Oracle Database on the AWS Cloud

Using Oracle Data Guard and Oracle ASM

Quick Start Reference Deployment

Hugo Rozestraten and Santiago Cardenas
Solutions Architects, Amazon Web Services

October 2016
Last update: January 2018 (revisions)

This guide is also available in HTML format at https://docs.aws.amazon.com/quickstart/latest/oracle-database/.
# Contents

About This Guide ................................................................................................................. 4  
Quick Links ............................................................................................................................ 4  
About Quick Starts ................................................................................................................ 5  
Overview ............................................................................................................................... 5  
Oracle Database on AWS .................................................................................................... 5  
High Availability on AWS .................................................................................................... 6  
HA Scenarios with Oracle Data Guard on AWS ................................................................. 6  
Oracle Network Environment in AWS ................................................................................ 7  
Costs and Licenses ............................................................................................................... 8  
AWS Services ...................................................................................................................... 9  
Architecture ......................................................................................................................... 10  
Planning the Deployment .................................................................................................... 13  
Deployment Scenarios ....................................................................................................... 13  
Oracle Database 12c Editions ............................................................................................ 13  
Operating System for Deployment .................................................................................... 14  
Storage Management .......................................................................................................... 14  
Oracle Memory Configuration ............................................................................................ 15  
Backup Options .................................................................................................................. 16  
Deployment Steps ............................................................................................................... 17  
Step 1. Prepare Your AWS Account ..................................................................................... 17  
Step 2. Download the Oracle Database Software ............................................................... 20  
Step 3. Launch the Quick Start ............................................................................................ 22  
  Outputs ............................................................................................................................... 30  
  Files and Directories .......................................................................................................... 30  
  Step 4. Access Your Oracle Database Instance ............................................................... 30  
  Step 5. Check Your Environment .................................................................................... 32  
Working with Oracle Data Guard ....................................................................................... 34
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Viewing Your Configuration</td>
<td>34</td>
</tr>
<tr>
<td>Changing Your Configuration</td>
<td>36</td>
</tr>
<tr>
<td>Switching to the Standby Database</td>
<td>37</td>
</tr>
<tr>
<td>Switching Your DNS Configuration</td>
<td>39</td>
</tr>
<tr>
<td>Troubleshooting</td>
<td>39</td>
</tr>
<tr>
<td>Security</td>
<td>40</td>
</tr>
<tr>
<td>Network Security</td>
<td>40</td>
</tr>
<tr>
<td>OS Security</td>
<td>41</td>
</tr>
<tr>
<td>Security Groups</td>
<td>41</td>
</tr>
<tr>
<td>Migrating Your Data to AWS</td>
<td>42</td>
</tr>
<tr>
<td>AWS Database Migration Service</td>
<td>42</td>
</tr>
<tr>
<td>AWS Snowball</td>
<td>42</td>
</tr>
<tr>
<td>Oracle RMAN Backup and Restore</td>
<td>43</td>
</tr>
<tr>
<td>Oracle Data Pump</td>
<td>43</td>
</tr>
<tr>
<td>Backing Up Your Data</td>
<td>43</td>
</tr>
<tr>
<td>Oracle Secure Backup</td>
<td>43</td>
</tr>
<tr>
<td>Backups to Disk and Amazon S3</td>
<td>45</td>
</tr>
<tr>
<td>AMIs and Snapshots</td>
<td>46</td>
</tr>
<tr>
<td>Additional Resources</td>
<td>47</td>
</tr>
<tr>
<td>Appendix A: Data Replication Between AWS Regions</td>
<td>50</td>
</tr>
<tr>
<td>Appendix B: Adding Disks to ASM Disk Groups</td>
<td>52</td>
</tr>
<tr>
<td>GitHub Repository</td>
<td>54</td>
</tr>
<tr>
<td>Document Revisions</td>
<td>54</td>
</tr>
</tbody>
</table>
About This Guide

This Quick Start reference deployment guide discusses architectural considerations and configuration steps for deploying Oracle Database 12c Enterprise Edition on the Amazon Web Services (AWS) Cloud. It also provides links for viewing and launching AWS CloudFormation templates that automate the deployment.

AWS provides a secure infrastructure to run your Oracle Database with an enterprise class architecture, high availability, and support for small, medium, and large databases. If you have an Oracle Database that is larger than 16 TiB, or a database that relies on features that are not currently supported by Amazon Relational Database Service (Amazon RDS), which is the AWS database platform, you can install your own Oracle Database in the AWS virtual computing environment and manage it from your data center.

This Quick Start uses AWS CloudFormation templates to automate the deployment of Oracle Database on AWS and guarantees a repeatable and secure process. This guide is for database administrators, enterprise architects, system administrators, and developers who would like to run their Oracle Database in a highly available Amazon Elastic Compute Cloud (Amazon EC2) environment, using Oracle Data Guard and Oracle Automatic Storage Management (ASM).

Quick Links

The links in this section are for your convenience. Before you launch the Quick Start, please review the architecture, configuration, network security, and other considerations discussed in this guide.

- If you have an AWS account, and you’re already familiar with AWS services and Oracle Database, you can launch the Quick Start to build the architecture shown in Figure 2 in a new or existing virtual private cloud (VPC). (See Costs and Licenses.) The deployment takes about an hour. If you’re new to AWS or to Oracle Database, please review the details and follow the step-by-step instructions provided later in this guide.

  ![Launch (for new VPC)](Launch)
  ![Launch (for existing VPC)](Launch)

- If you want to take a look under the covers, you can view the AWS CloudFormation templates that automate the deployment.

  ![View template](View template)
  ![View template](View template)
About Quick Starts

Quick Starts are automated reference deployments for key workloads on the AWS Cloud. Each Quick Start launches, configures, and runs the AWS compute, network, storage, and other services required to deploy a specific workload on AWS, using AWS best practices for security and availability.

Overview

Oracle Database on AWS

Companies around the world have relied on Oracle Database for their data for many years. The business requirements that make Oracle Database a popular choice include the need for a low recovery time objective (RTO) and recovery point objective (RPO) in case of a failure or disaster, to guarantee minimum business impact and to ensure customer confidence.

Amazon Relational Database Service (Amazon RDS) for Oracle Database is the easiest way to set up, operate, and scale a highly available Oracle Database in the cloud. You can deploy multiple editions of Oracle Database, including Enterprise Edition, Standard Edition, Standard Edition 1, and Standard Edition 2, with the Bring Your Own License (BYOL) model. Amazon RDS backs up your database automatically and also applies patches within the same patchset release.

Although Amazon RDS supports most Oracle Database use cases, it restricts access to some system procedures and tables that require advanced privileges. In some cases, you might want to install Oracle Database without restricting any features, privileges, storage sizes, and IOPS. For example, you might want to use over 16 TiB of storage and over 30,000 IOPS. For these higher workloads, we recommend that you install Oracle Database in an Amazon Elastic Compute Cloud (Amazon EC2) virtual computing environment.

This Quick Start provides a reliable and automated way to install and configure your Oracle Database Enterprise Edition on Amazon EC2 with high availability. The Quick Start templates install the Oracle ASM storage management layer with the Oracle Grid Infrastructure server, and Oracle Data Guard for database setup and replication using physical standby databases.

Oracle ASM is Oracle’s storage management solution that provides features such as disk striping, two-way and three-way mirroring, dynamic disk reconfiguration, and file management. Oracle Grid Infrastructure provides server support for Oracle Database and
Oracle ASM. Oracle Data Guard provides a GUI and command-line interface for managing your primary and secondary databases. For more information about these components, see the Oracle documentation.

High Availability on AWS

The AWS Cloud infrastructure is global, and is built around regions and Availability Zones. A region is a separate geographic area where you can place AWS instances and data. For a current map of AWS Regions, see AWS Global Infrastructure on the AWS website. When you launch this Quick Start, you can choose the AWS Region where you would like to deploy your Oracle Database.

Each region includes multiple Availability Zones, which are isolated locations with one or more discrete data centers—each with redundant power, networking and connectivity, housed in separate facilities. When you’re running databases on AWS, you can benefit significantly from Availability Zones, because they are connected to one another with fast, private, fiber-optic networking, providing automatic failover without interruption.

Oracle Database high availability (HA) on AWS relies on AWS Availability Zones. The primary database and the standby database are placed in different Availability Zones, so if your primary database fails, your standby database can take over.

HA Scenarios with Oracle Data Guard on AWS

Oracle Data Guard is a feature of Oracle Database Enterprise Edition that provides a set of tools to manage one or more Oracle standby databases for high availability and disaster recovery. To create an Oracle standby database, you replicate the Oracle primary database to a secondary machine by applying its online or archived redo logs.

When the standby database is set up, any changes to the primary database are replicated to the standby database, ensuring that the contents of the two databases are in sync.

The following table describes the replication methods associated with Oracle Data Guard protection modes.

<table>
<thead>
<tr>
<th>Protection mode</th>
<th>Replication</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum performance</td>
<td>Asynchronous</td>
<td>Primary database performance is not affected by any delays writing redo data to the standby database.</td>
</tr>
<tr>
<td>(default)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximum availability</td>
<td>Synchronous</td>
<td>Commit occurs when all redo data needed to recover transactions has been written to the</td>
</tr>
</tbody>
</table>
You can set up an Oracle primary and standby relationship between two Oracle virtual machines in different Availability Zones in the same AWS Region for synchronous or asynchronous replication, because they are connected with high-speed links. Alternatively, you can set up asynchronous replication between primary and standby databases in different AWS Regions. This Quick Start automates the first option: replication between different Availability Zones in the same region, as discussed in detail in the Architecture section. If you’re interested in the replication between regions, you can create an Amazon EC2 AMI from your standby database instance and copy it to another region. For more information about this alternate scenario, see Appendix A.

### Oracle Network Environment in AWS

To ensure a high level of security, AWS provides the Amazon Virtual Private Cloud (Amazon VPC) service, which lets you provision logically isolated sections of the AWS Cloud where you can launch AWS resources in a virtual network that you define. When you create a VPC, you specify its size in the form of a Classless Inter-Domain Routing (CIDR) block. You can define a VPC as large as a /16 CIDR block or as small as a /28 CIDR block, depending on the number of subnets and IP addresses you would like to use. For more information about VPCs and subnets, see the AWS documentation.

We recommend that you create databases in private subnets, and use NAT gateways for internet access. This will ensure that your databases are not exposed to the internet, but can be updated via download packages. Database subnets should be accessible from your VPN (or AWS Direct Connect connection, if you have one), and from your application subnets. This Quick Start automatically sets up a VPC based on AWS best practices. It also sets up public and private subnets, NAT gateways, and bastion hosts in an Auto Scaling group to access your database instances.
If you want to establish a secure connection between your on-premises data center and your VPC, you can use a VPN connection or a dedicated network connection (AWS Direct Connect).

**Figure 1: Oracle Database network planning**

Costs and Licenses

You are responsible for the cost of the AWS services used while running this Quick Start reference deployment. There is no additional cost for using the Quick Start. The AWS CloudFormation template for this Quick Start includes configuration parameters that you can customize. Some of these settings, such as instance type, will affect the cost of deployment. For cost estimates, see the pricing pages for each AWS service you will be
using in this Quick Start for full details. For information about instance pricing, see the Amazon EC2 Pricing page.

This deployment uses a Bring Your Own License (BYOL) model for Oracle Database Enterprise Edition. You must already own licenses for Oracle Database Enterprise Edition, and you must have access to the Oracle Technology Network to download the software.

This Quick Start launches the Amazon Machine Image (AMI) for the Red Hat Enterprise Linux (RHEL) or Oracle Linux operating system.

AWS Services

The core AWS components used by this Quick Start include the following AWS services. (If you are new to AWS, see the Getting Started section of the AWS documentation.)

- **Amazon VPC** – The Amazon Virtual Private Cloud (Amazon VPC) service lets you provision a private, isolated section of the AWS Cloud where you can launch AWS services and other resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.

- **Amazon EBS** – Amazon Elastic Block Store (Amazon EBS) provides persistent block-level storage volumes for use with Amazon EC2 instances in the AWS Cloud. Each Amazon EBS volume is automatically replicated within its Availability Zone to protect you from component failure, offering high availability and durability. Amazon EBS volumes provide the consistent and low-latency performance needed to run your workloads.

- **Amazon EC2** – The Amazon Elastic Compute Cloud (Amazon EC2) service enables you to launch virtual machine instances with a variety of operating systems. You can choose from existing AMIs or import your own virtual machine images.

- **Amazon S3** – Amazon Simple Storage Service (Amazon S3) provides secure, durable, highly scalable cloud storage. It enables you to store and retrieve data at any time, from anywhere on the web.

- **Amazon Route 53** – Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service.

- **Auto Scaling** – Auto Scaling helps you maintain high availability for your application and enables you to scale up and down according to conditions that you define.
• **CloudWatch Logs** – Amazon CloudWatch is a monitoring service for AWS Cloud resources and the applications you run on AWS. You can use Amazon CloudWatch Logs to monitor, store, and access your log files from EC2 instances, AWS CloudTrail, and other sources.

**Architecture**

Deploying this Quick Start for a new VPC with the **default parameters** builds a highly available Oracle Database environment in the AWS Cloud with data replication between Availability Zones. Figure 2 illustrates this environment.
Figure 2: Quick Start architecture for Oracle Database high availability
The Quick Start deploys the Oracle primary database (using the preconfigured, general-purpose starter database from Oracle) on an Amazon EC2 instance in the first Availability Zone. It then sets up a second EC2 instance in a second Availability Zone, copies the primary database to the second instance by using the `DUPLICATE` command, and configures Oracle Data Guard. (The template that deploys the Quick Start into an existing VPC skips the tasks marked by asterisks.)

- A highly available architecture that spans two Availability Zones.*
- A VPC configured with public and private subnets according to AWS best practices, to provide you with your own virtual network on AWS.*
- An internet gateway to allow access to the internet. This gateway is used by the bastion hosts to send and receive traffic.*
- Managed NAT gateways to allow outbound internet access for resources in the private subnets. Database instances use this layer to securely download Linux packages required for Oracle installation.*
- A Linux bastion host in each public subnet with an Elastic IP address to allow inbound Secure Shell (SSH) access to EC2 instances in public and private subnets.*
- Three security groups for fine-grained inbound access control from the bastion host, between the database instances, and for application access to the database.
- AWS Command Line Interface (AWS CLI) and an instance role for installation bucket access.
- The Amazon Route 53 Domain Name System (DNS) web service to provide independency from using IP addresses or server host names. This is useful in failover or switchover scenarios.
- A set of solid state drive (SSD) disks—six for ASM data (DATA) disk groups, and three for recovery (RECO) disk groups—which can be set to Provisioned IOPS (io1) volumes.

The Quick Start can also integrate with an existing Amazon S3 bucket, which helps you store backups and archive logs in a very cost-effective way. You can also use Oracle Secure Backup library commands to interact directly with Oracle Recovery Manager (RMAN).
Planning the Deployment

Before you deploy Oracle Database on AWS, please review the following sections for more information about the Oracle Database installation and deployment options.

Deployment Scenarios

This Quick Start provides two deployment options:

- **Deploy Oracle Database into a new VPC** (end-to-end deployment). This option builds a new AWS environment consisting of the VPC, subnets, NAT gateways, security groups, bastion hosts, and other infrastructure components, and then deploys Oracle Database into that new VPC.

- **Deploy Oracle Database into an existing VPC**. This option provisions Oracle Database in your existing AWS infrastructure.

The Quick Start provides separate templates for these options. It also lets you configure additional settings such as CIDR blocks and instance types, and install optional Oracle products, as discussed later in this guide.

Oracle Database 12c Editions

This Quick Start requires Oracle Database 12c Enterprise Edition, which is the only edition that includes Oracle Data Guard.

In Oracle Database 12c Enterprise Edition, you can choose to use the standard Linux file system (e.g., ext4) or Automatic Storage Management (ASM) to store your data files, redo logs, and archived logs. This Quick Start deploys an ASM infrastructure, which provides better performance and file management than the standard Linux file system. Oracle ASM supports automatic file management and data striping across multiple EBS volumes, providing high performance and an easy way to scale your storage area.
Operating System for Deployment

You can use Linux or Microsoft Windows for your Oracle Database installation on AWS. This Quick Start deployment uses Linux. The Quick Start supports Red Hat Enterprise Linux (RHEL) version 7.2 and Oracle Linux 7.3.

Storage Management

This Quick Start uses Oracle ASM as the storage layer of the virtual machine, because ASM helps you add and reduce capacity easily. ASM’s striping feature, which spreads data across multiple disks in 1-MiB blocks, also supports distributed I/O across all the EBS volumes.
This Quick Start gives you a choice of General Purpose SSD (gp2) volumes or Provisioned IOPS SSD (io1) volumes.

Setting an ASM disk group to several gp2 volumes provides a burst capability per EBS volume of up to 3,000 IOPS, which means 18,000 IOPS for a six-volume DATA disk group and 9,000 IOPS for a three-volume RECO disk group, even with disks as small as 1 GiB.

For sustained IOPS, each 1-GiB gp2 volume delivers 3 IOPS until the limit of 48,000 IOPS is reached. So with our six-volume data disks, the maximum I/O throughput is achieved with 2,666 GiB volumes (x 6 = 16,000 GiB disk group) or higher volumes.

For sustained IOPS that is higher than 3 IOPS/GiB, you can use io1 volumes. This option typically addresses the needs of critical business applications that require sustained IOPS performance or more than 10,000 IOPS or 160 MiB/s of throughput per volume, and can then achieve 320 MiB/s per volume.

If you choose Provisioned IOPS (io1) volumes, you will need to provide the amount of IOPS desired for both DATA and RECO volumes.

ASM is part of Oracle Grid Infrastructure, which is provided as a standalone download, so this Quick Start installs Oracle Grid Infrastructure followed by Oracle Database 12c.

**Oracle Memory Configuration**

The Quick Start maps each instance type to an Oracle SGA size in megabytes (60% of available RAM in the instance) and SHMMAX kernel, using 75% of the memory available to allow further SGA growth. For more information, see the Oracle support document 1529864.1 (access to the document requires an Oracle account). The Quick Start provides default settings that you can revise; see the Mappings section of the AWS CloudFormation template.

```json
"Mappings": {
  "AWSInstanceType2Value": {
    "t2.medium": {
      "sga": "2458",
      "shmall": "838861",
      "shmmmax": "3221225472"
    },
    ...
    "t2.large": {
      "sga": "4915",
```
Backup Options

The Amazon S3 service provides an easy way to handle Oracle Database backups. AWS provides large network bandwidth from the EC2 instances deployed by the Quick Start to the S3 buckets.

Oracle also provides a product called Oracle Secure Backup (OSB) Cloud Module. This product is not included in Oracle Enterprise Edition and requires a separate license with Oracle. This Quick Start gives you the option to install the OSB Cloud Module. You can use the parameters in the Oracle Secure Backup Cloud Module section of the template to configure the settings for this module, including the name of the S3 bucket for storing your Oracle Database backups, your Oracle Technology Network (OTN) account and password, and an access key and secret key for accessing the bucket. See step 3 in the deployment steps for details. To register for an OTN account, see the Oracle documentation.

For more information about Oracle Database backup options after deployment, see the Backing Up Your Data section later in this guide.
Deployment Steps

The procedure for an end-to-end deployment of Oracle Database on AWS consists of the following steps. For detailed instructions, follow the links for each step.

- **Step 1. Prepare an AWS account**
  This involves signing up for an AWS account, choosing a region, creating a key pair, and requesting increases for account limits, if necessary.

- **Step 2. Download the Oracle Database software**
  This step involves downloading the software from Oracle and placing the files in an S3 bucket.

- **Step 3. Launch the stack**
  In this step, you’ll launch the AWS CloudFormation template into your AWS account, specify parameter values, and create the stack. The Quick Start provides separate templates for end-to-end deployment and deployment into an existing VPC.

- **Step 4. Access your Oracle Database instance to verify your deployment**
  Access the Oracle Database instance from Amazon EC2 and use Java utilities to manage your instance.

- **Step 5. Check your Oracle Database environment on AWS**
  Verify that your standby database has been set up and configured correctly and that your log files are in sync.

**Step 1. Prepare Your AWS Account**

If you already have an AWS account, skip to step 2.

1. If you don’t already have an AWS account, create one at https://aws.amazon.com by following the on-screen instructions. Part of the sign-up process involves receiving a phone call and entering a PIN using the phone keypad.

2. Use the region selector in the navigation bar to choose the AWS Region where you want to deploy Oracle Database on AWS. For more information about regions and Availability Zones, and considerations for high availability, see the Overview.
Figure 5: Choosing an AWS Region

Consider choosing a region closest to your data center or corporate network to reduce network latency between systems running on AWS and the systems and users on your corporate network.

Note Some regions do not yet support NAT gateways or the full set of instance types. We recommend that you check the availability of AWS services before you choose a region.

3. Create a key pair in your preferred region. To do this, in the navigation pane of the Amazon EC2 console, choose Key Pairs, Create Key Pair, type a name, and then choose Create.
Figure 6: Creating a key pair

Amazon EC2 uses public-key cryptography to encrypt and decrypt login information. To be able to log in to your instances, you must create a key pair. On Linux, we use the key pair to authenticate SSH login.

4. (Production deployments only) If necessary, request a service limit increase for the instance type you’re using. If you already have an existing deployment that uses this instance type, and you think you might exceed the default limit with this reference deployment, you will need to request an increase. To do this, in the AWS Support Center, choose Create Case, Service Limit Increase, EC2 instances, and then complete the fields in the limit increase form. It might take a few days for the new service limit to become effective. For more information, see Amazon EC2 Service Limits in the AWS documentation.
Figure 7: Requesting a service limit increase

Step 2. Download the Oracle Database Software

Skip to step 3 if you already have the Oracle Database and Oracle Grid Infrastructure binaries in an S3 bucket. You have the option of using Oracle Database 12c Release 2 (12.2.0.1) or Oracle Database 12c Release 1 (12.1.0.2). By default, this Quick Start uses Oracle Database 12c Release 1 (12.1.0.2).

1. Go to the Oracle Database Software Downloads webpage to download the binaries for Oracle Database 12c Release 2 (12.2.0.1) or Oracle Database 12c Release 1 (12.1.0.2).

2. Go to the Oracle Linux Downloads webpage and download the Oracle Automatic Storage Management (ASM) Libraries.

<table>
<thead>
<tr>
<th>Product or component</th>
<th>Files to download and save in your S3 bucket</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Grid Infrastructure Release 1</td>
<td>linuxamd64_12102_grid_1of2.zip</td>
</tr>
<tr>
<td></td>
<td>linuxamd64_12102_grid_2of2.zip</td>
</tr>
<tr>
<td>Product or component</td>
<td>Files to download and save in your S3 bucket</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>Oracle Database</td>
<td>linuxamd64_12102_database_1of2.zip</td>
</tr>
<tr>
<td>Enterprise Edition</td>
<td></td>
</tr>
<tr>
<td>Release 1</td>
<td>linuxamd64_12102_database_2of2.zip</td>
</tr>
<tr>
<td>Oracle Grid</td>
<td>linuxx64_12201_grid_home.zip</td>
</tr>
<tr>
<td>Infrastructure</td>
<td></td>
</tr>
<tr>
<td>Release 2</td>
<td></td>
</tr>
<tr>
<td>Oracle Database</td>
<td>linuxx64_12201_database.zip</td>
</tr>
<tr>
<td>Enterprise Edition</td>
<td></td>
</tr>
<tr>
<td>Release 2</td>
<td></td>
</tr>
<tr>
<td>Oracle Automatic</td>
<td>oracleasmlib-2.0.4-1.el6.x86_64.rpm</td>
</tr>
<tr>
<td>Storage Management</td>
<td></td>
</tr>
<tr>
<td>(ASM) Libraries</td>
<td>oracleasm-support-2.1.8-1.el6.x86_64.rpm</td>
</tr>
</tbody>
</table>

3. Set up your S3 bucket by following the instructions in the AWS documentation.
   - You must place the S3 bucket in the same account (and preferably in the same region) where you're planning to launch the Quick Start.
   - Bucket name and key prefixes may only consist of letters, numbers, and hyphens (-).

4. Place the unextracted downloads in the S3 bucket, as shown in Figure 8.
   You can place the binaries in a subfolder (subkey) of the bucket, e.g., bucket-name/path/to/binaries. When you launch the Quick Start, you must specify this complete bucket name and key prefix in the Binaries Bucket Name parameter.

Figure 8. Staging downloads in the S3 bucket
Step 3. Launch the Quick Start

In this section, we’ve provided general instructions for deploying the Quick Start templates in the AWS CloudFormation console.

1. Choose one of the following options to launch the AWS CloudFormation template into your AWS account. For help choosing an option, see the discussion of deployment scenarios earlier in this guide.

   **Option 1**
   Deploy Oracle Database into a new VPC on AWS
   ![Launch]

   **Option 2**
   Deploy Oracle Database into an existing VPC
   ![Launch]

   **Important** If you’re deploying Oracle Database into an existing VPC, make sure that your VPC has two private subnets in different Availability Zones for the database instances. These subnets require NAT gateways in their route tables, to allow the instances to download packages and software without exposing them to the internet. You’ll also need the domain name option configured in the DHCP options as explained in the Amazon VPC documentation. You’ll be prompted for your VPC settings when you launch the Quick Start.

   Each stack takes approximately one hour to complete.

   **Note** You are responsible for the cost of the AWS services used while running this Quick Start reference deployment. There is no additional cost for using this Quick Start. Prices are subject to change. For full details, see the pricing pages for each AWS service you will be using in this Quick Start.

2. Check the region that’s displayed in the upper-right corner of the navigation bar, and change it if necessary. This is where the network infrastructure for Oracle Database will be built. The template is launched in the US West (Oregon) Region by default. You must choose the same region where you placed the Oracle Database binary files in step 2.

3. On the Select Template page, keep the default URL for the AWS CloudFormation template, and then choose Next.
4. On the **Specify Details** page, change the stack name if needed. Review the parameters for the template. Provide values for the parameters that require input. For all other parameters, review the default settings and customize them as necessary. When you finish reviewing and customizing the parameters, choose **Next**.

In the following tables, parameters are listed by category and described separately for the two deployment options:

- **Parameters for deploying Oracle Database into a new VPC**
- **Parameters for deploying Oracle Database into an existing VPC**

### Option 1: Parameters for deploying Oracle Database into a new VPC

**View the template for new VPC**

#### Network Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability Zones (AvailabilityZones)</td>
<td><strong>Requires input</strong></td>
<td>The list of Availability Zones to use for the subnets in the VPC. The Quick Start uses two Availability Zones from your list and preserves the logical order you specify.</td>
</tr>
<tr>
<td>VPC CIDR (VPCCIDR)</td>
<td>10.0.0.0/16</td>
<td>CIDR block for the VPC.</td>
</tr>
<tr>
<td>Private Subnet 1 CIDR (PrivateSubnet1CIDR)</td>
<td>10.0.0.0/19</td>
<td>CIDR block for the private subnet located in Availability Zone 1.</td>
</tr>
<tr>
<td>Private Subnet 2 CIDR (PrivateSubnet2CIDR)</td>
<td>10.0.32.0/19</td>
<td>CIDR block for the private subnet located in Availability Zone 2.</td>
</tr>
<tr>
<td>Public Subnet 1 CIDR (PublicSubnet1CIDR)</td>
<td>10.0.128.0/20</td>
<td>CIDR block for the public (DMZ) subnet located in Availability Zone 1.</td>
</tr>
<tr>
<td>Public Subnet 2 CIDR (PublicSubnet2CIDR)</td>
<td>10.0.144.0/20</td>
<td>CIDR block for the public (DMZ) subnet located in Availability Zone 2.</td>
</tr>
<tr>
<td>Allowed Bastion External Access CIDR (RemoteAccessCIDR)</td>
<td><strong>Requires input</strong></td>
<td>CIDR block that’s allowed SSH external access to the bastion hosts. We recommend that you set this value to a trusted CIDR block. For example, you might want to restrict access to your corporate network.</td>
</tr>
</tbody>
</table>

#### Amazon EC2 Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key Pair Name (KeyPairName)</td>
<td><strong>Requires input</strong></td>
<td>Public/private key pair, which allows you to connect securely to your instance after it launches. When you created an AWS account, this is the key pair you created in your preferred region.</td>
</tr>
</tbody>
</table>
### Database Operating System (OracleAMIOS)

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Instance Type</td>
<td>r3.2xlarge</td>
<td>EC2 instance type for the Oracle Database instances.</td>
</tr>
<tr>
<td>Database System Private IP</td>
<td>10.0.0.5</td>
<td>IP address for private subnet 1. This must be the valid address.</td>
</tr>
<tr>
<td>Standby Database Private IP</td>
<td>10.32.5</td>
<td>IP address for private subnet 2. This must be the valid address.</td>
</tr>
</tbody>
</table>

### Oracle Database Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Version</td>
<td>12.1.0.2</td>
<td>Version of the Oracle Database software. You can choose 12.2.0.1 or 12.1.0.2.</td>
</tr>
<tr>
<td>Domain Name</td>
<td>dataguard-env</td>
<td>The name of the domain for reaching your database endpoint after setup.</td>
</tr>
<tr>
<td>Database Port Number</td>
<td>1525</td>
<td>Port number for the Oracle Database listener.</td>
</tr>
<tr>
<td>Primary Database Name</td>
<td>orcl</td>
<td>The name of your primary database. Use 3-8 characters (limited by DB_NAME). The primary instance will also be assigned this name.</td>
</tr>
<tr>
<td>Data Guard Configuration</td>
<td>Data Guard</td>
<td>Set to Single for a single database instance. Keep the default setting of Data Guard for two instances (primary and standby).</td>
</tr>
<tr>
<td>Character Set</td>
<td>AL32UTF8</td>
<td>Character set for Oracle Database.</td>
</tr>
<tr>
<td>Standby Database Name</td>
<td>orclsb</td>
<td>The name of your standby database (3-12 characters).</td>
</tr>
<tr>
<td>Database Password</td>
<td>Requires input</td>
<td>Password for accessing Oracle Database. This is a 3-30 character string and may include lowercase letters, uppercase letters, numbers, pound sign (#), underscore (_), and dollar sign ($). This password is used for the SYS, SYSTEM, and DBSNMP users of Oracle Database.</td>
</tr>
</tbody>
</table>
| ASM Password                     | Requires input           | Password for accessing Oracle Automatic Storage Management (ASM). This is a 3-30 character string and may include lowercase letters, uppercase letters, numbers, pound sign (#), underscore (_), and dollar sign ($). This
<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data Diskgroup</strong></td>
<td></td>
<td>password is used for the SYSASM, ASM Monitor, and ASM EM admin users of Oracle ASM.</td>
</tr>
<tr>
<td><strong>Data Diskgroup GiB/Disk</strong> (EBSData)</td>
<td>20</td>
<td>Gigabytes for each ASM disk (Amazon EBS) for the DATA disk group. A total of 6 volumes, with a maximum of 16,000 GiB each, is allowed. For more information, see Storage Management earlier in this guide.</td>
</tr>
<tr>
<td><strong>Data Diskgroup Volume Type</strong> (DataVolumeType)</td>
<td>gp2</td>
<td>Volume type for the Amazon EBS DATA disk group. The two options are gp2 (performance ratio of 3 IOPS : 1 GiB) or io1 (performance ratio up to 50 IOPS : 1 GiB).</td>
</tr>
<tr>
<td><strong>Data – IOPS per EBS Volume</strong> (DataIOPS)</td>
<td>600</td>
<td>IOPS for EBS volumes for io1 volume types. The total IOPS for the DATA disk group will be this value x 6. A maximum of 20,000 IOPS is allowed per volume, and the ratio is limited to 50 IOPS : 1 GiB.</td>
</tr>
<tr>
<td><strong>Recovery (Reco) Diskgroup GiB/Disk</strong> (EBSReco)</td>
<td>6</td>
<td>Gigabytes for each ASM disk (Amazon EBS) for the RECO disk group. A total of 3 volumes, with a maximum of 16,000 GiB each, is allowed. For more information, see Storage Management earlier in this guide.</td>
</tr>
<tr>
<td><strong>Reco Diskgroup Volume Type</strong> (RecoVolumeType)</td>
<td>gp2</td>
<td>Volume type for the Amazon EBS RECO disk group. The two options are gp2 (performance ratio of 3 IOPS : 1 GiB) or io1 (performance ratio up to 50 IOPS : 1 GiB).</td>
</tr>
<tr>
<td><strong>Reco – IOPS per EBS Volume</strong> (RecoIOPS)</td>
<td>200</td>
<td>IOPS for EBS volumes for io1 volume types. The total IOPS for the RECO disk group will be this value x 3. A maximum of 20,000 IOPS is allowed per volume, and the ratio is limited to 50 IOPS : 1 GiB.</td>
</tr>
<tr>
<td><strong>Binaries Bucket Name</strong> (InstallBucketName)</td>
<td>Requires input</td>
<td>The name of the S3 bucket with your Oracle binaries from step 2 in the format bucket-name/key/prefix.</td>
</tr>
</tbody>
</table>

**Oracle Secure Backup Cloud Module (Optional):**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Install Oracle Secure Backup</strong> (OSBInstall)</td>
<td>false</td>
<td>Set to true to install Oracle Secure Backup (OSB) Cloud Module. For more information, see Backup Options earlier in this guide. If this parameter is true, all fields in this parameter group are mandatory.</td>
</tr>
<tr>
<td><strong>OTN Account Email/Login</strong> (OSBOTN)</td>
<td>Optional</td>
<td>Your Oracle Technology Network (OTN) account login. For more information, see the OTN website.</td>
</tr>
<tr>
<td><strong>OTN Account Password</strong> (OSBPass)</td>
<td>Optional</td>
<td>Your Oracle Technology Network (OTN) password.</td>
</tr>
<tr>
<td><strong>S3 Bucket for Backups</strong> (OSBBucketName)</td>
<td>Optional</td>
<td>Name of your S3 bucket for storing Oracle Database backups.</td>
</tr>
</tbody>
</table>
### AWS Access Key ID (OSBKey)

**Default:** Optional  
**Description:** The AWS access key for accessing your backup bucket, e.g., XXIAIV3KL4XXIT7DXXY. For more information, see the [AWS documentation](#).

### AWS Secret Access Key (OSBSecret)

**Default:** Optional  
**Description:** The AWS secret key for accessing your backup bucket, e.g., c/ZeLyziixx3HfxPVLX910KMXEdbqks3Acz7QHj. For more information, see the [AWS documentation](#).

### Linux Bastion Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AWS Quick Start configuration:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Linux Bastion Configuration:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>AWS Quick Start configuration:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Quick Start S3 Bucket Name</strong> (QSS3BucketName)</td>
<td>aws-quickstart</td>
<td>S3 bucket where the Quick Start templates and scripts are installed. Use this parameter to specify the S3 bucket name you’ve created for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. The bucket name can include numbers, lowercase letters, uppercase letters, and hyphens (-), but should not start or end with a hyphen.</td>
</tr>
<tr>
<td><strong>Quick Start S3 Key Prefix</strong> (QSS3KeyPrefix)</td>
<td>quickstart-oracle-database/</td>
<td>The S3 key name prefix used to simulate a folder for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. This prefix can include numbers, lowercase letters, uppercase letters, hyphens (-), and forward slashes (/).</td>
</tr>
</tbody>
</table>
• **Option 2: Parameters for deploying Oracle Database into an existing VPC**

**View the template for existing VPC**

**Network Configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VPC ID (VPCID)</td>
<td>Requires input</td>
<td>ID of your existing VPC (e.g., vpc-0343606e).</td>
</tr>
<tr>
<td>Private Subnet 1 ID (PrivateSubnet1ID)</td>
<td>Requires input</td>
<td>ID of the private subnet in Availability Zone 1 in your existing VPC (e.g., subnet-a0246dcd).</td>
</tr>
<tr>
<td>Private Subnet 2 ID (PrivateSubnet2ID)</td>
<td>Requires input</td>
<td>ID of the private subnet in Availability Zone 2 in your existing VPC (e.g., subnet-b58c3d67).</td>
</tr>
<tr>
<td>Bastion Security Group ID (BastionSecurityGroupID)</td>
<td>Requires input</td>
<td>ID of the bastion security group in your existing VPC (e.g., sg-7f16e910).</td>
</tr>
</tbody>
</table>

**Amazon EC2 Configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key Pair Name (KeyPairName)</td>
<td>Requires input</td>
<td>Public/private key pair, which allows you to connect securely to your instance after it launches. When you created an AWS account, this is the key pair you created in your preferred region.</td>
</tr>
<tr>
<td>Database Operating System (OracleAMIOS)</td>
<td>Red-Hat-Enterprise-Linux-7.2-HVM</td>
<td>Operating system and version for master/worker nodes.</td>
</tr>
<tr>
<td>Database Instance Type (OracleInstanceType)</td>
<td>r3.2xlarge</td>
<td>EC2 instance type for the Oracle Database instances.</td>
</tr>
<tr>
<td>Primary Database Private IP (PrimaryIPAddress)</td>
<td>10.0.0.5</td>
<td>IP address for private subnet 1. This must be the valid address.</td>
</tr>
<tr>
<td>Standby Database Private IP (StandbyIPAddress)</td>
<td>10.0.32.5</td>
<td>IP address for private subnet 2. This must be the valid address.</td>
</tr>
</tbody>
</table>

**Oracle Database Configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Version (OracleVersion)</td>
<td>12.1.0.2</td>
<td>Version of the Oracle Database software. You can choose 12.2.0.1 or 12.1.0.2.</td>
</tr>
<tr>
<td>Domain Name (DomainName)</td>
<td>dataguard-env</td>
<td>The name of the domain for reaching your database endpoint after setup.</td>
</tr>
<tr>
<td>Parameter label (name)</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>Database Port Number (DatabasePort)</td>
<td>1525</td>
<td>Port number for the Oracle Database listener.</td>
</tr>
<tr>
<td>Primary Database Name (DatabaseName)</td>
<td>orcl</td>
<td>The name of your primary database. Use 3-8 characters (limited by DB_NAME). The primary instance will also be assigned this name.</td>
</tr>
<tr>
<td>Data Guard Configuration (DataGuardConfig)</td>
<td>Data Guard</td>
<td>Set to Single for a single database instance. Keep the default setting of Data Guard for two instances (primary and standby).</td>
</tr>
<tr>
<td>Character Set (CharacterSet)</td>
<td>AL32UTF8</td>
<td>Character set for Oracle Database.</td>
</tr>
<tr>
<td>Standby Database Name (StandbyName)</td>
<td>orclsb</td>
<td>The name of your standby database (3-12 characters).</td>
</tr>
<tr>
<td>Database Password (DatabasePass)</td>
<td>Requires input</td>
<td>Password for accessing Oracle Database. This is a 3-30 character string and may include lowercase letters, uppercase letters, numbers, pound sign (#), underscore (_), and dollar sign ($). This password is used for the SYS, SYSTEM, and DBSNMP users of Oracle Database.</td>
</tr>
<tr>
<td>ASM Password (AsmPass)</td>
<td>Requires input</td>
<td>Password for accessing Oracle Automatic Storage Management (ASM). This is a 3-30 character string and may include lowercase letters, uppercase letters, numbers, pound sign (#), underscore (_), and dollar sign ($). This password is used for the SYSASM, ASM Monitor, and ASM EM admin users of Oracle ASM.</td>
</tr>
<tr>
<td>Data Diskgroup GiB/Disk (EBSData)</td>
<td>20</td>
<td>Gigabytes for each ASM disk (Amazon EBS) for the DATA disk group. A total of 6 volumes, with a maximum of 16,000 GiB each, is allowed. For more information, see Storage Management earlier in this guide.</td>
</tr>
<tr>
<td>Data Diskgroup Volume Type (DataVolumeType)</td>
<td>gp2</td>
<td>Volume type for the Amazon EBS DATA disk group. The two options are gp2 (performance ratio of 3 IOPS : 1 GiB) or io1 (performance ratio up to 50 IOPS : 1 GiB).</td>
</tr>
<tr>
<td>Data – IOPS per EBS Volume (DataIOPS)</td>
<td>600</td>
<td>IOPS for EBS volumes for io1 volume types. The total IOPS for the DATA disk group will be this value x 6. A maximum of 20,000 IOPS is allowed per volume, and the ratio is limited to 50 IOPS : 1 GiB.</td>
</tr>
<tr>
<td>Recovery (Reco) Diskgroup GiB/Disk (EBSReco)</td>
<td>6</td>
<td>Gigabytes for each ASM disk (Amazon EBS) for the RECO disk group. A total of 3 volumes, with a maximum of 16,000 GiB each, is allowed. For more information, see Storage Management earlier in this guide.</td>
</tr>
<tr>
<td>Reco Diskgroup Volume Type (RecoVolumeType)</td>
<td>gp2</td>
<td>Volume type for the Amazon EBS RECO disk group. The two options are gp2 (performance ratio of 3 IOPS : 1 GiB) or io1 (performance ratio up to 50 IOPS : 1 GiB).</td>
</tr>
<tr>
<td>Parameter label (name)</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Reco – IOPS per EBS Volume</strong></td>
<td>200</td>
<td>IOPS for EBS volumes for <code>io1</code> volume types. The total IOPS for the RECO disk group will be this value x 3. A maximum of 20,000 IOPS is allowed per volume, and the ratio is limited to 50 IOPS : 1 GiB.</td>
</tr>
<tr>
<td><strong>Binaries Bucket Name</strong></td>
<td>Requires input</td>
<td>The name of the S3 bucket with your Oracle binaries from step 2 in the format <code>bucket-name/key/prefix</code>.</td>
</tr>
</tbody>
</table>

**Oracle Secure Backup Cloud Module (Optional):**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Install Oracle Secure Backup</td>
<td>false</td>
<td>Set to <code>true</code> to install Oracle Secure Backup (OSB) Cloud Module. For more information, see Backup Options earlier in this guide. If this parameter is <code>true</code>, all fields in this parameter group are mandatory.</td>
</tr>
<tr>
<td>OTN Account Email/Login</td>
<td>Optional</td>
<td>Your Oracle Technology Network (OTN) account login. For more information, see the OTN website.</td>
</tr>
<tr>
<td>OTN Account Password</td>
<td>Optional</td>
<td>Your Oracle Technology Network (OTN) password.</td>
</tr>
<tr>
<td>S3 Bucket for Backups</td>
<td>Optional</td>
<td>Name of your S3 bucket for storing Oracle Database backups.</td>
</tr>
<tr>
<td>AWS Access Key ID</td>
<td>Optional</td>
<td>The AWS access key for accessing your backup bucket, e.g., AKIAIV3KL4XXIT7DXXYY. For more information, see the AWS documentation.</td>
</tr>
<tr>
<td>AWS Secret Access Key</td>
<td>Optional</td>
<td>The AWS secret key for accessing your backup bucket, e.g., c/ZeLyzixxx3Hf6PVLX9iORKMEdSbkks3Ac7QHj. For more information, see the AWS documentation.</td>
</tr>
</tbody>
</table>

**AWS Quick Start configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quick Start S3 Bucket Name</td>
<td>aws-quickstart</td>
<td>S3 bucket where the Quick Start templates and scripts are installed. Use this parameter to specify the S3 bucket name you’ve created for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. The bucket name can include numbers, lowercase letters, uppercase letters, and hyphens (-), but should not start or end with a hyphen.</td>
</tr>
<tr>
<td>Quick Start S3 Key Prefix</td>
<td>quickstart-oracle-database/</td>
<td>The S3 key name prefix used to simulate a folder for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. This prefix can include numbers, lowercase letters, uppercase letters, hyphens (-), and forward slashes (/).</td>
</tr>
</tbody>
</table>
5. On the **Options** page, you can specify tags (key-value pairs) for resources in your stack and set advanced options. When you’re done, choose **Next**.

6. On the **Review** page, review and confirm the template settings. Under **Capabilities**, select the check box to acknowledge that the template will create IAM resources.

7. Choose **Create** to deploy the stack.

8. Monitor the status of the stack. When the status is **CREATE_COMPLETE**, the Oracle Database cluster is ready.

**Outputs**

You can use the information displayed in the AWS CloudFormation **Outputs** tab to access and maintain your instances.

**Files and Directories**

Installation files are downloaded from your S3 bucket to the `/u01/install` directory. Logs and some temporary configuration files are created in `/tmp`. Files exchanged between the primary database instance and the standby database instance are placed in an NFS `/shared` file system. Oracle home directories are in:

- `/u01/app/oracle/product/12c/db_1` for Oracle Database
- `/u01/app/oracle/product/12c/grid` for Oracle Grid Infrastructure

**Step 4. Access Your Oracle Database Instance**

1. Access the bastion host (created by the Quick Start template for a new VPC) or launch an EC2 instance in the public subnet, and optionally associate it with the Access Database security group (if you want access to the Oracle listener or Enterprise Manager ports). You can use SSH Agent Forwarding if the same public keys are in use.

2. From this machine, you can access the instances with SSH (port 22) or the database in the port you defined. You can use the user name “oracle” or “ec2-user” with the PEM key you defined (SSH Agent Forwarding for the private key and `orcl.dataguard-env/10.0.0.5` as the example host/address for the Oracle database hosts in the example).
If you have an X terminal such as **MobaXterm**, you can start Java utilities like DBCA and NETCA to manage your EC2 Oracle instance. Database instances are already configured with X11 Linux graphic packages.

**Figure 9: Accessing Oracle Database**
You can also access port 5500 via a browser by using system or sys users with the password you set, with https://ip:5500/em/login (for example, https://172.31.3.20:5500/em/login).

![Oracle Enterprise Manager]

Figure 10: Oracle Enterprise Manager

**Step 5. Check Your Environment**

To check whether the standby database has been set up and the logs are in sync, log in to the primary database and force a log switch. This creates an archived log that is shipped to the standby database.

```bash
# ssh -A oracle@orcl.dataguard-env
# sqlplus / as sysdba
set pages 1000 line 150
alter system switch logfile;
ALTER SESSION SET nls_date_format='DD-MON-YYYY HH24:MI:SS';
SELECT sequence#, first_time, next_time, applied FROM v$archived_log ORDER BY sequence#;
```
Then log in to the standby machine to see the applied logs generated.

```
# ssh -A oracle@orcl.dataguard-env
# sqlplus / as sysdba
set pages 1000 line 150
ALTER SESSION SET nls_date_format='DD-MON-YYYY HH24:MI:SS';
SELECT sequence#, first_time, next_time, applied FROM v$archived_log ORDER BY sequence#;
```

<table>
<thead>
<tr>
<th>SEQUENCE#</th>
<th>FIRST_TIME</th>
<th>NEXT_TIME</th>
<th>APPLIED</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>24-SEP-2016 20:38:54</td>
<td>24-SEP-2016 20:40:19</td>
<td>NO</td>
</tr>
<tr>
<td>11</td>
<td>24-SEP-2016 20:40:19</td>
<td>24-SEP-2016 20:41:02</td>
<td>NO</td>
</tr>
<tr>
<td>12</td>
<td>24-SEP-2016 20:41:02</td>
<td>24-SEP-2016 20:41:02</td>
<td>NO</td>
</tr>
<tr>
<td>13</td>
<td>24-SEP-2016 20:41:02</td>
<td>24-SEP-2016 20:58:40</td>
<td>NO</td>
</tr>
<tr>
<td>14</td>
<td>24-SEP-2016 20:58:40</td>
<td>24-SEP-2016 20:59:19</td>
<td>NO</td>
</tr>
<tr>
<td>15</td>
<td>24-SEP-2016 20:59:19</td>
<td>24-SEP-2016 20:59:48</td>
<td>NO</td>
</tr>
<tr>
<td>16</td>
<td>24-SEP-2016 20:59:48</td>
<td>24-SEP-2016 20:59:49</td>
<td>NO</td>
</tr>
<tr>
<td>17</td>
<td>24-SEP-2016 20:59:49</td>
<td>24-SEP-2016 21:03:31</td>
<td>NO</td>
</tr>
<tr>
<td>18</td>
<td>24-SEP-2016 20:59:49</td>
<td>24-SEP-2016 21:03:31</td>
<td>NO</td>
</tr>
<tr>
<td>18</td>
<td>24-SEP-2016 21:03:31</td>
<td>24-SEP-2016 21:03:31</td>
<td>NO</td>
</tr>
<tr>
<td>19</td>
<td>24-SEP-2016 21:03:31</td>
<td>24-SEP-2016 21:03:31</td>
<td>NO</td>
</tr>
<tr>
<td>19</td>
<td>24-SEP-2016 21:03:31</td>
<td>24-SEP-2016 21:03:34</td>
<td>NO</td>
</tr>
<tr>
<td>20</td>
<td>24-SEP-2016 21:03:34</td>
<td>24-SEP-2016 21:03:34</td>
<td>NO</td>
</tr>
</tbody>
</table>

14 rows selected.
Working with Oracle Data Guard

After completing the deployment, you can change Oracle Data Guard configurations and also perform a switchover.

Viewing Your Configuration

You can use the Oracle command-line interface (DGMGRL) to administer your Data Guard solution. To view your configuration, log into DGMGRL and use the `SHOW CONFIGURATION` command. See if the protection mode is MaxPerformance, which means that it has asynchronous replication in place. In this example, the primary instance is `orcl1` and the standby instance is `orcl1sb`.

![Sequence and Times Table](image-url)
For a more detailed view, you can show the database replication configurations. In the following example, notice that LogXptMode='async'.

```
# dgmgrl sys/pass@instance
show database verbose 'databasename';
```
Changing Your Configuration

If you would like to change the protection mode from maximum performance (MaxPerformance) to maximum availability (MaxAvailability) for synchronous replication, you can change it with DGMGRL, and then monitor your performance. This will set up synchronous replication from the primary database (orcl) to the standby database (orclsb) with the lowest recovery point objective (RPO) in the event of primary database failure.
For more information about performance modes, see the section HA Scenarios with Oracle Data Guard on AWS previously in this guide.

**Switching to the Standby Database**

You can use the DGMGRL SWITCHOVER command to switch from the primary database (orcl) to the secondary database (orclsb).

```bash
DGMGRL> switchover to orclsb;
Performing switchover NOW... please wait
Operation requires a connection to instance "orclsb" on database "orclsb"
Connecting to instance "orclsb"... Connected as SYSDBA.
New primary database "orclsb" is opening...
Oracle Clusterware is restarting database "orcl" ...
Switchover succeeded, new primary is "orclsb"
```

EDIT CONFIGURATION SET PROTECTION MODE AS MAXAVAILABILITY;

EDIT database databaseName set property 'LogXptMode'='sync';
EDIT database standbydatabase set property 'LogXptMode'='sync';
If you have not logged in with the SYS password, you may receive the following error.

```sql
DGMGRL> switchover to orclsb;
Performing switchover NOW, please wait...  
New primary database "orclsb" is opening... 
Oracle Clusterware is restarting database "orcl" ...
ORA-01017: invalid username/password; logon denied

Warning: You are no longer connected to ORACLE.

    shut down instance "orcl" of database "orcl"
    start up instance "orcl" of database "orcl"
```

After the primary `orcl` instance is shut down and started back up, you may see the configuration switchover.

```sql
DGMGRL> show configuration
Configuration - awsguard

| Protection Mode: MaxAvailability |
| Members:                        |
| orclsb - Primary database      |
| orcl - Physical standby database |

Fast-Start Failover: DISABLED

Configuration Status: SUCCESS  (status updated 19 seconds ago)
```

If you are logged in with the SYS password, you may fail over without the need for manual shutdown and startup.

You can also enable fast-start failover by following the instructions in the Oracle documentation.
Switching Your DNS Configuration

When you switch over to the standby database, you should also update your record set in the Amazon Route 53 private hosted zone to point to the standby database:

1. Sign in to the AWS Management Console and open the Amazon Route 53 console at https://console.aws.amazon.com/route53/.

2. Find your IP record inside your private hosted zone and change it to your standby (new primary) IP, so your application can continuously reach your primary database on the private hosted zone, and then save the record.

![Figure 11: Switching your DNS](image)

Troubleshooting

When you deploy the Quick Start, if you encounter a CREATE_FAILED error instead of the CREATE_COMPLETE status code, we recommend that you relaunch the template with Rollback on failure set to No. (This setting is under Advanced in the AWS CloudFormation console, Options page.) With this setting, the stack’s state will be retained and the instance will be left running, so you can troubleshoot the issue. (You’ll want to look
at the log files in `/var/log/cloud-init.log`, `/tmp/bootstrap.log`, and `/tmp/oracleexec.log`.)

**Important** When you set Rollback on failure to No, you’ll continue to incur AWS charges for this stack. Please make sure to delete the stack when you’ve finished troubleshooting.

For additional information, see [Troubleshooting AWS CloudFormation](https://aws.amazon.com/) on the AWS website.

## Security

When you deploy systems on the AWS Cloud, security responsibilities are shared between you and AWS. AWS operates, manages, and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the services operate. In turn, you assume responsibility and management of the guest operating system (including updates and security patches), other associated application software such as Oracle Database and Oracle Grid Infrastructure, as well as the configuration of the AWS-provided security group firewall. For more information about security on AWS, visit the [AWS Security Center](https://aws.amazon.com/security/).

### Network Security

The default network security setup of this solution follows AWS security best practices. The provisioned Oracle Database instances are deployed in private subnets and can only be accessed in three ways:

- By connecting to the bastion host instance by using an SSH terminal.
- From AWS resources (such as EC2, RDS, or other instances) that you might have in the `OracleServerAccessSecurityGroup` security group, or that you might launch using the security group. You may include your application instance in this security group.
- By including new rules in `OracleServerSecurityGroup` to allow access to your database from a known IP block CIDR; for example, you might add an inbound rule to enable the VLAN 10.50.10.0/24 in your data center to connect through a VPN or AWS Direct Connect.
OS Security

To gain root access to your instances you may use `ec2-user` or `oracle` user, and then `sudo` to root.

You have to keep the Amazon EC2 PEM key you are using secure in your environment. Also, keep in mind that AWS doesn’t store your keys, so if you lose your key, you may not be able to access your instances.

Security Groups

A security group acts as a firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance. You add rules to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time. The new rules are automatically applied to all instances that are associated with the security group.

This Quick Start creates three security groups: `OracleServerAccessSecurityGroup`, `OracleServerSecurityGroup`, and `OracleServersSecurityGroup`. After the Quick Start deployment, you are responsible for maintaining these security groups and including or excluding rules.

- `OracleServerSecurityGroup` is used to grant the bastion hosts access to port 22 of the Oracle instances.
• **OracleServersSecurityGroup** is used only for communications between database instances: primary and standby instances on database ports, SSH, and NFS ports.

• **OracleServerAccessSecurityGroup** gives EC2 instances access to your database on the port you set up for database listeners, and on port 5500 for Oracle Enterprise Manager.

### Migrating Your Data to AWS

AWS provides several services that you can use to migrate your data to your Oracle Database installation. These are described briefly in the following sections. For detailed information, see the whitepaper [Strategies for Migrating Oracle Databases to AWS](https://aws.amazon.com/whitepaper/Strategies-for-Migrating-Oracle-Databases-to-AWS/).

#### AWS Database Migration Service

AWS Database Migration Service (AWS DMS) helps you migrate your databases to AWS with virtually no downtime. All data changes to the source database that occur during the migration are continuously replicated to the target, allowing the source database to be fully operational during the migration process. After the database migration is complete, the target database remains synchronized with the source for as long as you choose, allowing you to switch the database over at a convenient time.

AWS DMS makes it easy to load your tables from your local database to your database in AWS. You can migrate your Oracle Database with multiple parallel tasks, by using an internet link (VPN connection) or a dedicated 1-Gbps or 10-Gbps connection (AWS Direct Connect).

For more information about AWS DMS, see the [AWS website](https://aws.amazon.com/database-migration-service/).

#### AWS Snowball

AWS Snowball, which is a feature of AWS Import/Export, addresses common challenges with large-scale data transfers, including high network costs, long transfer times, and security concerns. Transferring data with Snowball is simple, fast, secure, and can cost as little as one-fifth the cost of high-speed internet.

With Snowball, you don’t need to write any code or purchase any hardware to transfer your data. Create a job in the AWS Management Console, and a Snowball appliance will be automatically shipped to you. Copy your data to the appliance (it will be encrypted), and then ship it back. The data will be loaded to Amazon S3 on AWS and made accessible from your instances.
For more information about AWS Snowball, see the [AWS website](https://aws.amazon.com/snowball/).

**Oracle RMAN Backup and Restore**

You can use the Oracle Recovery Manager (RMAN) to back up your data, send the backup files to AWS through AWS Snowball, or by using VPN or AWS Direct Connect, and restore your database on AWS. You can also use the RMAN backup file with AWS Database Migration Service to perform the first load.

For more information about Oracle RMAN, see the [Oracle documentation](https://docs.oracle.com/database/19/rman/index.html).

**Oracle Data Pump**

You can use Oracle Data Pump to perform network export/import operations, or send your dump file to the Oracle machines or to Amazon S3 for import operation.

For more information about Oracle Data Pump, see the [Oracle documentation](https://docs.oracle.com/database/19/rman/index.html).

**Backing Up Your Data**

After you deploy Oracle Database on AWS, you’ll want to perform and schedule your database backups. The following sections describe some of the backup options available to you. If you have a production database, we recommend that you set up a backup strategy and implement it as quickly as possible.

**Oracle Secure Backup**

If you chose to install the Oracle Secure Backup Cloud Module, the Quick Start performs an initial, complete backup of your database to the S3 bucket you specified in the Quick Start parameters.

Amazon S3 is the perfect place to save your backups, because it is durable, highly available, and cost-effective, and it has a high throughput to your instances.
After deploying the Quick Start, access your machine and view the Amazon S3 log to see the RMAN backup:

```
You can use the /tmp/rmanbackup.cmd script to perform new backups or to schedule backup tasks and customize the settings for your needs.
```
You can also schedule your backups by using Crontab or another scheduling tool.

### Backups to Disk and Amazon S3

You can perform backups to disk, create and attach a new Amazon EBS volume, create a new file system for local backup, and then copy the backup files by using the AWS CLI `cp` or `sync` command.

To use `cp` for individual files:

```bash
aws s3 cp /localfilesystem/backuppiece.bkp s3://YOURBUCKET/KEY/DB/
```

To use `cp` for a folder:

```bash
aws s3 cp /localfilesystem/backupfolder/ s3://YOURBUCKET/KEY/BACKUP/ --recursive
```

For more information about creating new EBS volumes and file systems, see the AWS documentation.
AMIs and Snapshots

You can use the AWS CLI to generate an Amazon Machine Image (AMI) based on the image of your running instance, and to take snapshots of EBS volumes.

**Important** We recommend that you shut down your database instances before creating an AMI or taking EBS snapshots.

After you create an AMI, you can launch it as a new instance. AWS will assign it a new private IP address, unless you specify your own IP address. If you move your instance to another VPC or another AWS Region, AWS might change its IP address. If so, follow these steps to change the IP address after launching the instance:

1. Edit the following files to reflect the correct IP and host names:
   ```
   /u01/app/oracle/product/12.1.0.2/grid/network/admin/listener.ora
   /u01/app/oracle/product/12.1.0.2/db_1/network/admin/tnsnames.ora
   /etc/hosts
   ```

2. Run the following command to update the installation:
   ```
   sudo su -
   cd /u01/app/oracle/product/12.1.0.2/grid/crs/install
   perl roothas.sh -deconfig -force
   cd /u01/app/oracle/product/12.1.0.2/grid
   ./root.sh
   cd /u01/app/oracle/product/12.1.0.2/grid/bin
   ./srvctl add asm
   ./srvctl add listener
   ./srvctl start asm
   ```

3. As an Oracle user with ASM environment variables loaded, connect to asmcmd and mount the DATA and RECO disk groups.
   ```
   $. oraenv
   +ASM
   asmcmd
   mount data
   ```
Additional Resources

AWS services

- AWS CloudFormation
  - Documentation: [https://aws.amazon.com/documentation/cloudformation/](https://aws.amazon.com/documentation/cloudformation/)

- Amazon EBS


• Amazon S3

• Amazon VPC
  https://aws.amazon.com/documentation/vpc/

**Red Hat Enterprise Linux documentation**

• Swap space

• xorg-x11 drivers packages

**Oracle Database documentation**

• Oracle Database Installation Guide
  https://docs.oracle.com/database/121/LADBI/toc.htm

• Oracle Grid Infrastructure Installation Guide
  https://docs.oracle.com/cd/E11882_01/install.112/e48194/app_nonint.htm#CWWIN417

• Oracle Data Guard Command-Line Interface Reference
  https://docs.oracle.com/database/121/DGBKR/dgmr1.htm#DGBK585

• Oracle Data Guard Concepts and Administration
  https://docs.oracle.com/cd/E11882_01/sbydb/concepts.htm#SBYDB00010

• Oracle ASMLib documentation
  http://www.oracle.com/technetwork/topics/linux/asmlib/index-101839.html

• Oracle ASMLib downloads for RHEL 7
  http://www.oracle.com/technetwork/server-storage/linux/asmlib/rhel7-2773795.html

**Oracle on AWS**

• Advanced Architectures for Oracle Database on Amazon EC2
  https://do.awsstatic.com/enterprise-marketing/Oracle/AWSAdvancedArchitecturesforOracleDBonEC2.pdf
• Best Practices for Running Oracle Database on AWS  
  https://d0.awsstatic.com/whitepapers/best-practices-for-running-oracle-database-on-aws.pdf

• Strategies for Migrating Oracle Database to AWS  
  https://d0.awsstatic.com/whitepapers/strategies-for-migrating-oracle-database-to-aws.pdf

• RDBMS in the Cloud: Oracle Database on AWS  
  https://d0.awsstatic.com/whitepapers/aws-rdbms-oracle.pdf

• Amazon RDS for Oracle Database  
  https://aws.amazon.com/rds/oracle/

**Oracle support notes**

Access to the following documents requires an Oracle account.

• OSB Cloud Module - FAQ (Doc ID 740226.1)  
  https://support.oracle.com/epmos/faces/DocumentDisplay?id=740226.1

• Requirements for Installing Oracle Database 12.1 on RHEL7 or OL7 64-bit (x86-64) (Doc ID 1961997.1)  
  https://support.oracle.com/epmos/faces/DocumentDisplay?id=1961997.1

• Master Note of Linux OS Requirements for Database Server (Doc ID 851598.1)  
  https://support.oracle.com/epmos/faces/DocumentDisplay?id=851598.1

**Quick Start reference deployments**

• AWS Quick Start home page  
  https://aws.amazon.com/quickstart/
Appendix A: Data Replication Between AWS Regions

For the greatest fault tolerance and stability, you can set up a Data Guard copy of your primary database in another AWS Region. This is a common disaster recovery scenario.

This copy must be replicated asynchronously, considering the latency caused by the distance between regions. Asynchronous replication avoids performance impact in your primary database, which are probably connected to each to standby database through a VPN connection.

For this scenario, you should also consider replicating your backup objects in an S3 bucket to make them available in more than one region using cross-region replication.
Figure 13: Oracle disaster recovery scenario with two AWS Regions
Appendix B: Adding Disks to ASM Disk Groups

After deployment, you might need more space for data files or archived log files. To gain more space, you can add EBS volumes to your DATA and RECO disk groups.

1. Create a new EBS volume in the Availability Zones in which you deployed your primary and standby instances.

![Figure 14: Creating a new EBS volume](image)

2. Attach the new volumes to your instances. Take note of the last letter of the device name (for example, for `/dev/sd1`, the last letter is 1, and will appear in the OS as `/dev/xvd1`).

![Figure 15: Attaching the volume](image)
3. Log in to your instances and create a primary partition for the device:

```bash
sudo fdisk /dev/xvdl
n (new partition)
p (primary )
<enter> (default 1 )
<enter> (default xxx)
w (write partition)
```

```
[ec2-user@ip-10-0-0-5 tmp]$ sudo fdisk /dev/xvdl
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel
Building a new DOS disklabel with disk identifier 0x32fd393a.
Changes will remain in memory only, until you decide to write them.
After that, of course, the previous content won't be recoverable.
Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)
WARNING: DOS-compatible mode is deprecated. It's strongly recommended to
switch off the mode (command 'c') and change display units to
sectors (command 'u').
Command (m for help): n
Command action
 e extended
 p primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-655270, default 1):
Using default value 1
Last cylinder, +cylinders or +size(K,M,G) (1-655270, default 655270):
Using default value 655270
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
```

4. Make the partition available to ASM with a name; for example, DATA7. As an Oracle user, add the disk to the corresponding disk group:

```bash
sudo /etc/init.d/oracleasm createdisk DATA7 /dev/xvdl1
```

```
[ec2-user@ip-10-0-0-5 ~]$ sudo /etc/init.d/oracleasm createdisk DATA7 /dev/xvdl1
Marking disk "DATA7" as an ASM disk: [ OK ]
```

```bash
sudo su - oracle
. oraenv
+ASM
sqlplus / as sysasm
alter diskgroup data add disk 'ORCL:DATA7';
```
GitHub Repository

You can visit our GitHub repository to download the templates and scripts for this Quick Start, to post your comments, and to share your customizations with others.

Document Revisions

<table>
<thead>
<tr>
<th>Date</th>
<th>Change</th>
<th>In sections</th>
</tr>
</thead>
<tbody>
<tr>
<td>January 2018</td>
<td>Fixed download links to ASM libraries; removed AMI subscription step</td>
<td>Step 2</td>
</tr>
<tr>
<td>September 2017</td>
<td>Updated for the latest releases of Oracle Linux and Oracle Database 12c</td>
<td>Changes in templates and throughout guide</td>
</tr>
<tr>
<td>April 2017</td>
<td>Added Auto Scaling group, CloudWatch Logs, and additional configuration options to bastion hosts</td>
<td>Changes in templates and throughout guide</td>
</tr>
<tr>
<td>October 2016</td>
<td>Initial publication</td>
<td>—</td>
</tr>
</tbody>
</table>