

Sharded Database Deployment on Oracle® Bare Metal Cloud Services

ORACLE WHITE PAPER | DECEMBER 2017







Table of Contents

Introduction	1
Overview	1
Getting Started	2
Set Up an Oracle Cloud Services Account	2
Create an SSH key pair	2
Provision the Virtual Cloud Network (VCN)	3
Create a Compartment	3
Create a Virtual Cloud Network with Two Subnets and an Internet Gateway	3
Provision the Bare Metal Instances	4
Launch the Oracle Bare Metal Instances	5
Record the Network Configuration Information	5
Connect to each Oracle Bare Metal Instance	5
Prepare the Bare Metal Instances for Installing Oracle Software	6
Configure additional swap space for each Instance	6
Configure the NVMe devices on the database hosts (sdb3-7)	6
Create a RAID array	6
Create and mount a partition with a file system	7
Enable Host Name Resolution on each Instance	7
Perform an Automatic Setup of the Oracle Installation Prerequisite Steps on Each Instance	8
Install Additional Packages on Each Instance	8
Update All Packages to Current Versions on Each Instance	8



ORACLE®

Configure the Oracle Software Related Operating System Groups and <code>oracle</code>	
Operating System Account on Each Instance	8
Open the Default SSH and Listener Ports on Each Oracle Bare Metal Instance	9
Create Directories to Contain the Oracle Software and Oracle Installation Shiphomes	
on Each Instance	9
Set Up Oracle Sharding Host Environment Scripts on Each Instance	10
Verify System Connectivity	10
Connectivity Requirements When Using the <code>ADD SHARD</code> Deployment Method	10
Connectivity Requirements When Using the <code>CREATE SHARD</code> Deployment	
Method	11
Copy and Install the Oracle Database and Oracle GSM Software	12
Copy and unzip the Oracle software shiphomes on each Bare Metal Instance	12
Configure a VNC Server on the Shard Catalog Host and First Shard Director Host	13
Install the Oracle Database Software	13
Install the Oracle Global Service Manager (GSM) software	15
Deploy the Sharded Database Configuration	16
Create the Shard Catalog Database	16
Set Up the Oracle Sharding Management and Routing Tier	16
Deploy a System-Managed Sharded Database	17
Create a Schema for the System-Managed SDB	18
Next Steps	18
Conclusion	18

Introduction

This paper describes how to deploy a demonstration Oracle Sharded Database (SDB) on Oracle Bare Metal Cloud Services. The steps that follow are based on the Sharded Database Deployment example in the Oracle Database Administrator's Guide 12c Release 2 (12.2)¹, the Oracle Bare Metal Cloud Service Tutorial – Launching Your First Instance², and on material from the Oracle Linux 7 documentation library³. The reader is assumed to be familiar with these documents and to have general familiarity with Oracle products and services.

The sharded database example in this paper is intended for demonstration purposes only. Refer to the MAA Best Practices for Oracle Sharded Databases for more information about best practices in a production environment⁴. Oracle Cloud also provides additional Database as a Service choices including single schemas, dedicated pluggable databases, virtualized databases, and databases running on world class engineered infrastructure⁵ or as a Bare Metal Database Service⁶. These Oracle-managed Database as a Service offerings are distinct from the user-managed Oracle Sharded Database deployed on Oracle-managed Bare Metal Infrastructure as a Service Instances example described in this paper.

Overview

The basic SDB configuration in this example will be deployed on seven Oracle Bare Metal Cloud instances as follows:

- » **Shard Director 1** (host: sdb1)
- » **Shard Director 2** (host: sdb2)
- » **Primary Shard 1** (host: sdb3)
- » **Standby Shard 1** (host: sdb4)
- » **Primary Shard 2** (host: sdb5)
- » **Standby Shard 2** (host: sdb6)
- » **Shard Catalog** (host: sdb7)

Two Oracle Bare Metal Availability Domains⁷ will be used, one for the primary shardgroup and the other for the standby shardgroup. A separate network subnet⁸ will be used in each availability domain. The shard catalog database and primary shardgroup (odd numbered hosts) will use SDB_Subnet_1. The standby shardgroup (even numbered hosts) will use SDB_Subnet_2. There will be one shard director configured for each subnet. The existing

1 https://docs.oracle.com/database/122/ADMIN/shard_part.htm

2 <https://docs.us-phoenix-1.oraclecloud.com/Content/GSG/Reference/overviewworkflow.htm>

3 https://docs.oracle.com/cd/E52668_01/

4 <http://www.oracle.com/technetwork/database/features/availability/maa-best-practices-155366.html>

5 <https://cloud.oracle.com/database>

6 <https://cloud.oracle.com/bare-metal-database>

7 <https://docs.us-phoenix-1.oraclecloud.com/Content/General/Concepts/regions.htm>

8 <https://docs.us-phoenix-1.oraclecloud.com/Content/Network/Concepts/overview.htm>

local NVMe storage⁹ available on the shard database and shard catalog Bare Metal instances will be used in the example sharded database configuration.

Note that typical best practices for a production SDB deployment include implementing additional features to eliminate potential points of failure such as configuring a standby shard catalog database in the standby availability domain and configuring multiple shard directors for each availability domain.

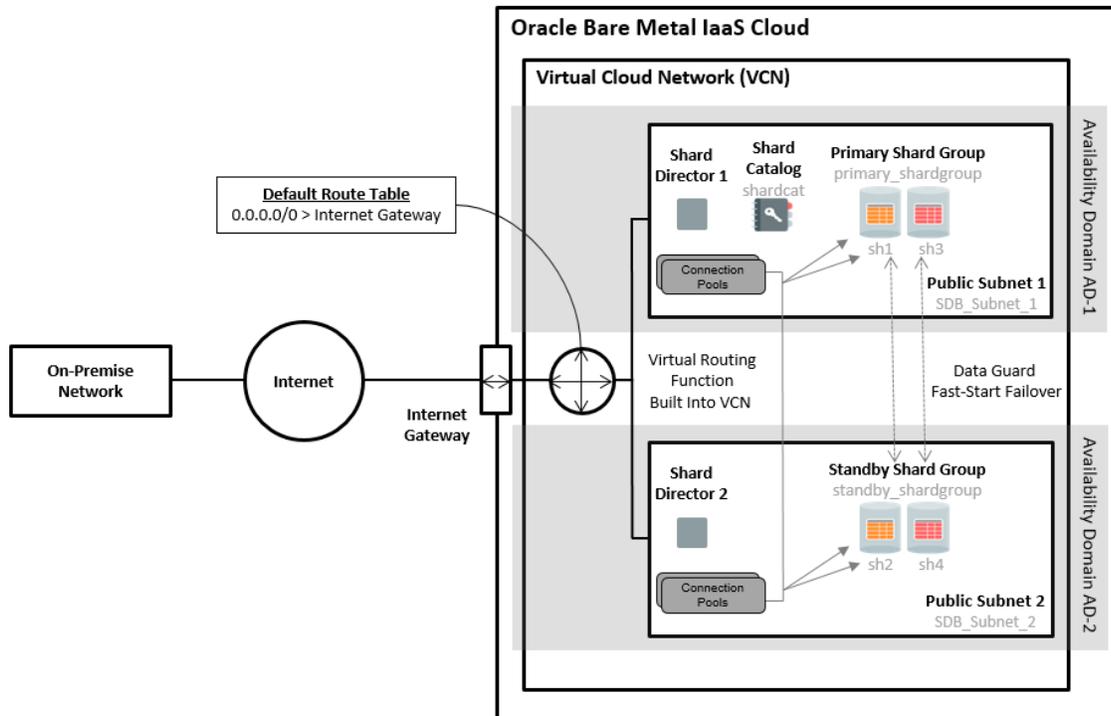


Figure 1: Example Sharded Database Configuration

Getting Started

Set Up an Oracle Cloud Services Account

- » Set up an Oracle Cloud Services account¹⁰.
- » Login to the Oracle Bare Metal Console¹¹ using a supported browser¹².

Create an SSH key pair

- » On your local system, create an SSH key pair for use with the default Oracle Bare Metal `opc` system user. In this example, the same key pair will be used on all Bare Metal Instances:

```
$ mkdir /home/oracle/shard
```

⁹ <https://docs.us-phoenix-1.oraclecloud.com/Content/Compute/Tasks/launchinginstance.htm>

¹⁰ <https://cloud.oracle.com/home>

¹¹ <https://console.us-phoenix-1.oraclecloud.com>

¹² <https://docs.us-phoenix-1.oraclecloud.com/Content/GSG/Tasks/signingin.htm>

```
$ ssh-keygen -t rsa -N "" -b 2048 -C "shard key" -f /home/oracle/shard/shard_key
```

» Change permissions on the private key file to allow only owner access:

```
$ chmod 400 /home/oracle/shard/shard_key
```

Provision the Virtual Cloud Network (VCN)

The Oracle Public Cloud resources used in this example are grouped within a compartment called SDB_Testing and consist of a Virtual Cloud Network¹³ and seven Bare Metal Cloud Instances running Oracle Linux 7.2.

Create a Compartment

A compartment¹⁴ is used to organize and control access to groups of related resources. If your administrator has already created a compartment for you to use, proceed to the next step and do not create a new compartment.

Otherwise, from the Oracle Bare Metal Console:

- » From the Console, click **Identity**, and then click **Compartments**.
- » Click **Create Compartment** and enter the following:
 - » **NAME:** SDB_Testing
 - » **DESCRIPTION:** Compartment for testing SDB deployment on Oracle Bare Metal IaaS
- » Click **Create Compartment**.

Create a Virtual Cloud Network with Two Subnets and an Internet Gateway

Ensure you are working in the compartment created above, and adjust the Classless Inter-Domain Routing (CIDR) and Availability Domain values used in the example as appropriate for the Oracle Public Cloud resources available to you.

- » Create a Virtual Cloud Network
 - » From the Console, click **Networking** and then click **Virtual Cloud Networks**
 - » Select the **SDB_Testing** compartment from the Compartment list on the left.
 - » Click **Create Virtual Cloud Network**.
 - Select **CREATE VIRTUAL CLOUD NETWORK ONLY**
 - **NAME:** SDB_Network
 - **CIDR BLOCK:** 10.1.0.0/16
 - » Click **Create**
- » Create Two Subnets
 - » From the **Virtual Cloud Networks** page, click the link for the new network (**SDB_Network**)
 - » Click **Create Subnet** and enter the following:
 - **NAME:** SDB_Subnet_1
 - **AVAILABILITY DOMAIN:** AD-1
 - **CIDR BLOCK:** 10.1.1.0/24
 - **ROUTE TABLE:** Select **Default Route Table for SDB_Network**
 - **DHCP OPTIONS:** Select **Default DHCP Options for SDB_Network**
 - **Security Lists:** Select **Default Security List for SDB_Network**
 - » Click **Create**

¹³ <https://docs.us-phoenix-1.oraclecloud.com/Content/Network/Tasks/managingVCNs.htm>

¹⁴ <https://docs.us-phoenix-1.oraclecloud.com/Content/Identity/Tasks/managingcompartments.htm>

- » Click **Create Subnet** and create the second subnet by entering the following:
 - **NAME:** SDB_Subnet_2
 - **AVAILABILITY DOMAIN:** AD-2
 - **CIDR BLOCK:** 10.1.2.0/24
 - **ROUTE TABLE:** Select **Default Route Table for SDB_Network**
 - **DHCP OPTIONS:** Select **Default DHCP Options for SDB_Network**
 - **Security Lists:** Select **Default Security List for SDB_Network**
- » Click **Create**
- » Create an Internet Gateway
 - » From the SDB_Network network details page, click **Internet Gateways** from the column on the left of the screen
 - » Click **Create Internet Gateway** and enter the following:
 - **NAME:** SDB_Gateway
 - » Click **Create**
- » Edit the Default Route Table
 - » While still on the SDB_Network network details page, click **Route Tables** from the column on the left of the screen
 - » Click **Default Route Table for SDB_Network** to display its details
 - » Click **Create Route Rule**
 - » For **Routing Rule 1** enter the following to direct all outbound network traffic not already covered by other rules in the route table to the Internet Gateway:
 - **CIDR BLOCK:** 0.0.0.0/0
 - **TARGET:** Select **SDB_Gateway** (Internet Gateway just created)
 - » Click **Create**
- » Edit the Default Security List
 - » From the SDB_Network network details page, click **Security Lists** from the column on the left of the screen
 - » Click **Default Security List for SDB_Network** to display its details
 - » Click **Edit Rules**
 - » Under **Allow Rules for Ingress**, click **+ Add Rule** and enter the following to allow incoming TCP traffic from all systems within the Virtual Cloud Network over all ports (for production environments, more restrictive security list rules typically would be configured):
 - **SOURCE CIDR:** 10.1.0.0/16
 - **IP PROTOCOL:** TCP
 - **DESTINATION PORT RANGE:** All
 - » Click **Save Security List Rules**

Provision the Bare Metal Instances

Configure seven Bare Metal instances. The two Shard Directors (sdb1, sdb2) should be on BM.Standard1.256 instances and the remaining five instances should be on instances with local NVMe storage. The **BM.HighIO1.512** instance has 12.8TB of raw NVMe storage and the **BM.DenseIO1.512** instance has 28.8TB of raw NVMe storage. Other configurations with NVMe storage also may be supported. Be sure to configure Oracle ASM or RAID protection on the NVMe devices and configure a file system before installing the database software. Name the instances sdb1 through sdb7. Configure the odd numbered instances to use the first Availability Domain (AD-1) and subnet (SDB_Subnet_1) and configure the even numbered instances to use the second Availability Domain (AD-2) and

Subnet (SDB_Subnet_2). Because instance names with non-alphanumeric characters may be incompatible with some software, use only alphanumeric characters for instance names when possible.

Launch the Oracle Bare Metal Instances

- » From the Console, click **Compute** and then click **Instances**
 - » Select the **SDB_Testing** compartment from the Compartment list on the left.
 - » Click **Launch Instance** and enter the following :
 - **NAME:** sdb1
 - **IMAGE:** Oracle-Linux-7.2-2016.09.18-0
 - **SHAPE:** BM.Standard1.36
 - **AVAILABILITY DOMAIN:** AD-1
 - **VIRTUAL CLOUD NETWORK:** SDB_Network
 - **SUBNET:** SDB_Subnet_1
 - **SSH KEYS:** Paste in contents of the public key shard_key.pub created previously.
 - » Click **Launch Instance** and wait until the status updates from **PROVISIONING...** to **RUNNING**.
- » Repeat this process to configure the six remaining instances sdb2-sdb7 using appropriate **NAME**, **SHAPE**, **AVAILABILITY DOMAIN**, and **SUBNET** values for each new instance.

Record the Network Configuration Information

Click the name link for each newly created instance to display its details and record the Availability Domain, Public IP address, and Overlay IP address using the blanks below each of the example values in the following table.

<u>Host</u>	<u>Availability Domain</u>	<u>Overlay (Private) IP</u>	<u>Public IP</u>
sdb1	AD-1	10.1.1.13	129.146.4.39
sdb2	AD-2	10.1.2.6	129.146.1.112
sdb3	AD-1	10.1.1.12	129.146.5.94
sdb4	AD-2	10.1.2.4	129.146.0.143
sdb5	AD-1	10.1.1.9	129.146.4.51
sdb6	AD-2	10.1.2.3	129.146.1.2
sdb7	AD-1	10.1.1.10	129.145.5.64

Table 1: Network Configuration Information for the Oracle Bare Metal Instances

Connect to each Oracle Bare Metal Instance

- » Create a terminal window for the first Instance.
- » Use ssh with the public IP address and previously created private key file to connect as the `opc` system user:

```
$ ssh -i /home/oracle/shard/shard_key -o ServerAliveInterval=100 opc@129.146.4.39
```

» Repeat this process for each of the remaining Instances.

Prepare the Bare Metal Instances for Installing Oracle Software

Configure additional swap space for each Instance

Oracle recommends a swap space of 16Gb for systems configured with memory of 16Gb or more. Add additional swap space as needed on each instance and update `/etc/fstab` accordingly.

The memory and swap space configured on an instance can be viewed by issuing the following command:

```
$ free
```

	total	used	free	shared	buff/cache	available
Mem:	263733416	879604	262552424	17516	301388	262543508
Swap:	8387580	0	8387580			

If, as in this example, the swap space for an instance is less than the recommended amount, issue commands similar to the following to increase the swap space by adding an appropriately sized swapfile (in this case 8Gb):

```
$ sudo su -
# mkdir /swap
# dd if=/dev/zero of=/swap/swapfile bs=1024 count=8290304
# chmod 600 /swap/swapfile
# mkswap /swap/swapfile
# swapon /swap/swapfile
# exit
$ free
```

	total	used	free	shared	buff/cache	available
Mem:	263733416	881864	254026340	17516	8825212	262450144
Swap:	16677880	0	16677880			

To ensure that the new swapfile is automatically enabled at boot time, add the following entry to `/etc/fstab`:

```
/swap/swapfile          swap          swap          defaults 0 0
```

Configure the NVMe devices on the database hosts (sdb3-7)

Refer to the Oracle MAA Best Practices¹⁵ for more information on best practices for database and storage configuration (e.g. optimal storage array configuration, isolation of storage used for datafiles and logfiles, etc.). This example uses a single RAID storage array configured with a file system.

Create a RAID array

The local NVMe storage on each instance is unprotected; create a RAID array to protect against the failure of an NVMe device. For the example, four locally attached NVMe devices will be configured as a RAID 6 array¹⁶:

¹⁵ <http://www.oracle.com/technetwork/database/features/availability/maa-best-practices-155366.html>

¹⁶ Adjust these commands as appropriate for the NVMe storage provided with your configuration. For example, to configure a RAID 6 array using nine NVMe devices use:

```
$ sudo mdadm --create /dev/md0 --raid-devices=9 --level=6 /dev/nvme0n1 /dev/nvme1n1 /dev/nvme2n1 /dev/nvme3n1 /dev/nvme4n1 /dev/nvme5n1 /dev/nvme6n1 /dev/nvme7n1 /dev/nvme8n1
```

See <https://docs.us-phoenix-1.oraclecloud.com/Content/Compute/References/nvmedeviceinformation.htm?Highlight=NVMe%20storage%20RAID%20configuration> for more information about protecting data on NVMe devices.

```
$ sudo yum install mdadm -y
$ sudo mdadm --create /dev/md0 --raid-devices=4 --level=6 /dev/nvme0n1 /dev/nvme1n1
/dev/nvme2n1 /dev/nvme3n1
$ sudo mdadm --detail --scan | sudo tee -a /etc/mdadm.conf >> /dev/null
```

Create and mount a partition with a file system

For the example, a single partition with an ext4 file system mounted as `/localstorage` will be created and used for the database installation, `oradata`, and `fast_recovery_area` directories:

```
$ sudo fdisk -l /dev/md0
$ sudo parted /dev/md0
(parted) mklabel gpt
(parted) unit TB
(parted) mkpart primary 0.00TB 6.00TB
(parted) print
Model: Linux Software RAID Array (md)
Disk /dev/md0: 6.40TB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name      Flags
  1      0.00TB  6.40TB  6.40TB                primary
(parted) quit
Information: You may need to update /etc/fstab.
$ sudo mkfs.ext4 /dev/md0p1
$ sudo mkdir /localstorage
$ sudo mount /dev/md0p1 /localstorage
```

To ensure that the new file system is automatically enabled at boot time, add the following entry to `/etc/fstab`:

```
/dev/md0p1          /localstorage      ext4    defaults 0 0
```

Enable Host Name Resolution on each Instance

Ensure that the computer host name is resolvable through a Domain Name System (DNS), a network information service (NIS), or a centrally-maintained host file, such as `/etc/hosts`.

For this example, for demonstration purposes, update the local `/etc/hosts` file on each system using the previously recorded private Overlay IP addresses by adding entries similar to the following:

```
# host name resolution for SDB hosts
10.1.1.13    sdb1.localdomain sdb1
10.1.2.6    sdb2.localdomain sdb2
10.1.1.12    sdb3.localdomain sdb3
10.1.2.4    sdb4.localdomain sdb4
10.1.1.9    sdb5.localdomain sdb5
10.1.2.3    sdb6.localdomain sdb6
10.1.1.10   sdb7.localdomain sdb7
```

Perform an Automatic Setup of the Oracle Installation Prerequisite Steps on Each Instance

On each instance, install the Oracle 12c Release 2 database preinstall package:

```
$ sudo yum install oracle-database-server-12cR2-preinstall -y
```

Install Additional Packages on Each Instance

Install additional packages as appropriate to configure VNC , Java, xterm, and any other components that you usually use to run Oracle Universal Installer (OUI) and related software; for example:

```
$ sudo yum install tigervnc.x86_64 tigervnc-server.x86_64 java-1.8.0-openjdk.x86_64
xterm.x86_64 xorg-x11-xinit.x86_64 xorg-x11-server-common.x86_64 xorg-x11-utils.x86_64
```

Update All Packages to Current Versions on Each Instance

For each instance, issue the following command:

```
$ sudo yum update -y
```

Configure the Oracle Software Related Operating System Groups and `oracle` Operating System Account on Each Instance

- » The Oracle preinstall package created an `oracle` operating system account and the `oinstall` and `dba` groups; create an `oper` group by executing the following command on each Bare Metal Instance:

```
$ sudo groupadd -g 54323 oper
```

- » Add the `oper` group to the `oracle` system account as a secondary group:

```
$ sudo /usr/sbin/usermod -g oinstall -G dba,oper oracle
$ id oracle
uid=54321(oracle) gid=54321(oinstall) groups=54321(oinstall),54322(dba),54323(oper)
```

- » Create a password for the `oracle` system account (to control access to sudo privileges, etc.)

```
$ sudo passwd oracle
```

- » Use `visudo` to edit the `/etc/sudoers` file to enable sudo privileges for the `oracle` system account (for root script execution):

```
$ sudo visudo
```

Append the following lines to the end of this file so that sudo is enabled for the `oracle` system account when the password created above is supplied:

```
# User rules for oracle
oracle ALL=(ALL) ALL
```

- » Confirm that sudo is now enabled for the `oracle` system account on each Instance; for example (for `sdb1`):

```
[opc@sdb1 ~]$ sudo su oracle
[oracle@sdb1 opc]$ sudo su -
[sudo] password for oracle:
[root@sdb1 ~]# exit
[oracle@sdb1 opc]$ exit
[opc@sdb1 ~]$
```

- » Enable ssh authentication for the `oracle` system account to remove the initial dependency on the `opc` account for connections using the `oracle` account.

- » On your local system, Create an SSH key pair for use with the `oracle` system account. For the example deployment, the same key pair will be used to configure the `oracle` system account on all the Bare Metal Instances.

```
$ mkdir /home/oracle/orashard
$ ssh-keygen -t rsa -N "" -b 2048 -C "orashard" -f /home/oracle/orashard/orashard_key
```

- » On each Bare Metal Instance, perform the following steps to enable ssh authentication for the `oracle` system account (steps shown for `sdb1`).

- From the `oracle` system account, create a `.ssh` directory:

```
$ sudo su oracle
$ cd /home/oracle
$ mkdir .ssh
```

- As the `oracle` system user, create an `authorized_keys` file in the `.ssh` directory to contain the public portion of the ssh key, paste in the contents of the `orashard_key.pub` file (the public key from the SSH key pair created above), and save the new `authorized_keys` file.

```
$ cd .ssh
$ vi authorized_keys
```

- From your local system where the key was created, confirm that you can now login with ssh using the `oracle` system account:

```
$ ssh -i /home/oracle/orashard/orashard_key -o ServerAliveInterval=100
oracle@129.146.4.39
```

- » For production deployments, additional operating system accounts and groups also can be configured at this time to meet increased levels of separation of roles and responsibilities based on business needs.

Open the Default SSH and Listener Ports on Each Oracle Bare Metal Instance

Instance traffic is controlled by `iptables` on each instance in addition to the security list previously configured for the Virtual Cloud Network. Connect to each instance and enable the default listener port 1521 and the default ssh port 22 by using the following commands:

```
$ sudo firewall-cmd --zone=public --add-port=1521/tcp
$ sudo firewall-cmd --zone=public --add-port=1521/tcp --permanent
$ sudo firewall-cmd --zone=public --add-port=22/tcp
$ sudo firewall-cmd --zone=public --add-port=22/tcp --permanent
$ sudo firewall-cmd --reload
```

Create Directories to Contain the Oracle Software and Oracle Installation Shiphomes on Each Instance

- » Create a demo directory on the local storage for each Instance:

```
$ sudo mkdir -p /localstorage/demo
```

- » Change the ownership and group membership of the demo directory to `oracle:oinstall` so that the installation user can write to the volume:

```
$ sudo chown oracle:oinstall /localstorage/demo
```

- » Create the `ORACLE_BASE` directory.

On each Bare Metal Instance, using the `oracle` system account, execute the following commands:

```
$ mkdir /localstorage/demo/oracle
```

- » Create a directory for the Oracle installation shiphomes.

On each Bare Metal Instance, using the `oracle` system account, execute the following commands:

```
$ mkdir /localstorage/demo/shiphomes
```

Set Up Oracle Sharding Host Environment Scripts on Each Instance

As the `oracle` system user, create a shell script in the `/home/oracle` directory on each host system in the SDB configuration to set the expected environment variables for that system to the correct values.

For systems where database software will be installed (hosts `sdb3-sdb6` for shard databases `sh1-sh4`, respectively, and host `sdb7` for shard catalog `shardcat`), the needed commands and scripts will be similar to the following:

```
$ ssh -i /home/oracle/orashard/orashard_key -o ServerAliveInterval=100 oracle@129.146.5.108
[oracle@sdb7 ~]$ pwd
/home/oracle
[oracle@sdb7~]$ vi shardcat.sh
[oracle@sdb7 ~]$ cat shardcat.sh
export SAVEPATH=$PATH
export ORACLE_SID=shardcat
export ORACLE_BASE=/localstorage/demo/oracle
export ORACLE_HOME=/localstorage/demo/oracle/product/12.2.0/dbhome_1
export LD_LIBRARY_PATH=$ORACLE_HOME/lib
export PATH=$SAVEPATH:$ORACLE_HOME/bin
```

For systems where global service manager software will be installed (hosts `sdb1` and `sdb2` for shard directors `sharddirector1` and `sharddirector2`, respectively), the needed commands and scripts will be similar to the following:

```
$ ssh -i /home/oracle/orashard/orashard_key -o ServerAliveInterval=100 oracle@129.146.5.81
[oracle@sdb1 ~]$ pwd
/home/oracle
[oracle@sdb1~]$ vi shardedirector1.sh
[oracle@sdb1 ~]$ cat shardedirector1.sh
export SAVEPATH=$PATH
export ORACLE_BASE=/localstorage/demo/oracle
export ORACLE_HOME=/localstorage/demo/oracle/product/12.2.0/gsmhome_1
export LD_LIBRARY_PATH=$ORACLE_HOME/lib
export PATH=$SAVEPATH:$ORACLE_HOME/bin
```

Verify System Connectivity

Before proceeding further, ensure that all of the following connectivity requirements are satisfied or will be satisfied when completing the steps described in this paper. Depending on your environment, additional ports and/or security list rules might need to be configured for your shard database, shard catalog, shard director, and application/client tier systems.

Note that new shards may be deployed using either the `GDSCTL ADD SHARD` or `CREATE SHARD` commands. The example in this paper uses the `CREATE SHARD` method to deploy shards, but the connectivity prerequisites for both methods are provided for reference (for example, in the event a pre-created shard is later added to the configuration).

Connectivity Requirements When Using the `ADD SHARD` Deployment Method

1. Each and every shard must be able to reach each and every shard director's listener and ONS ports (default shard director listener port is 1522 and default ONS ports on most platforms are 6123 for the localONS and 6234 for remoteONS). These ports also must be opened to the application/client tier, all shards, the shard catalog and all other shard directors.

Shard director listener ports are used by clients to make connections to global services. Each shard uses these ports to register themselves with the listener and the shard catalog uses these ports to register the catalog service. Therefore, it is critical that these ports are accessible to all clients, all shards, and the shard catalog.

2. Each and every shard must be able to reach the shard catalog (both primary and standby) TNS listener port (default shard catalog listener port is 1521).

All shards need to connect to the shard catalog database via a database link. Therefore, the port being used by the TNS listener on the shard catalog host for incoming connections must be made available for connections from all shard hosts.

3. Similarly, the shard database TNS listener port (default shard database listener port is 1521) of each shard must be opened to all shard directors and the shard catalog.

Connectivity Requirements When Using the CREATE SHARD Deployment Method

1. Each and every shard must be able to reach each and every shard director's listener and ONS ports (default shard director listener port is 1522 and default ONS ports on most platforms are 6123 for the localONS and 6234 for remoteONS). These ports also must be opened to the application/client tier, all shards, the shard catalog and all other shard directors.

Shard director listener ports are used by clients to make connections to global services. Each shard uses these ports to register themselves with the listener and the shard catalog uses these ports to register the catalog service. Therefore, it is critical that these ports are accessible to all clients, all shards, and the shard catalog.

2. Each and every shard must be able to reach the shard catalog (both primary and standby) TNS listener port (default shard catalog listener port is 1521).

All shards need to connect to the shard catalog database via a database link. Therefore, the port being used by the TNS listener on the shard catalog host for incoming connections must be made available for connections from all shard hosts.

3. Similarly, the shard database TNS listener port (default shard database listener port is 1521) of each shard must be opened to all shard directors and the shard catalog.
4. On the shard catalog system (primary and standby), the port specified for the `-agent_port` parameter in the `create shardcatalog` command (default is port 8080) must be visible to all the shards.

An XDB service is started on the shard catalog system to listen for incoming `schagent -registerdatabase` requests from shard hosts. The port used by this service is specified by the `-agent_port` parameter when the user issues the `GDSCTL create shardcatalog` command. To determine the port currently in use, execute `lsnrctl status` on the shard catalog system and look for the port associated with the `(Presentation=HTTP) (Session=RAW)` listener endpoint.

5. The remote scheduler agent port for each shard system must be visible to the shard catalog system. Execute `schagent -status` on each shard to identify these ports.

The remote scheduler agent running on each shard host listens on a port for incoming remote scheduler agent requests. This port can be identified by executing `schagent -status` on each shard host. It is also listed (and can be specified) in the `PORT=` setting in the `schagent.conf` file (located in the `$ORACLE_HOME` or `$ORACLE_HOME/data` directory, depending on the configuration).

Copy and Install the Oracle Database and Oracle GSM Software

Copy and unzip the Oracle software shiphomes on each Bare Metal Instance

For this example, it is assumed that the needed Oracle software is maintained in a local software repository and will be copied to the `/localstorage/demo/shiphomes` directory on each Bare Metal Instance using `scp` with `ssh` authentication. The steps to do this are shown below.

- » Download Oracle Database 12.2 and Oracle Database Global Service Manager 12.2 software to the local software repository¹⁷.
- » Enable `scp` with `ssh` authentication from the local software repository host to each Bare Metal Instance.

- » On your local software repository host, check to see if an `rsa` key pair already exists:

```
$ ls ~/.ssh/id_rsa*
/home/oracle/.ssh/id_rsa /home/oracle/.ssh/id_rsa.pub
```

If an `rsa` key pair does not exist in this location, create one using the following command:

```
$ ssh-keygen -t rsa
```

- » Copy the contents of the public key in `id_rsa.pub` from the local repository system and append this as a new line at the end of the `authorized_keys` file previously created for the `oracle` system user on each Bare Metal Instance.
- » Use `scp` with `ssh` authentication to copy the needed shiphomes from the local repository to each Bare Metal Instance. Depending on the available network bandwidth, this may take some time. For slower networks, the time to copy shiphomes can be minimized by enabling `scp` with `ssh` authentication between the Bare Metal Instances and then using the high speed intra-cloud network to copy shiphomes from one Bare Metal Instance to the others.

Note that the sharded database feature is available with the Oracle Database and Oracle Global Service Management software 12c Release 2 (12.2) and later. For some releases, shiphomes may be broken into several files to facilitate ease of transfer (e.g. `database1.zip`, `database2.zip`, etc.).

The shard catalog system and each shard database system will need a copy of the Oracle Database installation shiphome and each shard director system will need a copy of the Oracle Global Service Management installation shiphome. For the example deployment configuration, the needed commands are similar to the following:

```
$ cd /home/oracle/shiphomes/oracle_12.2_latest
$ scp gsm.zip oracle@129.146.4.39:/localstorage/demo/shiphomes
gsm.zip          100% 765MB 21.9MB/s 00:35
$ scp gsm.zip oracle@129.146.1.112:/localstorage/demo/shiphomes
gsm.zip          100% 765MB 18.2MB/s 00:42
$ scp database.zip oracle@129.146.5.94:/localstorage/demo/shiphomes
database.zip     100% 3284MB 23.3MB/s 02:21
$ scp database.zip oracle@129.146.0.143:/localstorage/demo/shiphomes
database.zip     100% 3284MB 19.8MB/s 02:46
$ scp database.zip oracle@129.146.4.51:/localstorage/demo/shiphomes
database.zip     100% 3284MB 21.5MB/s 02:33
```

¹⁷ <https://edelivery.oracle.com> or <http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html>

```
$ scp database.zip oracle@129.146.1.2:/localstorage/demo/shiphomes
database.zip      100% 3284MB  23.5MB/s   02:20
$ scp database.zip oracle@129.146.5.64:/localstorage/demo/shiphomes
database.zip      100% 3284MB  20.0MB/s   02:44
```

- » For each Bare Metal Instance, navigate to the shiphomes directory, install the zip and unzip commands, and unzip the copied shiphomes. For example, on the first shard director host (sdb1), issue the following commands:

```
[oracle@sdb1 .ssh]$ cd /localstorage/demo/shiphomes
[oracle@sdb1 shiphomes]$ sudo yum install zip unzip
[sudo] password for oracle:
[oracle@sdb1 shiphomes]$ unzip gsm.zip
```

Configure a VNC Server on the Shard Catalog Host and First Shard Director Host

The tigervnc server package was previously installed on each Bare Metal Instance; the following commands can be used to configure and start a VNC server as the `oracle` system user on the shard catalog Bare Metal Instance (sdb7) and on the first shard director Bare Metal Instance (sdb1) to facilitate, respectively, the initial interactive database and gsm software installations:

- » On each of these systems, use the `vncpasswd` command to create a password for the VNC desktop. The password must be created by the user that runs the VNC server (`oracle`) and not `root`.
- » Create a service unit configuration file on each of these hosts for the VNC desktop:

```
$ sudo cp /lib/systemd/system/vncserver@.service /etc/systemd/system/vncserver-
oracle@\:10.service
```

- » Edit the service unit configuration file on each host by replacing any instances of `<USER>` with `oracle`:

```
$ sudo vi /etc/systemd/system/vncserver-oracle@\:10.service
...
ExecStart=/usr/sbin/runuser -l oracle -c "/usr/bin/vncserver %i"
PIDFile=/home/oracle/.vnc/%H%i.pid
...
```

- » As the `oracle` system user, start and enable the VNC desktop on each host using display port 10 and configure the firewall to allow access (the external port used by the VNC server to make the display available is 5900 plus the display port number, so need to open port 5910 for the example configuration):

```
$ sudo systemctl daemon-reload
$ sudo systemctl start vncserver-oracle@\:10.service
$ sudo systemctl enable vncserver-oracle@\:10.service
$ sudo firewall-cmd --zone=public --add-service=vnc-server
$ sudo firewall-cmd --zone=public --add-service=vnc-server --permanent
$ sudo firewall-cmd --zone=public --add-port=5910/tcp
$ sudo firewall-cmd --zone=public --add-port=5910/tcp --permanent
```

- » Optionally, use the steps above to configure a VNC server on every Bare Metal Instance in the sharded database configuration to facilitate future administrative actions (for example if you later might need to run DBCA or NETCA on a shard database host). Note that for the example deployment steps VNC access is only required for the shard catalog host (sdb7) and first shard director host (sdb1).

Install the Oracle Database Software

- » Install the Oracle Database software on the shard catalog host and create a silent response file for the subsequent Oracle Database software installations on the shard database hosts.

- » The previously completed steps to enable scp via ssh authentication between the local software repository system and each of the Bare Metal Instances also enable connections using tigervnc viewer and other applications via ssh authentication.
- » If necessary, install and configure the tigervnc viewer on the local software repository host¹⁸.
- » Connect to the shard catalog host as the `oracle` system user using the tigervnc viewer:

```
$ vncviewer -via oracle@129.146.5.64 localhost:10
```

- » In the VNC window, start the Oracle Database installer (for release 12.2 or later):

```
[oracle@sdb7 ~]$ cd /localstorage/demo/shiphomes/database
[oracle@sdb7 database]$ ./runInstaller
```

- » Perform a single instance database software-only installation (Oracle RAC, Oracle RAC One Node, and Oracle Restart are not supported in this release) by stepping through the Oracle Universal Installer screens and specifying the installation options noted in the Installing the Oracle Data Database Software section of the Sharded Database Deployment chapter in the *Oracle Database Administrator's Guide 12c Release 2 (12.2)*. Be sure to click **Save Response File** on the Summary screen (save the response file as `/home/oracle/db.rsp`) before clicking **Install** and to execute the `oraInstRoot.sh` and `root.sh` scripts in a separate terminal window when prompted at the end of the installation.
- » Use the silent response file to install Oracle Database software on the primary shard and standby shard hosts.
- » Copy the database software-only installation silent response file to each additional Bare Metal Instance where Oracle Database software will be installed (for this example, hosts `sdb3` through `sdb6`) by executing commands similar to the following on the local software repository host:

```
$ scp oracle@129.146.5.64:/home/oracle/db.rsp /home/oracle/shiphomes/12.2_latest
db.rsp 100% 23KB 22.6KB/s 00:01
$ scp /home/oracle/shiphomes/12.2_latest/db.rsp oracle@129.146.5.94:/home/oracle
db.rsp 100% 23KB 22.6KB/s 00:00
$ scp /home/oracle/shiphomes/12.2_latest/db.rsp oracle@129.146.0.143:/home/oracle
db.rsp 100% 23KB 22.6KB/s 00:00
$ scp /home/oracle/shiphomes/12.2_latest/db.rsp oracle@129.146.4.51:/home/oracle
db.rsp 100% 23KB 22.6KB/s 00:00
$ scp /home/oracle/shiphomes/12.2_latest/db.rsp oracle@129.146.1.2:/home/oracle
db.rsp 100% 23KB 22.6KB/s 00:00
```

- » Perform a silent Oracle Database software-only installation as the `oracle` system user using the copied response file on each additional Bare Metal Instance where Oracle Database software will be installed and execute the root scripts (using `sudo`) when prompted:

```
$ ssh -i /home/oracle/orashard/orashard_key -o ServerAliveInterval=100
oracle@129.146.5.94
[oracle@sdb3 ~]$ cd /localstorage/demo/shiphomes/database
[oracle@sdb3 database]$ ./runInstaller -silent -responseFile /home/oracle/db.rsp
...
```

As a root user, execute the following script(s):

1. `/localstorage/demo/oraInventory/oraInstRoot.sh`
2. `/localstorage/demo/oracle/product/12.2.0/dbhome_1/root.sh`

Successfully Setup Software.

```
$ sudo sh /localstorage/demo/oraInventory/oraInstRoot.sh
$ sudo sh /localstorage/demo/oracle/product/12.2.0/dbhome_1/root.sh
```

¹⁸ https://docs.oracle.com/cd/E52668_01/E54669/html/oi7-vnc-connect.html

Install the Oracle Global Service Manager (GSM) software

- » Install the Oracle Global Service Manager (GSM) software (used to configure the shard directors) on the first shard director Bare Metal Instance and create a silent response file for the subsequent Oracle GSM software installations.

- » Connect to the first shard director host (sdb1) using the tigervnc viewer:

```
$ vncviewer -via oracle@129.146.4.39 localhost:10
```

- » In the VNC window, navigate to the Global Service Manager installation directory and start the Oracle Global Service Manager installer (for release 12.2 or later):

```
[oracle@sdb1 ~]$ cd /localstorage/demo/shiphomes/gsm  
[oracle@sdb1 gsm]$ ./runInstaller
```

- » Perform a software-only installation by stepping through the Oracle Universal Installer screens. Enter the `ORACLE_BASE` and `ORACLE_HOME` values from the previously created host environment scripts for the Specify Installation Location screen and accept the default values for the Create Inventory screen. Be sure to click **Save Response File** on the Summary screen (save the response file as `/home/oracle/gsm.rsp`) before clicking **Install** and to execute the `orainstRoot.sh` and `root.sh` scripts in a separate terminal window when prompted at the end of the installation.

- » Use the silent response file to install Oracle GSM software on the second shard director system.

- » Copy the GSM installation silent response file to each additional Bare Metal Instance where Oracle GSM software will be installed (for this example, host sdb2) by executing commands similar to the following on the local software repository host (which is already configured to enable scp with ssh authentication):

```
$ scp oracle@129.146.4.39:/home/oracle/gsm.rsp  
/home/oracle/shiphomes/12.2_latest/gsm.rsp  
gsm.rsp 100% 2082 2.0KB/s 00:00  
$ scp /home/oracle/shiphomes/12.2_latest/gsm.rsp oracle@129.146.1.112:/home/oracle  
gsm.rsp 100% 2082 2.0KB/s 00:00
```

- » Perform a silent Oracle GSM software installation as the `oracle` system user using the copied response file on each additional Bare Metal Instance where Oracle GSM software will be installed and execute the root scripts (using `sudo`) when prompted:

```
$ ssh -i /home/oracle/orashard/orashard_key -o ServerAliveInterval=100  
oracle@129.146.1.112  
Last login: Thu Jun 16 19:22:18 2016 from 156.151.8.1  
[oracle@sdb2 ~]$ cd /localstorage/demo/shiphomes/gsm  
[oracle@sdb2 gsm]$ ./runInstaller -silent -responseFile /home/oracle/gsm.rsp  
Starting Oracle Universal Installer...  
Checking Temp space: must be greater than 551 MB. Actual 48918 MB Passed  
Preparing to launch Oracle Universal Installer from /tmp/OraInstall2016-06-21_07-55-  
25PM. Please wait ...  
You can find the log of this install session at:  
/localstorage/demo/oraInventory/logs/installActions2016-06-21_07-55-25PM.log  
The installation of Oracle Distributed Service and Load Management was successful.  
Please check '/localstorage/demo/oraInventory/logs/silentInstall2016-06-21_07-55-  
25PM.log' for more details.
```

As a root user, execute the following script(s):

1. `/localstorage/demo/oraInventory/orainstRoot.sh`
2. `/localstorage/demo/oracle/product/12.2.0/gsmhome_1/root.sh`

Successfully Setup Software.

```
$ sudo sh /localstorage/demo/oraInventory/orainstRoot.sh
$ sudo sh /localstorage/demo/oracle/product/12.2.0/gsmhome_1/root.sh
```

Deploy the Sharded Database Configuration

As outlined previously, the example sharded database configuration will consist of a shard catalog database, two shard directors and two pairs of primary and standby shard databases each configured with Oracle Active Data Guard.

Create the Shard Catalog Database

- » Using the `tigervnc` viewer, connect to the shard catalog database host as the `oracle` system user:

```
$ vncviewer -via oracle@129.146.5.64 localhost:10
```

- » In the VNC window, execute the previously created shard catalog database environment script:

```
[oracle@sdb7 database]$ cd /home/oracle
[oracle@sdb7 ~]$ . ./shardcat.sh
[oracle@sdb7 ~]$ env | grep ORA
ORACLE_SID=shardcat
ORACLE_BASE=/localstorage/demo/oracle
ORACLE_HOME=/localstorage/demo/oracle/product/12.2.0/dbhome_1
```

- » Follow the instructions in the section *Creating the Shard Catalog Database* in the chapter *Sharded Database Deployment* of the *Oracle Database Administrator's Guide 12c Release 2 (12.2)* to create a non-CDB Oracle Database using DBCA to contain the shard catalog.
 - » Use the `ORACLE_BASE` path from the environment script when creating the `oradata` and `fast_recovery_area` directories.
 - » Use SID `shardcat`, global database name `shardcat.localdomain`, and create a new listener `LISTENER` using the previously opened port 1512.
- » After completing the database creation steps, edit the “`etc/oratab`” file on the shard catalog host to set the restart flag for the shard catalog database to “`Y`” so that the shard catalog database will restart automatically whenever the host system is restarted. For example:

```
[oracle@sdb7 ~]$ sudo vi /etc/oratab
[sudo] password for oracle:
...
shardcat:/localstorage/demo/oracle/product/12.2.0/dbhome_1:Y
```

Set Up the Oracle Sharding Management and Routing Tier

- » Follow the instructions in the section *Setting Up the Sharding Management and Routing Tier* in the chapter *Sharded Database Deployment* of the *Oracle Database Administrator's Guide 12c Release 2 (12.2)* with the modifications noted below to enable communications between the shard catalog, shard directors, and shards.

- » For the example configuration, the shard catalog host is `sdb7`, shard catalog name is `shardcat`, shard director listener port is 1521, and remote scheduler agent registration port is 8080:

```
GDSCCTL> create shardcatalog -database sdb7:1521:shardcat -chunks 12 -user
mysdbadmin/Dbpwd_11 -sdb cust_sdb -region region1,region2 -agent_port 8080 -
agent_password <rsa_password>
```

- » After creating the shard catalog, open the remote scheduler agent registration port (`-agent_port` value above) on the shard catalog host:

```
[oracle@sdb7 ~]$ sudo firewall-cmd --zone=public --add-port=8080/tcp
```

```
[oracle@sdb7 ~]$ sudo firewall-cmd --zone=public --add-port=8080/tcp --permanent
[oracle@sdb7 ~]$ sudo firewall-cmd --reload
```

- » Open the ONS tcp ports after starting each shard director. For example, for the first shard director:

```
[oracle@sdb1 ~]$ sudo netstat -tulpn | grep tcp | grep ons
tcp        0      0 127.0.0.1:6123        0.0.0.0:*             LISTEN      69279/ons
tcp6       0      0 :::6123              :::*                   LISTEN      69279/ons
tcp6       0      0 :::6234              :::*                   LISTEN      69279/ons
[oracle@sdb1 ~]$ sudo firewall-cmd --zone=public --add-port=6123/tcp
[oracle@sdb1 ~]$ sudo firewall-cmd --zone=public --add-port=6123/tcp --permanent
[oracle@sdb1 ~]$ sudo firewall-cmd --zone=public --add-port=6234/tcp
[oracle@sdb1 ~]$ sudo firewall-cmd --zone=public --add-port=6234/tcp --permanent
[oracle@sdb1 ~]$ sudo firewall-cmd --reload
```

- » For the example, when adding operating system credentials, create one credential `os_credential` based on the `oracle` system user that can be used on all hosts:

```
GDSCTL> add credential -credential os_credential -osaccount oracle -ospassword
<oracle_os_password>
```

- » Open the port used by the remote scheduler agent on each shard host after starting the agent. For example,

```
[oracle@sdb3 ~]$ schagent -start
Scheduler agent started using port 47255
[oracle@sdb3 ~]$ sudo firewall-cmd --zone=public --add-port=47255/tcp
[oracle@sdb3 ~]$ sudo firewall-cmd --zone=public --add-port=47255/tcp --permanent
[oracle@sdb3 ~]$ sudo firewall-cmd --reload
[oracle@sdb3 ~]$ schagent -registerdatabase sdb7 8080
```

The port and other configuration information used by the scheduler agent is recorded on each host in `$ORACLE_HOME/schagent.conf` and can be modified by editing this file.

Deploy a System-Managed Sharded Database

- » Now that the basic sharded database infrastructure has been configured, a system-managed sharded database can be deployed by creating shardgroups and shards, executing the `DEPLOY` command, and creating role-based global services to facilitate client access. Follow the steps the section *Deploying a System-Managed SDB* in the chapter *Sharded Database Deployment* of the *Oracle Database Administrator's Guide 12c Release 2 (12.2)* with the modifications noted below. Be sure the global service manager is running on each of the shard director hosts before starting the deployment steps.

- » When configuring valid node checking for registration (VNCR), note that the Bare Metal Instances were configured so that the odd numbered instances (sdb1, scb3, etc.) were in Availability Domain AD-1 corresponding to region1 and the primary_shardgroup and the even numbered instances (sdb2, sdb4, etc.) were in Availability Domain AD-2 corresponding to region2 and the standby_shardgroup. The `os_credential` is the previously created operating system credential for each respective host system (for the example, the same operating system credential is used on each host). Note that it is not necessary to specify a `SYS` account password for a standby shard because the password file from its associated primary shard is automatically copied to the standby shard at the time the standby shard is created.

```
GDSCTL>add invitednode sdb3
GDSCTL>create shard -shardgroup primary_shardgroup -destination sdb3 -credential
os_credential -sys_password <sys_password>
GDSCTL>add invitednode sdb4
GDSCTL>create shard -shardgroup standby_shardgroup -destination sdb4 -credential
os_credential
GDSCTL>add invitednode sdb5
```

```
GDSCCTL>create shard -shardgroup primary_shardgroup -destination sdb5 -credential
os_credential -sys_password <sys_password>
GDSCCTL>add invitednode sdb6
GDSCCTL>create shard -shardgroup standby_shardgroup -destination sdb6 -credential
os_credential
```

- » After completing the deployment steps, edit the “etc/oratab” file on each shard host to set the restart flag for each database instance to “Y” so that the shard database instances will restart automatically whenever the host system is restarted. For example:

```
[oracle@sdb3 ~]$ sudo vi /etc/oratab
[sudo] password for oracle:
...
sh1:/localstorage/demo/oracle/product/12.2.0/dbhome_1:Y
```

Create a Schema for the System-Managed SDB

- » Follow the steps the section *Creating a Schema for a System-Managed SDB* in the chapter *Sharded Database Deployment of the Oracle Database Administrator's Guide 12c Release 2 (12.2)* with the modifications noted below. The schema created here is the schema used by the SDB Demo Application and also can be used to demonstrate basic SDB features and functionality.

- » By default, the observer for each Data Guard configuration (primary and standby shard pair) will be started on the shard director host where the shard director for the standby database region is running (sharddirector2 for region2 in this example). You can connect to the shard catalog and issue the following query to identify the host and logfile information for each observer in the sharded database configuration:

```
[oracle@sdb7 ~]$ sqlplus / as sysdba
...
SQL> select observer_state from gsmadmin_internal.broker_configs;

OBSERVER_STATE
-----
GSM server SHARDDIRECTOR2. Observer started. Log files at
'/localstorage/demo/oracle/product/12.2.0/gsmhome_1/network/admin/gsm_observer_1.log'
.
GSM server SHARDDIRECTOR2. Observer started. Log files at
'/localstorage/demo/oracle/product/12.2.0/gsmhome_1/network/admin/gsm_observer_2.log'
```

Next Steps

To learn more about Oracle Sharded Databases, download and deploy the system-managed SDB demo application. The demo application uses the SDB environment and schema you have just created to simulate the workload of an online retail store. You can download the latest version of the demo application, along with a README file that describes how to run and monitor it, from Master Note for Handling Oracle Sharding - Oracle Database 12.2 Technology (Doc ID 2226341.1)¹⁹.

Conclusion

The example in this paper has demonstrated the basic steps required to deploy a system-managed sharded database in the Oracle Bare Metal Cloud Service. Sharded databases are well suited to cloud deployment and shards may be sized to accommodate whatever cloud infrastructure is available while still achieving required service

¹⁹ <https://support.oracle.com/epmos/faces/DocumentDisplay?id=2226341.1>



levels. The Oracle Bare Metal Cloud provides a flexible platform for deploying Oracle Sharded Databases where additional shards can be easily provisioned as the database grows over time.



Oracle Corporation, World Headquarters

500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries

Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

-  blogs.oracle.com/oracle
-  facebook.com/oracle
-  twitter.com/oracle
-  oracle.com

Integrated Cloud Applications & Platform Services

Copyright © 2017, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

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

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. 0116

Sharded Database Deployment on Oracle® Bare Metal Cloud Services
December 2017

Author: Laurence Clarke

Contributing Authors: Srinagesh Battula, Virginia Beecher, Craig Carl, Robert Mcguirk, Sandeep Singla, Mahesh Thiagarajan