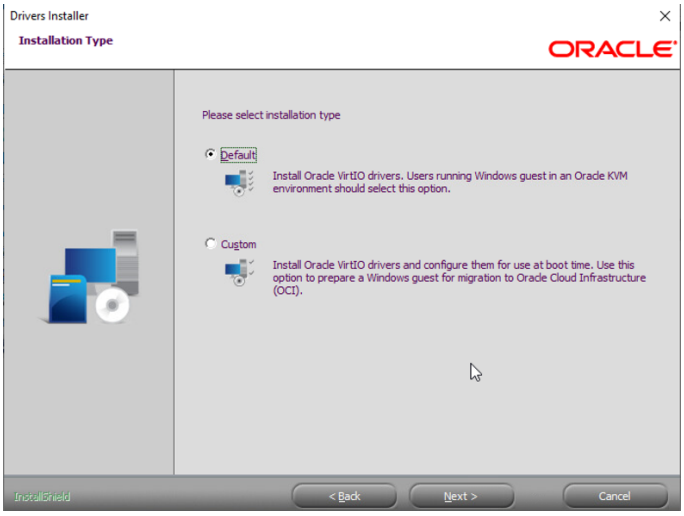
In order to send 'Ctrl+Alt+Delete' to the Windows OS, click inside the window and press F8 key (on a Mac fn+F8).

9.) Install the VirtIO drivers by running Setup from the CDROM drive E:

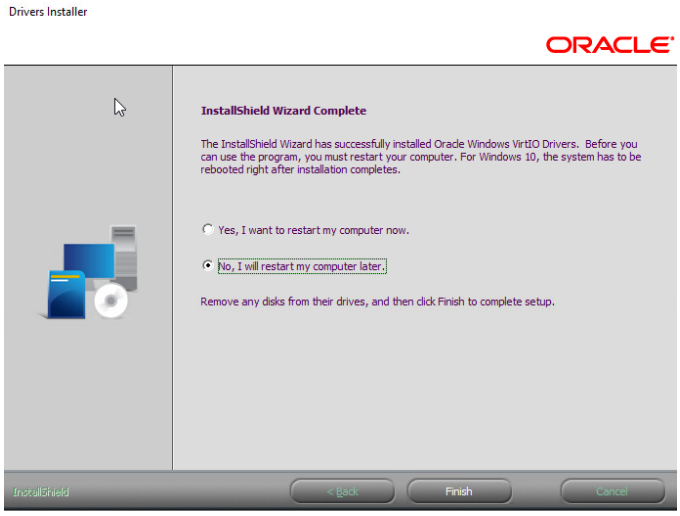


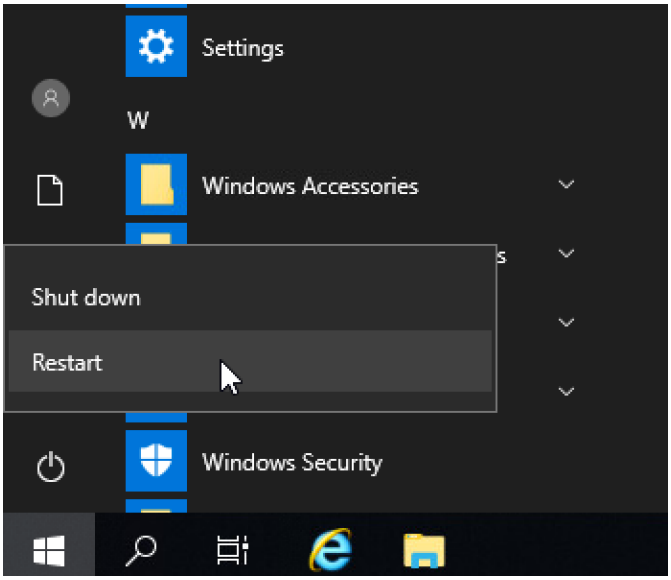


Select the default option.



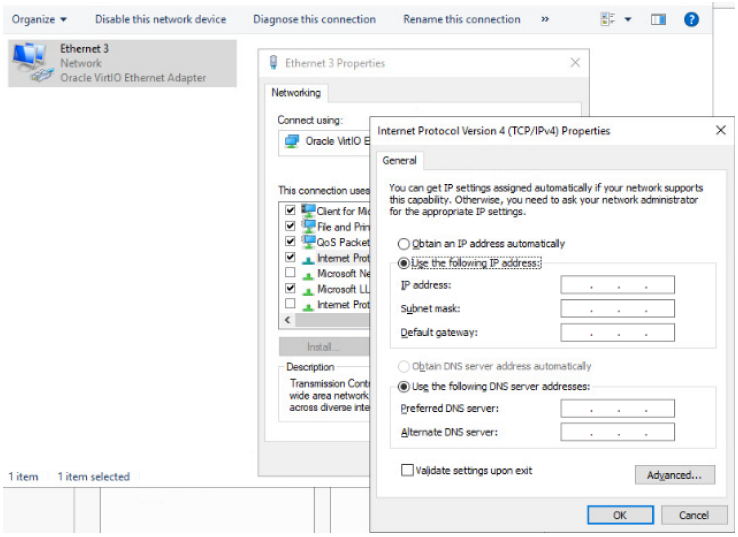
When asked, select not to restart the computer and shut it down instead via the Windows Start button.



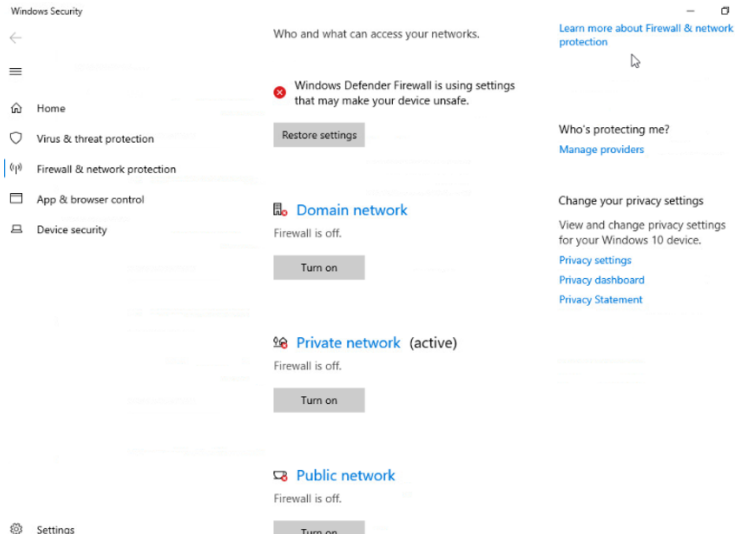


10.) Finish configuration

VNC back to your VM and configure it according to you needs. For example assign a fixed IP address.



After disabling the Windows firewall you should be able to ping the BM host IP from the KVM guest and vice versa.



11.) Remove the ISO image files

In order to be able to delete the ISO files from the /tmp directory you have to remove the files from the KVM configuration.

11.1.) odacli stop-vm -n winvm
11.2) virsh domblklist winvm

This will display a list of the configured disks:

```
--------------------------------------------------------------------------------
vda     /u05/app/sharedrepo/winvmstor/.ACFS/snaps/vm_winvm/winvm
vdb     /u05/app/sharedrepo/winvmstor/.ACFS/snaps/vdisk_winvmdisk/winvmdisk
hda     /tmp/win19.iso
hdb     /tmp/winvirtio.iso
```

11.3) virsh detach-disk winvm **hda** –config (to detach the install media)
11.4) virsh detach-disk winvm **hdb** –config (to detach the virtio driver disk)
11.5) replace the KVM xml file for in /u05/app/sharedrepo/<your kvm storage> with the on in /etc/libvirt/qemu:
cp /etc/libvirt/qemu/winvm.xml /u05/app/sharedrepo/winvmstor/.ACFS/snaps/vm_winvm/winvm.xml
11.6) rename or remove the .iso files from /tmp
11.7) odacli start-vm -n winvm
11.8) test the KVM migration to the other node
#odacli migrate-vm -n winvm -to <node_name> (replace <node_name> with node name of second node)

Create a DB System

ODA version 19.10 is the minimum software version requirement to deploy a KVM DB System on ODA, but we recommend to use ODA 19.12 or later since it offers a more complete support of DB Systems. It is recommended to always use the latest ODA software version. To deploy or upgrade your ODA with latest version, follow the Oracle documentation. The documentation is as well the source for more detailed information and restrictions that may exist with a specific release version. Always review the Oracle documentation before you start with the DB System creation. For example the 19.10 restriction that each DB System has a dedicated CPU Pool that cannot be shared was removed with 19.11, and NUMA support was implemented in ODA 19.12. The space reserved in ACFS for a DB system node is fixed to a size of 200GB. This is the space assigned for the Operation System, the Grid Infrastructure (GI) home and the database home of the DB System. The database files are stored in ASM of the bare metal ODA to which the DB System GI connects as a client.

1.) Confirm that your ODA is minimum version 19.12.0.0.0

```
# /opt/oracle/dcs/bin/odacli describe-component -v
System Version
---------------
19.12.0.0.0
```

2.) Download the DB System image (Patch 32451228: ORACLE DATABASE APPLIANCE 19.12.0.0.0 DBSYSTEM IMAGE DOWNLOAD FOR KVM) into a temporary directory to your ODA.

3.) Unzip the the file

```
# cd /tmp
#unzip p32451228_1912000_Linux-x86-64.zip
odacli-dcs-19.12.0.0.0-210827-ODAVM-19.12.0.0.zip
```

4.) Update the ODA repository

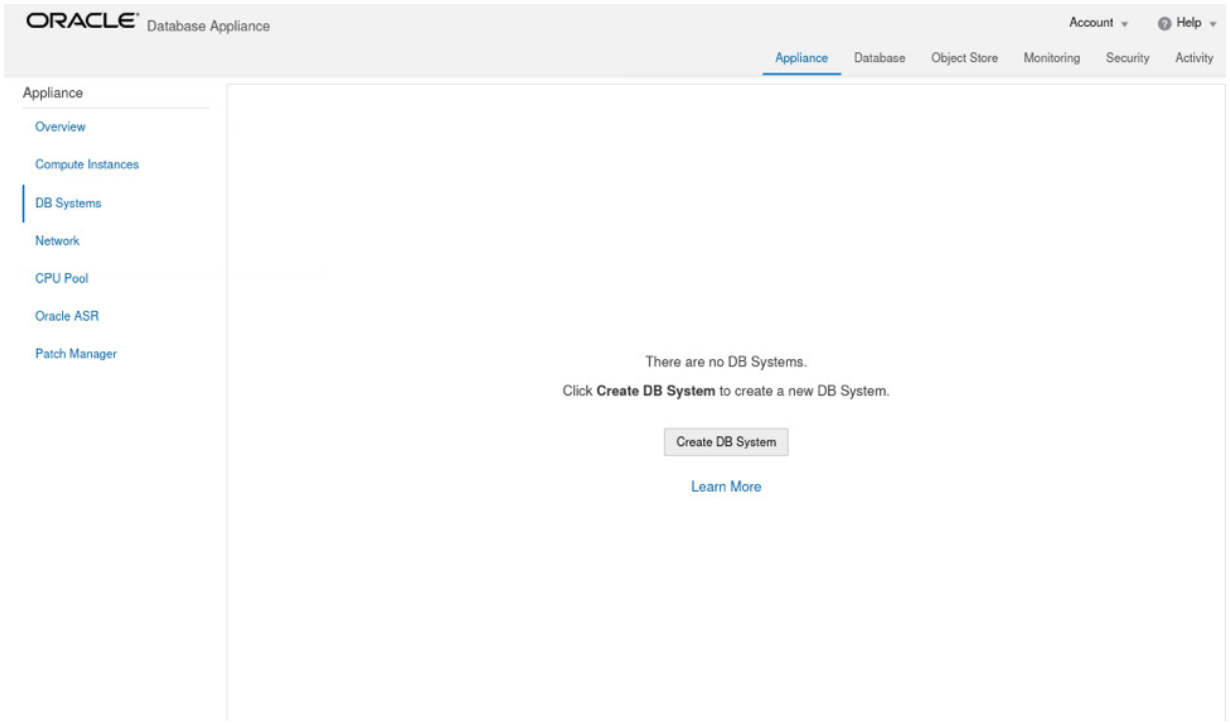
```
# /opt/oracle/dcs/bin/odacli update-repository -f /tmp/odacli-dcs-19.12.0.0.0-210827-ODAVM-
19.12.0.0.zip
{
  "jobId" : "2e392dad-587b-4ed1-87b6-995f3131cab6",
  "status" : "Created",
  "message" : "/tmp/odacli-dcs-19.12.0.0.0-210827-ODAVM-19.12.0.0.zip",
  "reports" : [ ],
  "createTimestamp" : "October 27, 2021 00:26:39 AM HDT",
  "resourceList" : [ ],
  "description" : "Repository Update",
  "updatedTime" : "October 27, 2021 00:26:39 AM HDT"
```

5.) Confirm the DB System image is correctly registered:

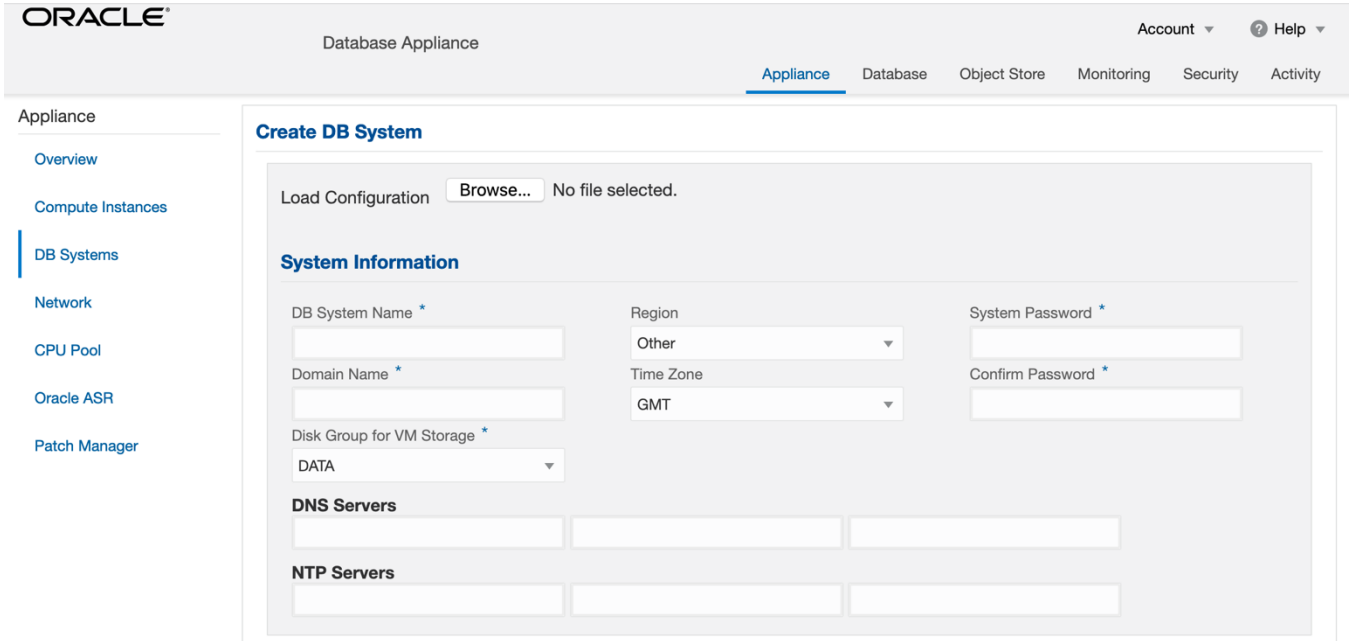
```
# /opt/oracle/dcs/bin/odacli describe-dbsystem-image
DB System Image details
--------------------------------------------------------------------------------
Component Name         Supported Versions    Available Versions
---------------------  --------------------  --------------------

DBVM                   19.12.0.0.0           19.12.0.0.0

GI                     19.12.0.0.210720      19.12.0.0.210720
                       21.3.0.0.210720       not-available

DB                     19.12.0.0.210720      19.12.0.0.210720
                       21.3.0.0.210720       not-available
```

6.) Create the DB System

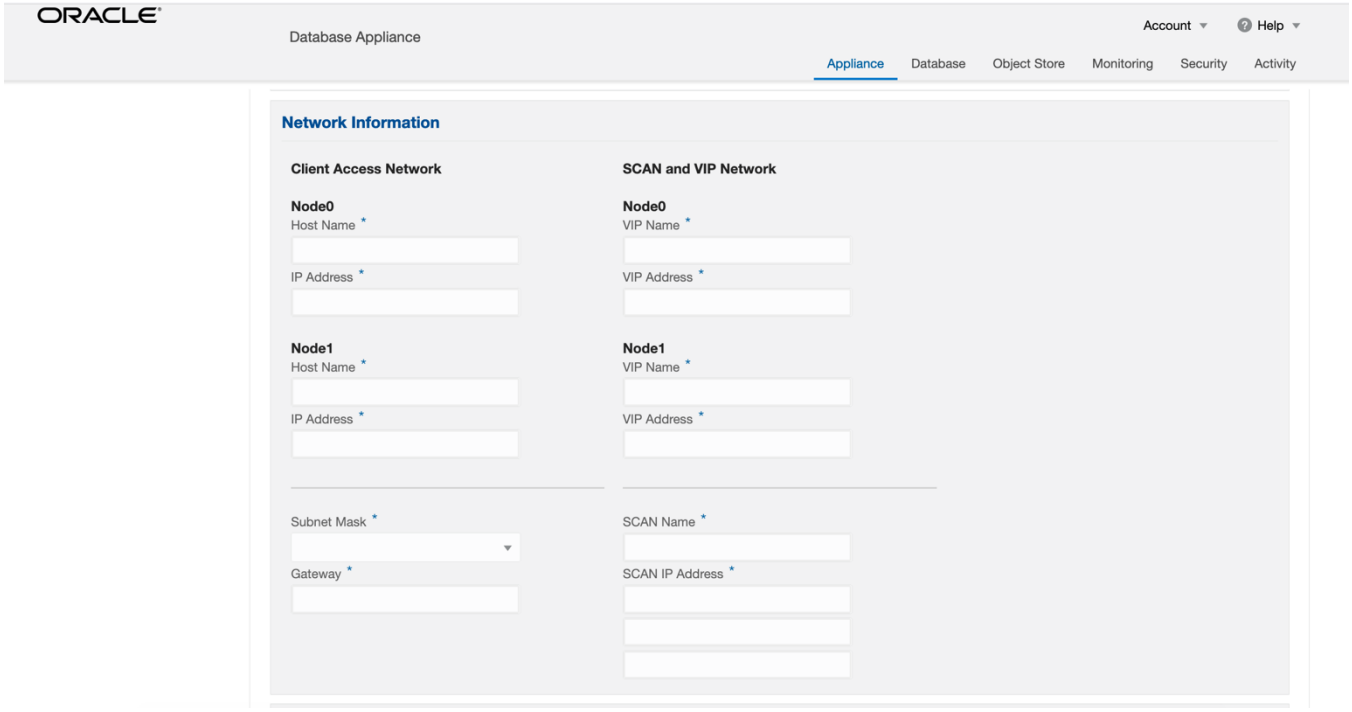
Decide on the database shape according to your Database Class (OLTP, DSS or IMDB) resource requirements and available licenses. Keep in mind that the 'shape' parameter defines the number of cores you assign to the DB System. For a complete list of DB shapes please review the ODA documentation. (The table for OLTP Database Shapes X8-2 can be found in Appendix B). You can use the BUI to create a DB System or you can create a json file with the details and kick off the creation from the CLI. In BUI go to Appliance -> DB Systems and select the 'Create DB System' button. In case you need to assign for example more cores later, you can change this after creation.



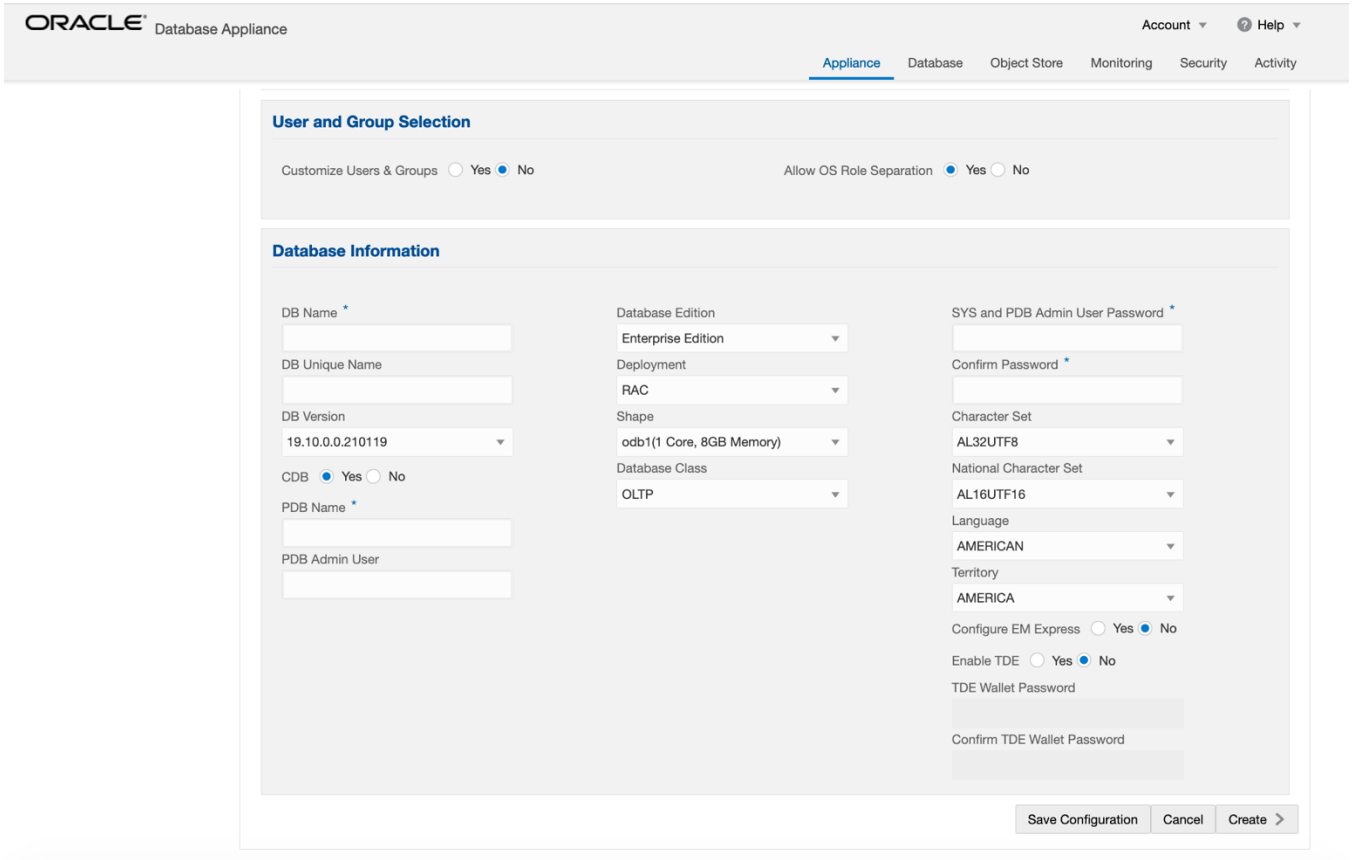
The information required to build a DB System is partitioned in three sections: 'System Information'



The middle section covers the network information.



In next section you can customize users and groups and decide if you want to use OS role separation. You have to use the same setting here as in the BM Grid Infrastructure installation. If in your BM you have a user 'grid and a user 'oracle', you are using role separation. In the lowest section provide the information about the database you would like to create.



If you prefer the CLI, create a prov.json file (an HA example json file in Appendix A) and run the create-dbsystem command.

```
# odacli create-dbsystem -p /tmp/prov.json
Job details
----------------------------------------------------------------
 ID:  b8dc3738-e542-44fd-bdb2-fd06456d0e17
Description:  DB System mydbsystem creation
Status:  Created
Created:  October 13, 2021 11:19:44 PM HST
Message:
```

7.) Confirm that the DB System was created successfully

```
# odacli describe-dbsystem -n mydbsystem
DB System details
--------------------------------------------------------------------------------
                       ID:  69e49676-3400-49aa-a2a2-bcee062065e5
                     Name:  mydbsystem
                    Image:  19.12.0.0.0
                    Shape:  odb2
             Cluster name:  dbs6cd795483
                   Memory:  16.00 GB
                   Status:  CONFIGURED
                  Created:  2021-10-13 04:56:17 HDT
                  Updated:  2021-10-13 06:01:32 HDT

 CPU Pool
--------------------------
                     Name:  c6cd795483
          Number of cores:  2

                     Host:  node1
        Effective CPU set:  0-1,32-33
              Online CPUs:  0, 1, 32, 33
             Offline CPUs:  NONE

                     Host:  node2
        Effective CPU set:  0-1,32-33
              Online CPUs:  0, 1, 32, 33
             Offline CPUs:  NONE

 VM Storage
--------------------------
               Disk group:  DATA
              Volume name:  S6CD795483
            Volume device:  /dev/asm/s6cd795483-433
                     Size:  400.00 GB
              Mount Point:  /u05/app/sharedrepo/ mydbsystem
          Acc Volume name:  AS6CD795483
        Acc Volume device:  /dev/asm/as6cd795483-433
          Acc Volume size:  1.60 GB

 VMs
--------------------------
                     Host:  node1
                  VM Name:  x6cd795483
             VM Host Name:  node1n1.us.oracle.com
             Target State:  ONLINE
            Current State:  ONLINE

                     Host:  node2
                  VM Name:  y6cd795483
             VM Host Name:  node2n2.us.oracle.com
             Target State:  ONLINE
            Current State:  ONLINE

 VNetworks
--------------------------
                     Host:  node1
                  VM Name:  x6cd795483
```

```
                    Public:  xx.xx.xx.xx     / 255.255.248.0   / ens3 / BRIDGE(pubnet)
                       ASM:  192.168.17.4    / 255.255.255.128 / ens4 / BRIDGE(privasm)
VLAN(icbond0.100)
               Interconnect:  192.168.17.129  / 255.255.255.252 / ens5 / BRIDGE(privnet1)
VLAN(icbond0.101)

                      Host:  node2
                   VM Name:  y6cd795483
                    Public:  xx.xx.xx.xx     / 255.255.248.0   / ens3 / BRIDGE(pubnet)
                       ASM:  192.168.17.5    / 255.255.255.128 / ens4 / BRIDGE(privasm)
VLAN(icbond0.100)
               Interconnect:  192.168.17.130  / 255.255.255.252 / ens5 / BRIDGE(privnet1)
VLAN(icbond0.101)

 Databases
--------------------------
                      Name:  my19db
               Resource ID:  e2978cc2-3ae0-4083-aea6-3f4cd7f38073
               Unique name:  mydb19
               Database ID:  2242777418
               Domain name:  us.oracle.com
                DB Home ID:  bc1f64a9-7782-4997-8c31-918e8c92a444
                     Shape:  odb2
                   Version:  19.12.0.0.210720
                   Edition:  EE
                      Type:  RAC
                      Role:  PRIMARY
                     Class:  OLTP
                   Storage:  ASM
                Redundancy:
          Target node name:
             Character set:  AL32UTF8
         NLS character set:
                  Language:  AMERICAN
                 Territory:  AMERICA
           Console enabled:  false
              SEHA enabled:  false
      Associated networks:  Public-network
         Backup config ID:
        Level 0 Backup Day:  sunday
        Autobackup enabled:  true
               TDE enabled:  false
                  CDB type:  false
                  PDB name:
            PDB admin user:
```

8.) Test connectivity to your DB System

During the creation of the DB system you defined scan-name and database name. With this information you should be able to build your TNS connect string using the default service name which is the same as the database name. It is a best practice to create your own database services based on your HA requirements. For an initial connection test, the default service is perfectly sufficient.

Connect as well via ssh to the KVM guest nodes and verify you can connect to the database instances locally.

```
# ssh root@node1n1.us.oracle.com
# su – oracle
$ export ORACLE_HOME=/u01/app/oracle/product/19.0.0.0/dbhome_1
$ export ORACLE_SID=my19db1     (the '1' after the DB name is for instance 1 on the first RAC
node)
$ /u01/app/oracle/product/19.0.0.0/dbhome_1/bin/sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Mar 15 08:32:37 2021
Version 19.10.0.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.10.0.0.0

SQL>
```

SOLUTION IN A BOX EXAMPLES FOR ODA

JDE

JDE team is only providing an old Xen VM Template and you should not use it with KVM. To install JDE in KVM on ODA:

- Create the application KVM Windows guests
- Create an DB System for the Oracle Database
- Follow the JDE documentation steps to install JDE::

https://docs.oracle.com/cd/E84502_01/learnjde/installation.html#one-click-provisioning

EBS

Considerations for E-Business Suite 12.2 on Oracle Database Appliance:

1. The Rapid Install tool is not supported on ODA DBSystem. This prevents a direct installation of fresh E-Business Suite environment on ODA DBSystem.
2. Existing EBS databases can be migrated to ODA DBSystem. Use the following procedure to migrate E-Business Suite databases to ODA DBSystem:
    a. Create a DBSystem.
    b. Follow either of the following procedures to migrate the database from source system to ODA:
        Export/Import Process for Oracle E-Business Suite Release 12.2 Database Instances Using Oracle Database 19c (Doc ID 2554156.1)
        Using Transportable Tablespaces to Migrate Oracle E-Business Suite Release 12.2 Using Oracle Database 19c Enterprise Edition On a Multitenant Environment (Doc ID 2674405.1)
    c. If ODA DBSystem RAC database, then run Autoconfig on remaining nodes following the instructions in Step 8.2 of the note:
        Using Oracle 19c RAC Multitenant (Single PDB) with Oracle E-Business Suite Release 12.2 (Doc ID 2530665.1)
3. E-Business Suite requires one-off patches to be installed. This would conflict that future upgrades via tooling. Typically, these one-off patches should be rolled back, database home patched and the equivalent EBS patches applied again.
4. This process was tested with EBS 12.2 with 12.2 and 19.11.

E-Business Suite References:

Interoperability Notes: Oracle E-Business Suite Release 12.2 with Oracle Database 19c (Doc ID 2552181.1)
Oracle E-Business Suite Release 12.2: Consolidated List of Patches and Technology Bug Fixes (Doc ID 1594274.1)
Export/Import Process for Oracle E-Business Suite Release 12.2 Database Instances Using Oracle Database 19c(Doc ID 2554156.1)
Using Transportable Tablespaces to Migrate Oracle E-Business Suite Release 12.2 Using Oracle Database 19c Enterprise Edition On a Multitenant Environment (Doc ID 2674405.1)
Using Oracle 19c RAC Multitenant (Single PDB) with Oracle E-Business Suite Release 12.2 (Doc ID 2530665.1)

FlexCube

FLEXCUBE team does not provide any Xen VM Template. To install FLEXCBE in KVM on ODA:

- Install the application KVM guests depending on the deployment option you choose
- Follow the FLEXCUBE documentation for the installation:

https://docs.oracle.com/cd/F18207_01/install.htm

Enterprise Manager Cloud Control

As mentioned under 'Application and Database installed in BM' you could install the Enterprise manager directly into the BM if needed. As a best practice we recommend to use a KVM guest.

Prerequisites:

1. Create one Application KVM Guest with Oracle Linux
2. Create a DB System to host OEM repository database

Follow the Oracle Enterprise Manager Cloud Control 13.5 documentation to install and configure the software. No ODA specific step is required.
https://docs.oracle.com/en/enterprise-manager/cloud-control/enterprise-manager-cloud-control/13.5/index.html

The procedure was tested with Oracle Linux 8.2 on the Application KVM guest and 19.12 DB System with 19.12 RAC database.

Please also install the ODA plugin on the bare metal nodes' Enterprise Manager Agent to monitor the hardware and other ODA specific metrics:

Add-on Plug-ins Download
https://www.oracle.com/enterprise-manager/downloads/oem-v135-update-plugins-downloads.html

ODA Plug-in installation steps:
https://docs.oracle.com/en/engineered-systems/oracle-database-appliance/plugin.html

CONCEPT OF USING ODACLI KVM COMMANDS

You can use odacli commands to get information about your configured resources like CPU Pools, VM Disks, VMs and so on. You use the odacli list' command to get a list of configured resources (e.g. odacli list-cpupools, odacli list-vmstorages etc.) you use the odacli describe' command to get more information about a specific resource (e.g. odacli describe-cpupools, odacli describe-vmstorages etc.). The same way it works for 'start', 'stop', 'modify' and 'delete'. (mind that not all are commands are available for all resource types, e.g. vdisks aren't stoppable). For a full list of available commands review the ODA documentation. Below examples of the most often used with KVM:

List DB System

odacli list-dbsystems

Show the details about a DB System

odacli describe-dbsystem -n dbsystem_name

Start a DB System

odacli start-dbsystem -n dbsystem_name

Stop a DB System

odacli stop-dbsystem -n dbsystem_name

Delete a DB System

odacli delete-dbsystem -n dbsystem_name

NOTE: ensure GI and *all* databases within the DB system are properly deinstalled before running the command

List KVM guest

odacli list-vms

Show the details about a KVM guest

odacli describe-vm -n vm1

Start a KVM guest

odacli start-vm -n vm1

Stop a KVM guest

odacli stop-vm -n vm1

Delete a KVM guest

odacli delete-vm -n vm1

MIGRATING OVM GUESTS FROM ODA VP TO KVM ON ODA BM

Creating a Solution-in-a-Box doesn't necessary mean that you have to create a new Virtual Machine. You might have an existing Oracle Database Appliance running Oracle Virtual Machine (OVM) and you would like to migrate the OVM from this machine to a KVM guest. The steps are document in My Oracle Support note: "Migrating OVM guests from ODA VP to KVM on ODA BM (Doc ID 2773840.1)".

BACKUP YOUR KVM GUEST

Backup and Recovery procedures for the DB Systems are not different from the well-known database backup procedures every DBA is familiar with. For the Application VMs please review My Oracle Support note "Backup of KVM guests on ODA 19.9 BM and later(Doc ID 2779329.1)" For Database Systems review: "Database System backup on Oracle Database Appliance Release 19.10 and later (Doc ID 2784991.1)"

APPENDIX A: EXAMPLE PROV.JSON

```
INFO: ensure to review every parameter in the file and adjust to your environment and needs.
This example is for high availability DB System:

{
    "system": {
        "name": "mydbsystem",
        "shape": "odb1",
        "systemPassword": "Hallo_Welt_12",
        "timeZone": "America/Los_Angeles",
        "diskGroup": "DATA",
        "enableRoleSeparation": true,
        "customRoleSeparation": {
            "groups": [
                {
                    "name": "oinstall",
                    "id": 1001,
                    "role": "oinstall"
                },
                {
                    "name": "dbaoper",
                    "id": 1002,
                    "role": "dbaoper"
                },
                {
                    "name": "dba",
                    "id": 1003,
                    "role": "dba"
                },
                {
                    "name": "asmadmin",
                    "id": 1004,
                    "role": "asmadmin"
                },
                {
```

```
                    "name": "asmoper",
                    "id": 1005,
                    "role": "asmoper"
                },
                {
                    "name": "asmdba",
                    "id": 1006,
                    "role": "asmdba"
                }
            ],
            "users": [
                {
                    "name": "grid",
                    "id": 1000,
                    "role": "gridUser"
                },
                {
                    "name": "oracle",
                    "id": 1001,
                    "role": "oracleUser"
                }
            ]
        }
    },
    "database": {
        "name": "mydb19",
        "uniqueName": "mydb19",
        "domainName": "us.oracle.com",
        "adminPassword": "Hallo_Welt_12",
        "version": "19.10.0.0.210119",
        "edition": "EE",
        "type": "RAC",
        "dbClass": "OLTP",
        "shape": "odb1",
        "role": "PRIMARY",
        "targetNodeNumber": null,
        "enableDbConsole": false,
        "redundancy": null,
        "characterSet": {
            "characterSet": "AL32UTF8",
            "nlsCharacterset": "AL16UTF16",
            "dbTerritory": "AMERICA",
            "dbLanguage": "ENGLISH"
        },
        "rmanBackupPassword": null,
        "enableTDE": false,
        "isCdb": false
    },
    "network": {
        "domainName": "us.oracle.com",
        "ntpServers": ["xx.xx.xx.xx, xx.xx.xx.xx "],
        "dnsServers": ["xx.xx.xx.xx, xx.xx.xx.xx "],
        "nodes": [
            {
                "name": "node1",
                "ipAddress": "xx.xx.xx.xx",
                "netmask": "xx.xx.xx.xx",
                "gateway": "xx.xx.xx.xx",
                "number": 0,
                "vipName": "node1-vip",
                "vipAddress": "xx.xx.xx.xx"
            },
            {
                "name": "node2",
                "ipAddress": "xx.xx.xx.xx",
                "netmask": "xx.xx.xx.xx",
                "gateway": "xx.xx.xx.xx",
                "number": 1,
```

```
                "vipName": "node2-vip",
                "vipAddress": "xx.xx.xx.xx"
            }
        ],
        "scanName": "mysbsystem-scan",
        "scanIps": [
            " xx.xx.xx.xx ",
            " xx.xx.xx.xx "
        ]
    },
    "grid": {
        "language": "en"
    }
}
```

APPENDIX B: EXAMPLE DATABASE SHAPES

Oracle Database Appliance X8-2-HA and X8-2M OLTP Database Shapes

Shape	CPU Cores	SGA (GB)	PGA (GB)	Processes	Log File Size (GB)	Log Buffer (MB)
odb1s	1	2	1	200	4	32
odb1	1	4	2	200	4	32
odb2	2	8	4	400	4	32
odb4	4	16	8	800	4	64
odb6	6	24	12	1200	8	256
odb08	8	32	16	1600	8	256
odb10	10	40	20	2000	8	256
odb12	12	48	24	2400	16	256
odb16	16	64	32	3200	16	256
odb20	20	80	40	4000	16	256
odb24	24	96	48	4800	16	256
odb28	28	112	56	5600	16	256
odb32	32	128	64	6400	16	256

CONNECT WITH US

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

 blogs.oracle.com

 facebook.com/oracle

 twitter.com/oracle

Solution-in-a-Box: Deploying Database Systems and Application VMs on Oracle Database Appliance (Bare Metal)
November 2121
Author: Hagen Herbst, RAC Pack MAA, Cloud Innovation and Solution Engineering Team
Contributing Author: Krisztian Fekete, RAC Pack MAA, Cloud Innovation and Solution Engineering Team