

Configuring Amazon RDS as an Oracle PeopleSoft Database

July 2019



Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

© 2019 Amazon Web Services, Inc. and DLZP Group. All rights reserved.

Contents

About this Guide	4
Introduction	1
Prerequisites	2
Decide Which AWS Region to Use	2
Identify VPC and Subnets	2
Validate IAM Permissions	2
Determine the Size of the Database	3
Set Up the AWS Command Line Interface (Optional)	3
Certification, Licensing, and Availability	4
PeopleSoft Certification	4
Oracle Licensing	4
Amazon RDS for Oracle Availability	4
Configuring the Database Instance	6
Create Security Groups	6
Create a DB Subnet Group	9
Create an Option Group	11
Create a Parameter Group	13
Modifying Parameters	15
Create the Database Instance	17
Create a DNS Alias for the Database Instance	27
Running the PeopleSoft DB Creation Scripts	30
Editing the Database Scripts	30
Conclusion.....	35
References.....	35
Contributors.....	35
Document Revisions.....	35

About this Guide

Amazon Web Services (AWS) provides a comprehensive set of services and tools for deploying enterprise-grade solutions in a rapid, reliable, and cost-effective manner. Oracle Database is a widely used relational database management system that is deployed and used with many Oracle applications of all sizes to manage various forms of data in many phases of business transactions. In this guide, we describe the preferred method for configuring an Amazon Relational Database Service (Amazon RDS) for Oracle Database as a backend database for Oracle PeopleSoft Enterprise, a widely used enterprise resource planning (ERP) application.

Introduction

An Amazon Relational Database Service (Amazon RDS) for Oracle Database provides scalability, performance, monitoring, and backup and restore support. Deploying an Amazon RDS for Oracle Database in multiple Availability Zones (AZs) simplifies creating a highly available architecture because a multi-AZ deployment contains built-in support for automated failover from your primary database to a synchronously replicated secondary database in an alternative AZ. Amazon RDS for Oracle always provides the latest version of Oracle Database with the latest patch set updates (PSUs) and manages the database upgrade process on your schedule, eliminating manual database upgrade and patching tasks.

You can use Oracle PeopleSoft Enterprise with Amazon RDS and the preferred Oracle Database edition (using your own license, or a license managed by AWS) to create a production Amazon RDS for Oracle Database instance, or the Standard Edition/Standard Edition One/Standard Edition Two to create Amazon RDS for Oracle pre-production environments. Before you can use the PeopleSoft components, you must create and populate schemas for them in your Amazon RDS for Oracle Database. To do so, use the Amazon RDS console or AWS Command Line Interface (AWS CLI) to launch your database (DB) instance. After the instance is created you need to modify the delivered PeopleSoft Database Creation Scripts and run them against the Amazon RDS for Oracle Database instance.

After completing the procedures described in this guide, you can leverage the manageability features of Amazon RDS for Oracle—such as multiple Availability Zones for high availability, hourly pricing of an Oracle Database, and a virtual private cloud (VPC) for network security—while operating the PeopleSoft Enterprise application on AWS.

Prerequisites

Before you create an Amazon RDS for Oracle Database instance, you need to make some decisions about your configuration and complete some basic tasks. You will use the decisions you make in this section later to configure your Amazon RDS for Oracle Database instance.

Decide Which AWS Region to Use

Decide which of the available AWS Regions you want to use for your workload. When choosing a Region, consider the following factors:

- Latency between the end users and the [AWS Region](#).¹
- Latency between your data center and the AWS Region. This is one of the most critical factors when you have PeopleSoft running in the cloud and backends running on premises.
- AWS cost: The AWS service cost varies depending on the Region.
- Legislation and compliance: There might be restrictions on which country your customers' data can be stored in.

Identify VPC and Subnets

Determine which VPC and subnets you will be using to deploy your resources. If you don't have a VPC, you can create an Amazon Virtual Private Cloud (Amazon VPC) by referring to the [Amazon Virtual Private Cloud User Guide](#).²

NOTE: If creating an Amazon VPC, follow Step 1: Create the VPC from the [Amazon Virtual Private Cloud User Guide](#). You will be creating a security group using this guide.

Validate IAM Permissions

You must have AWS Identity and Access Management (IAM) permissions to perform the actions described in this guide. You will need permissions to configure the following AWS services:

- [Amazon Virtual Private Cloud](#)³
- [Amazon Elastic Compute Cloud \(Amazon EC2\)](#)⁴

- [Amazon Relational Database Service](#)⁵
- [Amazon Route 53](#)⁶

Determine the Size of the Database

Determine the database (DB) size you will require for the installation.

Table 1 lists various DB instance classes by size of the PeopleSoft Environment. Note that this table is provided only as a guideline. You should validate your individual class size requirements against your actual usage.

For a current listing of available instance classes, refer to [Amazon RDS for Oracle Pricing](#).

Table 1: DB instance classes by size of the PeopleSoft environment

DB Instance Class	Notes
medium	Ideal for a small PeopleSoft demo/dev environment
large	Ideal for a medium PeopleSoft environment: <100 users
xlarge	Ideal for a medium PeopleSoft environment: <1,000 users
2xlarge	Ideal for a medium PeopleSoft environment: <10,000 users
4xlarge	Ideal for a large PeopleSoft environment: <50,000 users
8xlarge	Ideal for a very large PeopleSoft environment: <250,000 users

Set Up the AWS Command Line Interface (Optional)

You can use either the AWS Management Console or the AWS CLI to perform the tasks described in this guide. To use the AWS CLI, ensure that you have installed AWS CLI and that you have either an Amazon EC2 instance that has an AWS IAM role associated with it (recommended) or an access key ID and secret key.

Certification, Licensing, and Availability

Before getting started with installing a PeopleSoft application on Amazon RDS for Oracle, check for certification, make licensing considerations, and verify general availability.

PeopleSoft Certification

Oracle certification for PeopleSoft software is controlled by the PeopleTools version that is being used. Use your My Oracle Support account to check that your PeopleTools version is currently certified to run on the Oracle Database Release you plan to use with Amazon RDS, as well as review any PeopleSoft application certification notes that may apply.

NOTE: Oracle has numerous documents on **My Oracle Support** regarding support for Oracle Applications in the Cloud. Documents regarding issues with deploying PeopleSoft on Amazon RDS for Oracle are resolved by the steps in this guide. In addition, there are features that are specific to a database release that may or may not be available based on the database edition that you own.

Oracle Licensing

When creating an Amazon RDS for Oracle database, you can select either **Bring Your Own License (BYOL)** or **License Included (LI)**. Not all editions are available for License Included. Before creating the database instance, verify which license(s) your organization holds, if any. For more details, refer to the [Amazon RDS for Oracle FAQs](#).

Amazon RDS for Oracle Availability

After reviewing certification and licensing, refer to Table 2 to identify the Oracle Database Release and corresponding PeopleTools Release along with other details that are available on Amazon RDS for Oracle.

Refer to the Amazon RDS User Guide section on [Oracle on Amazon RDS](#) for an up to date list of available RDS Oracle Releases.

Table 2: Certified Oracle releases for PeopleTools available on Amazon RDS

PeopleTools Release	Amazon RDS Oracle DB Release	Amazon RDS Oracle DB Edition	Amazon RDS Oracle DB License Model
8.57, 8.56	12.2.0.1	Enterprise Edition	LI
		Standard Edition Two	LI, BYOL
	12.1.0.2	Enterprise Edition	LI
		Standard Edition Two	LI, BYOL
8.55	12.2.0.1	Enterprise Edition	LI
		Standard Edition Two	LI, BYOL
	12.1.0.2	Enterprise Edition	LI
		Standard Edition Two	LI, BYOL
	11.2.0.4	Enterprise Edition	LI
		Standard Edition	LI, BYOL
Standard Edition One		LI, BYOL	
8.54, 8.53	12.1.0.2	Enterprise Edition	LI
		Standard Edition Two	LI, BYOL
	11.2.0.4	Enterprise Edition	LI
		Standard Edition	LI, BYOL

PeopleTools Release	Amazon RDS Oracle DB Release	Amazon RDS Oracle DB Edition	Amazon RDS Oracle DB License Model
		Standard Edition One	LI, BYOL

Configuring the Database Instance

You can configure AWS resources by using either the AWS Management Console or AWS CLI. Steps for both options are provided in this guide. If you plan to use AWS CLI, follow the console procedure first because it provides context for the step. The AWS CLI commands provided in this guide map directly to the tasks executed using the console.

NOTE: This guide provides the steps for creating a Demo PeopleSoft environment. As such, the settings and configurations provided apply towards this smaller environment where performance is not a requirement.

Create Security Groups

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. For more information on Security Groups, reference [Security Groups for Your VPC](#).

You will create two security groups to define network level traffic to the Amazon RDS database; one for the Amazon RDS database and one for the Amazon EC2 application servers that will access the database. Using security groups to define database network access allows you to be more restrictive and intentional about security. For example, by having separate security groups for Production and Development environments, you can prevent Development servers from communicating with the Production database.

Table 3: Security groups inbound rules

Security Group Name	Protocol	Port Range	Source	Notes
peoplesoft-demo-app	None	None	None	Attach this security group to your PeopleSoft Application server EC2 instances
peoplesoft-demo-db	TCP	1521	peoplesoft-demo-app	Allow 1521 traffic from peoplesoft-demo-app

Using the AWS Management Console

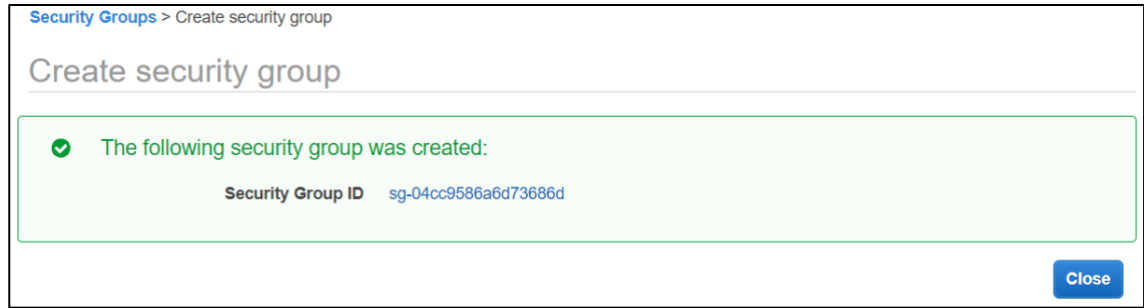
Create peoplesoft-demo-app

1. In the console, choose **Services, VPC, Security Groups, Create Security Group**.
2. Enter **peoplesoft-demo-app** for the **Security group name** and **PeopleSoft Demo Application Server** for the **Description**. Select the appropriate **VPC** for your account.

The screenshot shows the 'Create security group' page in the AWS Management Console. At the top, it says 'Security Groups > Create security group'. Below that is the title 'Create security group'. A note states: 'A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group fill in the fields below.' There are three input fields: 'Security group name*' with the value 'peoplesoft-demo-app', 'Description*' with the value 'PeopleSoft Demo Application Server', and 'VPC' with a dropdown menu showing 'No VPC'. Each field has an information icon (i) to its right. At the bottom left, there is a note '* Required'. At the bottom right, there are 'Cancel' and 'Create' buttons.

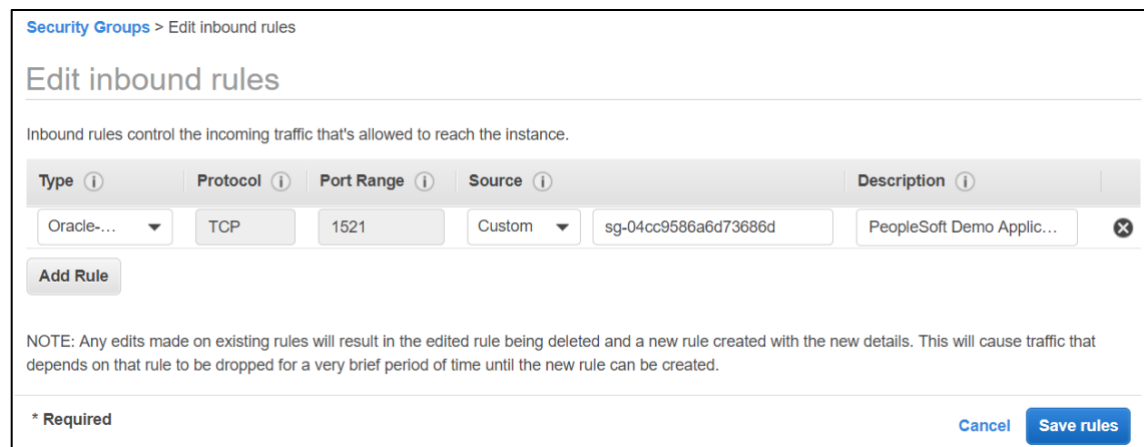
3. Choose **Create**.

- Note the Security Group ID for later use.



Create peoplesoft-demo-db

- In the console, choose **Services, VPC, Security Groups, Create Security Group**.
- For the **Security group name**, enter *peoplesoft-demo-db* and for the **Description**, enter *PeopleSoft Demo RDS Database*.
- Select the appropriate **VPC** for your account and choose **Create**.
- To update the Inbound Rules:
 - Select **peoplesoft-demo-db** from the security group list
 - Choose **Actions, Edit inbound rules**
 - For **Type** select **Oracle-RDS**
 - For **Source** select **Custom**
 - Enter the **Security Group ID** from the previous step
 - For **Description**, enter *PeopleSoft Demo Application Server*



- Choose **Save rules**.

Using the AWS CLI

Update the VPC_ID variable below with your VPC and execute in your CLI environment.

```
VPC_ID=<Replace with your VPC ID>

PS_APP_SG=$(aws ec2 create-security-group --group-name
peoplesoft-demo-app --description "PeopleSoft Demo
Application Server" --vpc-id $VPC_ID --output text)

aws ec2 create-tags --resources $PS_APP_SG --tags
"Key=Name,Value= PeopleSoft Demo Application Server"

PS_RDS_SG=$(aws ec2 create-security-group --group-name
peoplesoft-demo-db --description "PeopleSoft Demo RDS
Database" --vpc-id $VPC_ID --output text)

aws ec2 create-tags --resources $PS_RDS_SG --tags
"Key=Name,Value= PeopleSoft Demo RDS Database"

aws ec2 authorize-security-group-ingress --group-id
$PS_RDS_SG --protocol tcp --port 1521 --source-group
$PS_APP_SG
```

Create a DB Subnet Group

Before you can create an Amazon RDS for Oracle Database instance, you must define a subnet group. A subnet group is a collection of subnets (typically private) that you create in a VPC and designate for your DB instances. For an Amazon RDS for Oracle Database, you must select two subnets, each in a different Availability Zone.

Using the AWS Management Console

1. In the Console, choose **Services, RDS, Subnet Groups, Create DB Subnet Group**.
2. For the **Subnet Group details** section, specify the following:
 - For **Name**, enter *peoplesoft-demo-subnetgroup*
 - For **Description**, enter *PeopleSoft Demo Subnet Group*

- For **VPC**, choose your VPC.

Create DB subnet group

To create a new subnet group, give it a name and a description, and choose an existing VPC. You will then be able to add subnets related to that VPC.

Subnet group details

Name
You won't be able to modify the name after your subnet group has been created.

Must contain from 1 to 255 characters. Alphanumeric characters, spaces, hyphens, underscores, and periods are allowed.

Description

VPC
Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created.

3. For the **Add Subnets** section, use the **Availability zone** drop down to choose an AZ, select a Subnet designated for databases, and Choose **Add subnet**. Choose a minimum of 2 Subnet's each from a different Availability Zone.

Add subnets

Add subnet(s) to this subnet group. You may add subnets one at a time below or add all the subnets related to this VPC. You may make additions/edits after this group is created. A minimum of 2 subnets is required.

Availability zone

Subnet

Subnets in this subnet group (2)

Availability zone	Subnet ID	CIDR block	Action
us-west-2b	subnet-	10.21.12.0/23	<input type="button" value="Remove"/>
us-west-2a	subnet-	10.21.2.0/23	<input type="button" value="Remove"/>

4. Choose **Create**.

Using the AWS CLI

Update the `PS_RDS_SN_1` and `PS_RDS_SN_2` variables below with the subnets you created in the previous step and execute in your CLI environment.

```
PS_RDS_SN_1=<Replace with Subnet ID 1>
PS_RDS_SN_2=<Replace with Subnet ID 2>

aws rds create-db-subnet-group --db-subnet-group-name
peoplesoft-demo-subnetgroup --db-subnet-group-description
"PeopleSoft Demo Subnet Group" --subnet-ids $PS_RDS_SN_1
$PS_RDS_SN_2
```

Create an Option Group

An Option Group provides additional feature options that you might want to add to your Amazon RDS for Oracle DB instance. Amazon RDS provides default Option Groups, but they cannot be modified. For this reason, create a new option group so feature options can be added or modified later. You can assign an option group to multiple Amazon RDS for Oracle DB instances. For a production database, always review your current Oracle licensing agreement. For more information reference [Options for Oracle DB Instances](#). Licensing may be required.

IMPORTANT: The only option *required* for PeopleSoft to run correctly is *Timezone*. This must be set to have the desired timestamps inside PeopleSoft.

Using the AWS Management Console

1. In the AWS Management Console, choose **Services, RDS, Option Groups, Create Group**.
2. Specify values for the following fields:
 - For **Name**, enter `peoplesoft-demo-oracle-ee-122`
 - For **Description**, enter `PeopleSoft Demo Option Group`
 - For **Engine**, choose the Engine that correlates with the Oracle Database Edition chosen in the [Certification, Licensing, and Availability](#) section of this paper.

- For **Major Engine version**, choose the Major engine version that correlates with the Oracle Database Release chosen in the [Certification, Licensing, and Availability](#) section of this paper.

The screenshot shows the 'Create option group' dialog box. It has a title bar 'Create option group' and a section 'Option group details'. The fields are: Name (peoplesoft-demo-oracle-ee-122), Description (PeopleSoft Demo Option Group), Engine (oracle-ee), and Major engine version (12.2). At the bottom right, there are 'Cancel' and 'Create' buttons.

3. Choose **Create**.
4. To update the Option Group, select **peoplesoft-demo-og** from the list of option groups and choose **Add option**.
5. Select **Timezone** from the Option list and choose your local Time Zone you want to be reflected in PeopleSoft.

The screenshot shows the 'Add Option' dialog box. It has a title bar 'Add Option' and a section 'Option details'. The fields are: Option group name (peoplesoft-demo-og), Option (Timezone), and Time Zone (US/Pacific). There is also an 'Apply Immediately' section with radio buttons for 'Yes' (selected) and 'No', and an 'Info' link. At the bottom right, there are 'Cancel' and 'Add Option' buttons.

6. Select whether or not you want the DB instance option to be applied immediately and choose **Add Option**.

Using the AWS CLI

Update `PS_TZ` with a valid Time Zone, such as US/Pacific, and execute in your CLI environment.

```
PS_TZ=<Replace with Time Zone>

aws rds create-option-group --option-group-name peoplesoft-
demo-oracle-ee-122 --engine-name oracle-ee --major-engine-
version 12.2 --option-group-description "PeopleSoft Demo
Option Group"

aws rds add-option-to-option-group --option-group-name
peoplesoft-demo-oracle-ee-122 --apply-immediately --options
"OptionName=Timezone,OptionSettings=[{Name=TIME_ZONE,Value=$
PS_TZ}]"
```

Create a Parameter Group

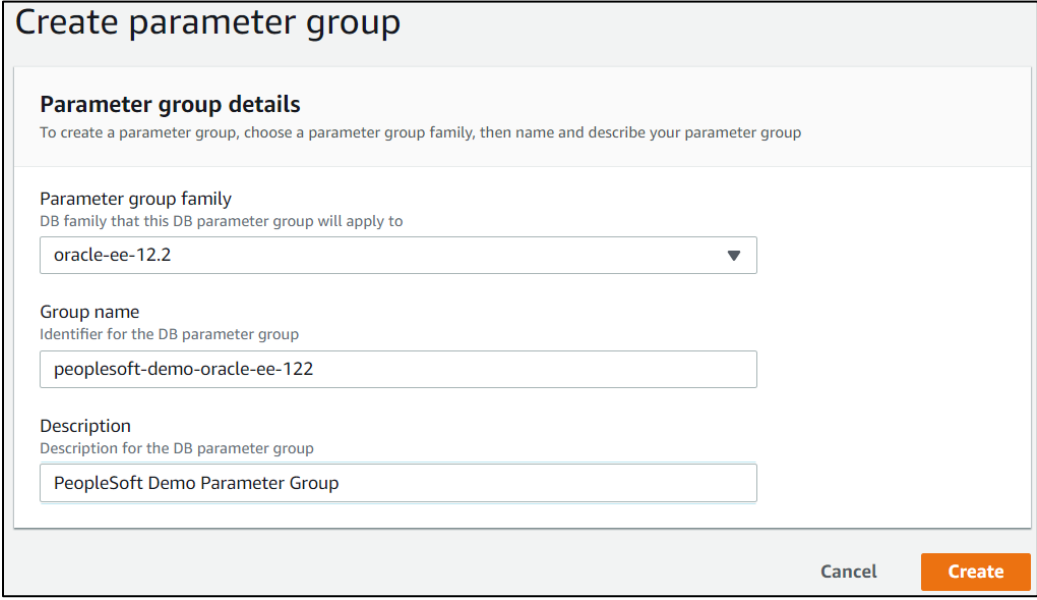
A DB parameter group acts as a container for engine configuration values that are applied to one or more DB instances. If you create a DB instance without specifying a DB parameter group, the DB instance uses a default DB parameter group. It is not possible to update the default parameter group, therefore it is recommended that you create a new parameter group, even if you don't need to customize any parameters at this point.

It is also recommended to consider how you will reuse the parameter groups among multiple Amazon RDS for Oracle DB instances. For a PeopleSoft deployment, it is recommended that you use a unique parameter group for each environment (DEV, TEST, PROD), since parameters may need to be modified to suit a particular use case and/or provide you the ability to test a change in the configuration prior to applying it to a new environment.

For more information, refer to the Amazon RDS User Guide [Working with DB Parameter Groups](#).

Using the AWS Management Console

1. In the AWS Management Console, choose **Services, RDS, Parameter Groups, Create Parameter Group**.
2. Specify values for the following fields:
 - For **Parameter group family**, choose the database edition you want to use in your RDS for Oracle DB instances. In this example, *oracle-ee-12.2* is used.
 - For **Group name**, enters *peoplesoft-demo-oracle-ee-122*
 - For **Description**, enter *PeopleSoft Demo Parameter Group*



The screenshot shows the 'Create parameter group' form in the AWS Management Console. The form is titled 'Create parameter group' and has a subtitle 'Parameter group details'. Below the subtitle is a note: 'To create a parameter group, choose a parameter group family, then name and describe your parameter group'. The form contains three input fields: 'Parameter group family' (a dropdown menu with 'oracle-ee-12.2' selected), 'Group name' (a text input field with 'peoplesoft-demo-oracle-ee-122' entered), and 'Description' (a text input field with 'PeopleSoft Demo Parameter Group' entered). At the bottom right of the form are two buttons: 'Cancel' and 'Create'.

3. Choose **Create**.

Using the AWS CLI

Execute the following command in your CLI environment.

```
aws rds create-db-parameter-group --db-parameter-group-name
peoplesoft-demo-oracle-ee-122 --db-parameter-group-family
oracle-ee-12.2 --description "PeopleSoft Demo Parameter
Group"
```

Modifying Parameters

For PeopleSoft, there are recommended parameters for Oracle databases as shown in Table 4.

Table 4: List of parameters to customize

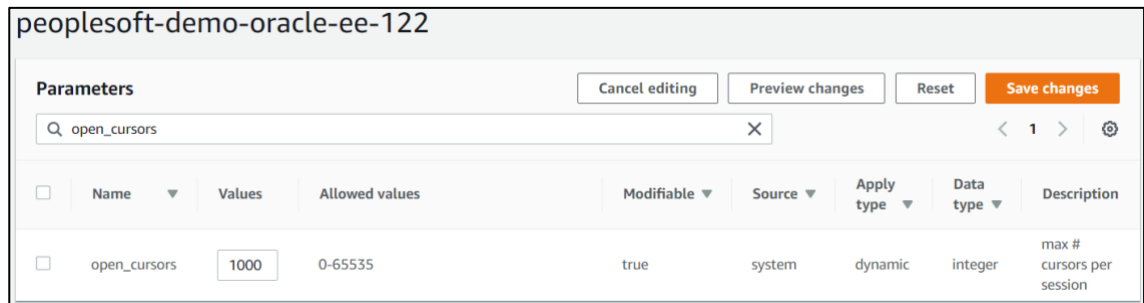
Parameter	Value	Notes
open_cursors	1000	
db_block_size		This parameter is automatically set by Amazon RDS (although the creation of nonstandard block size tablespaces and setting DB_nK_CACHE_SIZE parameters is supported)
db_files	1021	Optionally, you can leave the default setting provided by Amazon RDS.
nls_length_semantic		CHAR for Unicode, BYTE for non-Unicode
memory_target	{DBInstanceClassMemory*3/4};	The default may be used; change if you have a specific requirement.
_gby_hash_aggregation_enabled	false	This hash scheme enables group-by and aggregation
_unnest_subquery	false	Enable un-nesting of complex subqueries
optimizer_adaptive_features	false	This parameter is for version 12.1.x. You can either enable or disable the adaptive optimize features.
optimizer_adaptive_plans	true (default)	This parameter is for version 12.2.x. You can either enable or disable the adaptive optimize features.
optimizer_adaptive_statistics	false (default)	This parameter is for version 12.2.x. You can either enable or disable the adaptive optimize features.

Parameter	Value	Notes
<code>_fix_control</code>	14033181:0	This parameter is for Oracle 12.1.x Databases ONLY and is an interim patch resolution that you can enable. This patch is not required for Version 12.2.x release.

NOTE: This is an example of known settings as of the publication date. For more details about specific PeopleSoft–Oracle parameter settings, refer to the following Oracle Support Document: **E-ORA Advice for the PeopleSoft Oracle DBA** (Doc ID 1445965.1). For instance classes with at least 100 GiB of memory, use `sga_target` and enable [Huge Pages](#).

Using the AWS Management Console

1. Select the parameter group you created (example shown below).
2. Choose **Parameter group actions, Edit**.
3. Type the name of the **Parameter** you want to edit (as listed in Table 4).
4. Optional: for example, enter `open_cursors` into the filter, change the **Values** field to **1000**, and then choose **Save Changes**.



5. Repeat step 3 and 4 for each parameter you want to edit.

Using the AWS CLI

Update the commands if you need to customize other parameters. Execute the following command in your CLI environment.

```
aws rds modify-db-parameter-group --db-parameter-group-name
peoplesoft-demo-oracle-ee-122 --parameters
"ParameterName=open_cursors,ParameterValue=1000,ApplyMethod=
pending-reboot"
```

Create the Database Instance







Next, you are ready to create a highly available Oracle Database across two Availability Zones. Keep in mind that running a database in multiple Availability Zones increases cost. Depending on your SLA requirements, you can consider running the database in a single Availability Zone instead.

Using the AWS Management Console

1. In the Amazon RDS console, select **Create database**, choose **Oracle** and your **Edition** and choose **Next**.


Select engine

Engine options

<input type="radio"/> Amazon Aurora 	<input type="radio"/> MySQL 	<input type="radio"/> MariaDB 
<input type="radio"/> PostgreSQL 	<input checked="" type="radio"/> Oracle 	<input type="radio"/> Microsoft SQL Server 

Oracle Edition

- Oracle Enterprise Edition**
Efficient, reliable, and secure database management system that delivers comprehensive high-end capabilities for mission-critical applications and demanding database workloads.
- Oracle Standard Edition
Affordable and full-featured database management system supporting up to 32 vCPUs.
- Oracle Standard Edition One
Affordable and full-featured database management system supporting up to 16 vCPUs.
- Oracle Standard Edition Two
Affordable and full-featured database management system supporting up to 16 vCPUs. Oracle Database Standard Edition Two is a replacement for Standard Edition and Standard Edition One.

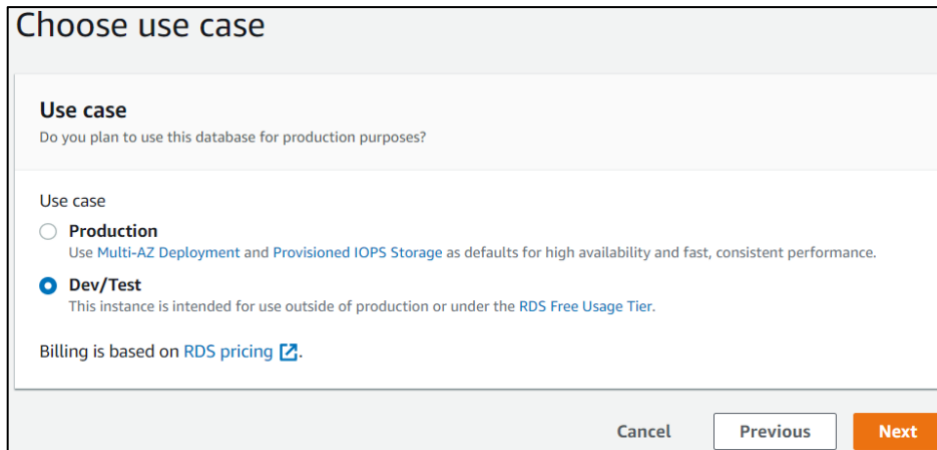
 **Aurora global database feature is now available.**
This feature is now available in our new database creation flow.

[Try it now](#)

Only enable options eligible for RDS Free Usage Tier [Info](#)

[Cancel](#) [Next](#)

- For our example, choose the **Dev/Test** template, then choose **Next**.



Choose use case

Use case
Do you plan to use this database for production purposes?

Use case

Production
Use [Multi-AZ Deployment](#) and [Provisioned IOPS Storage](#) as defaults for high availability and fast, consistent performance.

Dev/Test
This instance is intended for use outside of production or under the RDS Free Usage Tier.

Billing is based on [RDS pricing](#).

Cancel Previous **Next**

- Specify the DB details:

- License model:** Choose the License model, which depends on your Oracle Edition. Reference the [Certification, Licensing, and Availability](#) section.
- DB engine version:** Choose the most recent engine version. The most recent version will have all Oracle patches available to Amazon RDS. Reference the [Certification, Licensing, and Availability](#) section.
- DB instance class:** Because this is for Demo Purposes, choose a relatively small DB instance class, such as **db.t3.medium**. You can change the DB instance class at any point, which requires restarting the DB instance.
- Multi-AZ deployment:** Choose **Yes** so that you can have a second standby instance running in a second Availability Zone.
- Storage type:** For a Dev/Test environment, choose **General Purpose (SSD)**. For a high-performance environment in production, **Provisioned IOPS (SSD)** should be used.
- Allocated storage:** Allocate **200 GB**. Note that baseline I/O performance for General Purpose SSD storage is 3 IOPS for each GiB. This will give you a 600 IOPS baseline and can burst to 3000 IOPS using credits. For more information on credits, refer to the Amazon RDS User Guide [I/O Credits and Burst Performance](#).

Specify DB details

Instance specifications
 Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#)

DB engine
 Oracle Database Enterprise Edition

License model [Info](#)
 bring-your-own-license

DB engine version [Info](#)
 Oracle 12.2.0.1.ru-2019-04.rur-2019-04.r1

Free tier
 The Amazon RDS Free Tier provides a single db.t2.micro instance as well as up to 20 GiB of storage, allowing new AWS customers to gain hands-on experience with Amazon RDS. Learn more about the RDS Free Tier and the instance restrictions [here](#).

Only enable options eligible for RDS Free Usage Tier [Info](#)

DB instance class [Info](#)
 db.t3.medium — 2 vCPU, 4 GiB RAM

Multi-AZ deployment [Info](#)
 Create replica in different zone
Creates a replica in a different Availability Zone (AZ) to provide data redundancy, eliminate I/O freezes, and minimize latency spikes during system backups.

No

Storage type [Info](#)
 General Purpose (SSD)

Allocated storage
 GiB
(Minimum: 20 GiB, Maximum: 32768 GiB) Higher allocated storage [may improve](#) IOPS performance.

4. Review the monthly costs.

Estimated monthly costs

DB Instance	99.28 USD
Storage	46.00 USD
Total	145.28 USD

Billing estimate is based on on-demand usage as described in [Amazon RDS Pricing](#). Estimate does not include costs for backup storage, IOs (if applicable), or data transfer.

Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#)

5. Specify the **Settings**:

- **DB instance identifier:** Create a unique name for the DB instance identifier. Amazon RDS uses this identifier to define the database hostname. Our example uses **psfdmo**.
- **Master username:** This is similar to the SYS user, but with fewer privileges because Amazon RDS does not allow you to use either a SYS user or the SYSDBA role. Our example uses **psftadmin**.
- **Master password:** Create a password for the master user. The master password must be at least eight characters long and can include any printable ASCII character except for the following: /, ", or @.

Settings

DB instance identifier [Info](#)
Specify a name that is unique for all DB instances owned by your AWS account in the current region.

DB instance identifier is case insensitive, but stored as all lower-case, as in "mydbinstance". Must contain from 1 to 63 alphanumeric characters or hyphens (1 to 15 for SQL Server). First character must be a letter. Cannot end with a hyphen or contain two consecutive hyphens.

Master username [Info](#)
Specify an alphanumeric string that defines the login ID for the master user.

Master Username must start with a letter. Must contain 1 to 30 alphanumeric characters.

Master password [Info](#)

Master Password must be at least eight characters long, as in "mypassword". Can be any printable ASCII character except "/", "", or "@".

Confirm password [Info](#)

Cancel Previous Next

6. Choose **Next**.

7. Specify the DB instance advanced settings:

- **VPC:** Choose the VPC where the database will be deployed.
- **Subnet group:** Choose the subnet group you created in [Create a DB Subnet Group](#). The DB instance is deployed against the subnets associated with the subnet group.
- **Public Accessibility:** Allows external access to the database, provided it's deployed in a public subnet. In most cases you would choose **No** to restrict access to (1) within our VPC and (2) use security groups.

- **Availability zone:** Use the default **No preference**.
- **VPC security groups:** Choose the security group that will be associated with your DB instance. This security group provides access to the database listener. You created this security group **peoplesoft-demo-db** in [Create Security Groups](#). Remove the default security group.

Configure advanced settings

Network & Security

Virtual Private Cloud (VPC) [Info](#)
VPC defines the virtual networking environment for this DB instance.

▼ ↻

Only VPCs with a corresponding DB subnet group are listed.

Subnet group [Info](#)
DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

▼

Public accessibility [Info](#)

Yes
EC2 instances and devices outside of the VPC hosting the DB instance will connect to the DB instances. You must also select one or more VPC security groups that specify which EC2 instances and devices can connect to the DB instance.

No
DB instance will not have a public IP address assigned. No EC2 instance or devices outside of the VPC will be able to connect.

Availability zone [Info](#)

▼

VPC security groups
Security groups have rules authorizing connections from all the EC2 instances and devices that need to access the DB instance.

Create new VPC security group

Choose existing VPC security groups

▼

✕

8. Specify your database options:

- **Database name:** Choose the database service name, which your database clients will use to connect. Our example uses **PSFDMO**. This follows Oracle Database DB_NAME naming conventions, reference Oracle's documentation [Selecting a Database Name](#) for more info.
- **Port:** Choose the TCP port that the database listener listens on. In our example, we chose **1521**, which is a default port for Oracle.
- **DB parameter group:** The database engine parameters. Choose the **peoplesoft-demo-oracle-ee-122** DB parameter group that you created in [Create a Parameter Group](#).
- **Option group:** The features that will be enabled in the database. Choose the **peoplesoft-demo-oracle-ee-122** option group, which you created in [Create an Option Group](#).
- **Character set name:** Choose the character set for your database. In our example, we use **WE8ISO8859P15**, a non-Unicode database. Use the character set that is required for your PeopleSoft installation.

Database options

Database name [Info](#)

If you do not specify a database name, Amazon RDS does not create a database.

Port [Info](#)
TCP/IP port the DB instance will use for application connections.

DB parameter group [Info](#)

Option group [Info](#)


Character set name [Info](#)

9. Choose whether to enable or disable the encryption.

If you require your database data to be encrypted, choose **Enable encryption**. For more information, see the Amazon RDS User Guide [Encrypting Amazon RDS Resources](#). For our example, we chose **No**.

Encryption

Encryption

Enable encryption [Learn more](#) 

Select to encrypt the given instance. Master key ids and aliases appear in the list after they have been created using the Key Management Service(KMS) console.

Disable encryption

10. Specify the backup options:

- **Backup retention period:** Set the backup retention period in days (maximum of 35) for your database. Set to 0 to disable automatic backups. Our example uses the default of **7 days**.
- **Backup window:** Set the backup window for your daily backup. Choose the default **No Preference**.
- **Copy tags to snapshots:** When this option is enabled, Amazon RDS copies any tag associated with your DB instance to the database snapshots; useful for tracking usage and cost. Select the checkbox to enable this option.

Backup

Backup retention period [Info](#)
Select the number of days that Amazon RDS should retain automatic backups of this DB instance.

7 days ▼

Backup window [Info](#)

Select window

No preference

Copy tags to snapshots

11. Choose whether to enable or disable the monitoring option.

Enhanced Monitoring provides Amazon RDS metrics in real time for the operating system (OS) that your DB instance runs on. For more information, see the Amazon RDS User Guide [Enhanced Monitoring](#). Our example disables this option.

Monitoring

Enhanced monitoring

Enable enhanced monitoring
Enhanced monitoring metrics are useful when you want to see how different processes or threads use the CPU.

Disable enhanced monitoring

12. Choose whether to enable or disable the performance insights option.

Amazon RDS **Performance Insights** monitors your Amazon RDS DB instance load so that you can analyze and troubleshoot your database performance. For more information, see the Amazon RDS User Guide [Using Amazon RDS Performance Insights](#). In our example, we chose **Disable Performance Insights**.

Performance Insights

Enable Performance Insights

Disable Performance Insights

13. Specify the log(s) to export, if any.

Database logs can be exported to Amazon CloudWatch, this can be useful in a Production environment. In our example, we chose not to export any logs.

Log exports

Select the log types to publish to Amazon CloudWatch Logs

- Alert log
- Audit log
- Listener log
- Trace log

IAM role
The following service-linked role is used for publishing logs to CloudWatch Logs.

RDS Service Linked Role

14. Specify the maintenance settings.

- **Auto Minor Version Upgrade:** If you want to manage when database maintenance runs, then select to **Disable auto minor version upgrade**.
- **Maintenance Window:** Set the timing for the minor maintenance window. For our example, we chose **No Preference**.

Maintenance

Auto minor version upgrade [Info](#)

- Enable auto minor version upgrade
Enables automatic upgrades to new minor versions as they are released. The automatic upgrades occur during the maintenance window for the DB instance.
- Disable auto minor version upgrade

Maintenance window [Info](#)
Select the period in which you want pending modifications or patches applied to the DB instance by Amazon RDS.

- Select window
- No preference

15. Determine whether to enable the delete protection option (for most use cases, this should be checked to prevent accidental deletion of the instance). For our example, the option is unchecked.

Deletion protection

Enable deletion protection
Protects the database from being deleted accidentally. While this option is enabled, you can't delete the database.

Cancel
Previous
Create database

16. Choose **Create database** to launch the DB instance.

Using the AWS CLI

Update the `RDS_MASTER_PWD` and `PS_RDS_SG` variables and modify as necessary. Execute the in your CLI environment.

```

RDS_MASTER_PWD=<Replace with password>
PS_RDS_SG=<Replace with DB Security Group ID>

aws rds create-db-instance \
  --db-name PSFDMO \
  --db-instance-identifier psfdmo \
  --allocated-storage 200 \
  --db-instance-class db.t3.medium \
  --engine oracle-ee \
  --master-username psftadmin \
  --master-user-password $RDS_MASTER_PWD \
  --vpc-security-group-ids $PS_RDS_SG \
  --db-subnet-group-name peoplesoft-demo-subnetgroup \
  --db-parameter-group-name peoplesoft-demo-oracle-ee-122 \
  --port 1521 \
  --multi-az \
  --engine-version 12.2.0.1.ru-2019-04.rur-2019-04.r1 \
  --no-auto-minor-version-upgrade \
  --license-model bring-your-own-license \
  --option-group-name peoplesoft-demo-oracle-ee-122 \
  --character-set-name WE8ISO8859P15 \
  --no-storage-encrypted \
  --no-publicly-accessible \
  --no-enable-performance-insights \
  --no-deletion-protection \
  --storage-type gp2 \
  --copy-tags-to-snapshot

```

Create a DNS Alias for the Database Instance

When you create an RDS for Oracle DB instance, Amazon RDS creates a unique DNS hostname for your instance (for example, `psdmo.c6jc3rya3ntd.us-east-1.rds.amazonaws.com:1521`). You can use that hostname to connect to the database. However, you have no control over the hostname, so you end up with a database URL that is not easy to remember.

In addition, you might at some point need to restore the database from a snapshot. For example, you would need to do so if an operator makes a mistake in manipulating the data, or if a bug in your application corrupts the data. But you can't restore a snapshot to an existing RDS DB instance. When you restore a database from a snapshot, Amazon RDS creates a new DB instance and generates a new hostname.

To avoid affecting existing applications and having to update their database endpoints, create a DNS alias for your RDS DB instance. Depending on your architecture, you may register the DNS alias either in your corporate DNS server running on premises, or in a DNS server running in AWS.

We will show you how to register a DNS alias in AWS as a private hosted zone using Amazon Route 53. When you use a private hosted zone, only hosts in your VPC can resolve the DNS names for your database. (There is a way to extend the name resolution outside of the VPC, but that's beyond the scope of this guide.)

Create an Amazon Route 53 Private Hosted Zone

Create a private hosted name using either the AWS Management Console or using the AWS CLI.

Using the AWS Management Console

1. In the AWS Management Console, choose **Services, Route 53, Hosted zones** and then choose **Create Hosted Zone**.
2. Provide the details of your hosted zone:
 - **Domain Name:** Type the private domain name. Our example uses **peoplesoft.local**.
 - **Type:** Choose **Private Hosted Zone for Amazon VPC**.

- **VPC ID:** Choose the ID of the VPC used by your PeopleSoft infrastructure in AWS.

The screenshot shows the 'Create Hosted Zone' page in the AWS Management Console. The page includes a sidebar with navigation options like Dashboard, Hosted zones, Health checks, Traffic flow, Traffic policies, Policy records, Domains, Registered domains, and Pending requests. The main content area has a search bar and a description of a hosted zone. The form fields are filled with: Domain Name: peoplesoft.local, Comment: PeopleSoft Local DNS, Type: Private Hosted Zone for Amazon VPC, and VPC ID: vpc-5524ff30 | us-east-1. An 'Important' note is displayed, stating that to use private hosted zones, certain VPC settings must be enabled: enableDnsHostnames and enableDnsSupport. A 'Learn more' link is provided for further details.

Using the AWS CLI

Execute the following command:

```
aws route53 create-hosted-zone --name peoplesoft.local --
vpc '{"VPCRegion":"us-east-1", "VPCId": "vpc-*****21"}' --
hosted-zone-config '{"PrivateZone": true}' --caller-
reference 112017
```

Note: Retain the **HostedZone ID** because you will need it to create the record sets in the next section. For the command provided above, the **HostedZone ID** is: Z234334ABCDEF.

Create a DNS Alias

By creating a DNS alias, you can manage your database's endpoint and avoid the need to change the code in your existing application.

Using the AWS Management Console

1. Choose **Create Record Set** and then provide the following details:
 - **Name:** The fully qualified domain name, which in our example is **psfdmo**. The value you type will be prepended to the domain name. In our example, it is **psfdmo.peoplesoft.local**.

- **Type:** Choose **CNAME**.
- **Alias:** Choose **No**.
- **TTL (Seconds):** Enter **300**.
- **Value:** The RDS instance hostname (do not add the port information 1521).

The screenshot shows the AWS Route 53 console interface for creating a record set. The 'Create Record Set' form is active, with the following fields filled out:

- Name:** psdmo.peoplesoft.local
- Type:** CNAME - Canonical name
- Alias:** No
- TTL (Seconds):** 300
- Value:** psdmo.ak34e3k.rds.amazonaws.com
- Routing Policy:** Simple

The left sidebar shows a list of record sets for the hosted zone 'peoplesoft.local', including NS and SOA records.

Using the AWS CLI

Execute the following command:

```
aws route53 change-resource-record-sets --hosted-zone-id
Z234334ABCDEF --change-batch '{"Changes": [{"Action":
"CREATE", "ResourceRecordSet": {"Name":
"psdmo.peoplesoft.local", "Type": "CNAME", "TTL": 300,
"ResourceRecords": [{"Value":
"psdmo.ak34e3k.rds.amazonaws.com"}]}}]}'
```

After creating a DNS alias, connect to our demo database using the following URL:

```
psfdmo.peoplesoft.local:1521/psfdmo
```

Running the PeopleSoft DB Creation Scripts

With the database created, you are ready to create the PeopleSoft DB. The following steps illustrate the procedure detailed in the **PeopleTools 8.5x Installation for Oracle** guide. Ensure that you use the appropriate PeopleTools install guide for your installation and note the following changes in the manual installation steps.

Editing the Database Scripts

To start, modify the delivered database creation scripts. There are 2 types of changes that we need to make to these scripts: (1) Tablespace creation and (2) SYSDBA SQL commands.

Tablespace Creation

For creating tablespace, Amazon RDS supports only the Oracle Managed Files (OMF) for data files, log files, and control files. When you create data files and log files, you cannot specify the physical file names. By default, Oracle delivers these scripts using physical file paths, so they must be updated to the OMF format. Reference Amazon RDS User Guide [Creating and Sizing Tablespaces](#) for more information.

SYSDBA SQL Commands

When you create a DB instance in Amazon RDS, the master account used to create the instance gets DBA user privileges (with some limitations). Use this account for any administrative tasks such as creating additional user accounts in the database. The SYS user, SYSTEM user, and other administrative accounts cannot be used. These commands have been identified below along with the RDS procedure to properly run. Reference Amazon RDS User Guide [Granting SELECT or EXECUTE Privileges to SYS Objects](#) for more information.

Create DB SQL [createdb.sql]

Skip this script. It does not need to be modified nor run because you already created the database.

UTL Space Script [utlspace.sql]

Modify the create tablespace commands for creating the PSDEFAULT tablespace to OMF format.

Delivered:

```
CREATE TEMPORARY TABLESPACE PSTEMP
TEMPFILE      '/u03/oradata/<SID>/pstemp01.dbf'      SIZE 300M
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K
;

CREATE TABLESPACE          PSDEFAULT
DATAFILE      '/u03/oradata/<SID>/psdefault.dbf'    SIZE 100M
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
SEGMENT SPACE MANAGEMENT AUTO
;
```

Modified:

```
CREATE TEMPORARY TABLESPACE PSTEMP TEMPFILE EXTENT
MANAGEMENT LOCAL UNIFORM SIZE 128K
;

CREATE TABLESPACE PSDEFAULT EXTENT MANAGEMENT LOCAL
AUTOALLOCATE SEGMENT SPACE MANAGEMENT AUTO
;
```

Application-Specific Tablespace Creation [`xxddl.sql`]

Modify the application-specific tablespace creation script for the PeopleSoft application you are creating. For example, `epddl.sql` for FSCM or `hcddl.sql` for HCM. Please refer to the PeopleTools Installation for Oracle for details on the DDL scripts that are appropriate for your application. Modify all `CREATE TABLESPACE` commands as below.

Delivered:

```
CREATE TABLESPACE AMAPP DATAFILE
'/u04/oradata/<SID>/amapp.dbf' SIZE 2M

EXTENT MANAGEMENT LOCAL AUTOALLOCATE

SEGMENT SPACE MANAGEMENT AUTO

/
```

Modified:

```
CREATE TABLESPACE AMAPP EXTENT MANAGEMENT LOCAL AUTOALLOCATE
SEGMENT SPACE MANAGEMENT AUTO

/
```

DB Owner Script [`dbowner.sql`]

Replace `system/manager` with the RDS Master username and Master password.

Delivered:

```
CONNECT system/manager;
```

Modified (Replace with your credentials):

```
CONNECT <RDS Master username>/<RDS Master password>;
```

PS Roles Script [psroles.sql]

Modify all grants below, if present.

Delivered:

```
GRANT SELECT ON V_$MYSTAT to PSADMIN;  
GRANT SELECT ON USER_AUDIT_POLICIES to PSADMIN;  
GRANT SELECT ON FGACOL$ to PSADMIN;  
grant execute on dbms_refresh to PSADMIN;  
GRANT SELECT ON ALL_DEPENDENCIES to PSADMIN;
```

Modified:

```
exec  
rdsadmin.rdsadmin_util.grant_sys_object('V_$MYSTAT', '  
PSADMIN', 'SELECT');  
  
exec  
rdsadmin.rdsadmin_util.grant_sys_object('USER_AUDIT_P  
OLICIES', 'PSADMIN', 'SELECT');  
  
exec  
rdsadmin.rdsadmin_util.grant_sys_object('FGACOL$', 'PS  
ADMIN', 'SELECT');  
  
exec  
rdsadmin.rdsadmin_util.grant_sys_object('DBMS_REFRESH  
, 'PSADMIN', 'EXECUTE');  
  
exec  
rdsadmin.rdsadmin_util.grant_sys_object('ALL_DEPENDEN  
CIES', 'PSADMIN', 'SELECT');
```

PS Admin Script [`psadmin.sql`]

Replace system/manager with the Amazon RDS master username and master password.

Delivered:

```
connect system/manager
```

Modified (Replace with your credentials):

```
connect <RDS Master username>/<RDS Master password>
```

Connect Script [`connect.sql`]

No changes necessary.

Execute Database creation scripts

After all the scripts have been updated, execute on the database as usual as per the PeopleSoft installation guide.

Create and Run Data Mover Import Scripts

Follow the PeopleSoft installation guide for creating and running the Data Mover Import scripts for your PeopleSoft Application.

Conclusion

We have now completed the setup of a PeopleSoft demo database that is fully managed on Amazon RDS and ready to perform. This guide described how to configure Amazon RDS for Oracle as a backend database for an Oracle PeopleSoft Enterprise demo application. By using these procedures, you can use Amazon RDS for Oracle to set up and operate many different PeopleSoft application databases. As a result, you have the steps to run your PeopleSoft applications on an Amazon RDS for Oracle Database.

References

- PeopleTools 8.57 Installation for Oracle
- Amazon Web Services API Reference

Contributors

The following individuals and organizations contributed to this document:

- David Brunet, VP Research and Development, DLZP Group
- Nick Sefiddashti, AWS Solutions Architect, DLZP Group
- Muhammed Sajeed, PeopleSoft Architect, DLZP Group
- Yoav Eilat, Senior Product Marketing Manager, AWS
- Tsachi Cohen, Software Development Manager, AWS
- Michael Barras, Senior Database Engineer, AWS

Document Revisions

Date	Description
March 2017	First publication
July 2019	Updated with the latest features of Amazon RDS, PeopleSoft release versions, AWS Management Console updates

Notes

¹ <https://docs.aws.amazon.com/general/latest/gr/rande.html>

² <https://docs.aws.amazon.com/AmazonVPC/latest/GettingStartedGuide/GetStarted.html>

³ https://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/VPC_IAM.html

⁴ <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/UsingIAM.html>

⁵ <https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/UsingWithRDS.IAM.AccessControl.IdentityBased.html>

⁶ <https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/auth-and-access-control.html>