

Oracle Application Express and Amazon Cloud

*An Oracle White Paper
April 2009*

Note: The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Oracle Application Express and Amazon Cloud

Terminology	2
Introduction	3
Overview	3
Required Software	3
Delploying Oracle APEX On Amazon Cloud System	4
Creating EBS Volume And S3 Configuration	6
Conclusion.....	10

TERMINOLOGY

Amazon Cloud	Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers. One can host virtual machines in the cloud.
AMI	Amazon Machine Image . To use Amazon EC2, create an Amazon Machine Image (AMI) containing your applications, libraries, data and associated configuration settings.
EBS	Amazon Elastic Block Store (EBS) provides block level storage volumes for use with Amazon EC2 instances.
S3	Amazon S3 is storage for the Internet. It is designed to make web-scale computing easier for developers.
AWS Account	To access any Amazon Web Services you must first create an AWS Account at http://aws.amazon.com .

INTRODUCTION

Oracle Application Express, a feature of the Oracle database 11g, combines rapid web application development with the power of the Oracle database. Its easy to use browser based application builder enables developers and non-programmers to develop and deploy data driven web applications in very little time.

Oracle APEX doesn't depend on any client software for developing or deploying web applications. Simple architecture, and browser-based features make the transition for developers and end-users seamless- simply provide the URL for the cloud environment.

The multi-tenant capabilities allow multiple users and their associated applications to co-exist within one Oracle Database, minimizing cost. Only one instance is needed and users work in a dedicated work area called a workspace. Taking a database backup also takes a backup of application source. Oracle Application Express can integrate with other applications by consuming Web services.

OVERVIEW

This whitepaper intends to explain how to start an 11gR1 instance in the cloud and install Oracle Application Express 3.2 on that instance. Also, it explains how to modify the original installation to use EBS and S3 so that your data is persistent and you can switch between different levels of service with EC2.

To start with you need to get an AWS account and sign up for [EC2](#).

Required Software

You need the following software to complete the steps mentioned in this whitepaper:

1. [Elasticfox](#) is a Firefox plug-in, which is an extension for managing the Amazon EC2 account. It requires Apache License 2.0. Go through the [Getting Started Guide](#) for more details. This guide contains instructions on associating your AWS EC2 account with Elasticfox and creating a KeyPair to identify yourself when connecting to your machine.
2. [Putty and PuttyGen](#) to connect to your instance via SSH. No license is required for this.
3. FTP client for file transfer like [WinSCP](#). WinSCP is an open source SFTP client and FTP client for Windows available via the GNU General Public License.
4. [Oracle Application Express 3.2](#) is the latest version of Oracle APEX and is a “no-cost” option of the Oracle Database.

As mentioned in the [Getting Started Guide](#), convert the KeyPair to a format that can be used by Putty and WinSCP. PuttyGen will convert the key to the proper format.

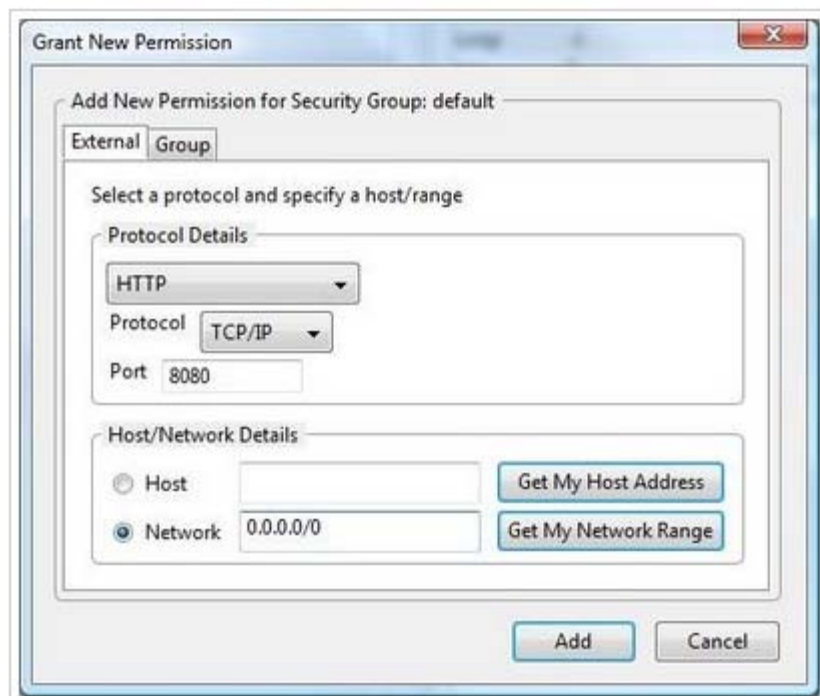
DELPOYING ORACLE APEX ON AMAZON CLOUD SYSTEM

Once you have the AWS account and the required software view the [Deploying Oracle Database in the Cloud](#) viewlet that is available on the [Oracle Cloud Computing Center](#) page on OTN. The viewlet shows how to find and start image ami-cecb2fa7, which is 32bit Enterprise Linux with 11g. For this whitepaper we chose to start a c1.medium instance type, which costs 20 cents/hr, described as:

High-CPU Medium Instance 1.7 GB of memory, 5 EC2 Compute Units (2 virtual cores with 2.5 EC2 Compute Units each), 350 GB of instance storage, 32-bit platform

To configure your instance for Oracle APEX, add permissions by following these steps:

1. First identify the Security Group in Elasticfox. It would be “default” unless another Security Group exists.
2. Click the Security Group tab, highlight default and click the green checkmark under the Group Permissions pane to add a permission.



3. Allow your host to connect on port 22 and then add another permission to allow anyone to connect on port 8080, the default port for the EPG.
4. Choose your network and allow 0.0.0.0/0 for port 8080.

PuTTYgen and PuTTY are now used to connect to this machine by following these steps:

1. Start PuTTYgen
2. Click 'Load', browse to find the .pem KeyPair file created with Elasticfox.
3. Click Save Private to save it as a .ppk file and start PuTTY
4. Enter the Public DNS name given above for the Host Name, Port '22', Connection Type 'SSH'.
5. If behind a firewall enter the Proxy settings, under Category -> Connection -> Proxy.
6. Specify the KeyPair within SSH, using Category -> Connection -> SSH -> Auth, and browsing for the KeyPair (.ppk) file saved with PuTTYgen.
7. Click 'Open'. When the Putty screen appears, enter 'login as:' of 'root'. The KeyPair will be used to authenticate your session.
8. Review the license agreements and accept if appropriate by entering 'y'.
9. Enter the "oracle" password
10. You will be prompted if you want to create a database now. Enter 'y' as this is a new instance.
11. Select defaults for disk devices to store your datafiles, redo logs and flash recovery area files on. Enter 'y' when prompted if your details are correct and to continue.
12. Enter your Oracle SID and passwords for SYS, SYSTEM, DBSNMP, SYSMAN, and APEX ADMIN.

Creating and configuring your new database instance will now complete.

Your database is fully operational and APEX 3.0.1 the default Oracle APEX release with 11g instance is also installed. Follow the steps mentioned below to upgrade to the latest version of Oracle APEX 3.2.

1. Start your FTP client, such as WinSCP and connect to your instance
2. Copy the apex_3.2.zip file to your instance

3. From your Putty terminal, login and unzip the installer for Oracle Application Express 3.2, install APEX 3.2, upgrade the images and set the ADMIN password:

```
su - oracle
```

```
unzip apex_3.2.zip
```

```
cd apex
```

```
sqlplus / as sysdba;
```

```
SQL>@apexins SYSAUX SYSAUX TEMP /i/
```

```
SQL>@apxldimg /home/oracle
```

```
SQL>@apxchpwd <enter ADMIN password when prompted>
```

4. Login to the Oracle Application Express as an instance administrator with username 'ADMIN' and the password you specified in step 3 above and create a workspace. The URL for the internal administration application is http://<host>:8080/apex/apex_admin. Create new workspace for your group.

CREATING EBS VOLUME AND S3 CONFIGURATION

The Amazon EC2 environment configured as mentioned above stores all the datafiles within the same environment and if you were to shutdown the instance everything would be gone. Amazon's [Elastic Block Storage](#) (EBS) allows you to create persistent volumes that can be used with your EC2 instance. Once configured, you can shutdown the EC2 instance, change EC2 instance type if necessary, and then simply mount your EBS volume when starting up the new instance. Following steps show how to create your EBS volume:

1. Create an EBS volume, as mentioned in the [Elasticfox Getting Started Guide](#). Make sure that you create your EBS volume in the same availability zone as the instance you have started.
2. Attach the EBS volume to your instance. Go to Instances tab in Elasticfox, right click on your instance and choose to attach an EBS volume. Give it a device name e.g. /dev/sdf1.
3. Next format the volume as a file system before it could be used. As root issue the command:

```
$ mkfs.ext3 <EBS device name> (e.g /dev/sdf1)
```

4. Mount it to a temporary place as shown below:

```
$ mount <EBS device name> /mnt
```

```
df -m {last entry should be something similar to u02}
```

5. Now shut down the database and copy all the necessary files for the database including the control files, datafiles, redo log files etc, to the temporary mount point on /mnt.

```
$ su - oracle
```

```
$ sqlplus / as sysdba
```

```
SQL> shutdown immediate
```

```
SQL> exit
```

```
$ cp /u02/* /mnt/
```

```
$ exit {this will return to the root user}
```

6. Now unmount the volumes, remount the EBS volume and start the database as shown below:

```
$ umount <EC2 device name> (e.g. /dev/sda2)
```

```
$ umount <EBS device name> (e.g /dev/sdf1)
```

```
$ mount <EBS device name>
```

```
$ su - oracle
```

```
$ sqlplus / as sysdba;
```

```
SQL> startup;
```

7. The database should start successfully. If the database does not start then repeat the above steps and don't proceed.

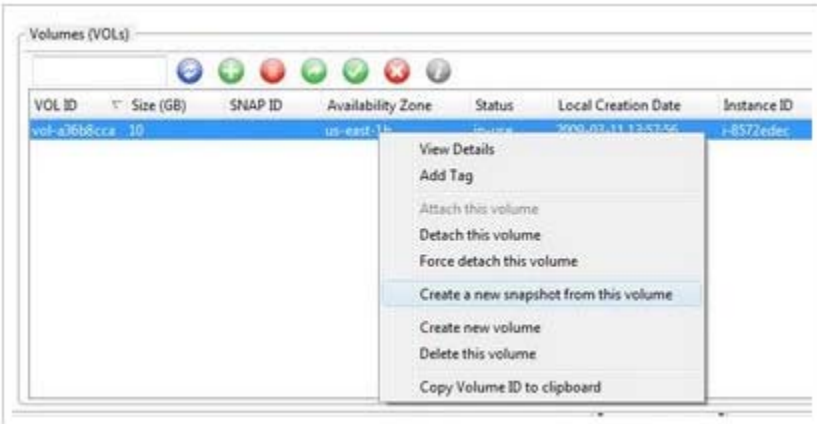
Now, shutdown the database and unmount /dev/sdf1 (the device name you chose in step 2 above).

```
SQL> shutdown immediate
```

```
SQL> exit
```

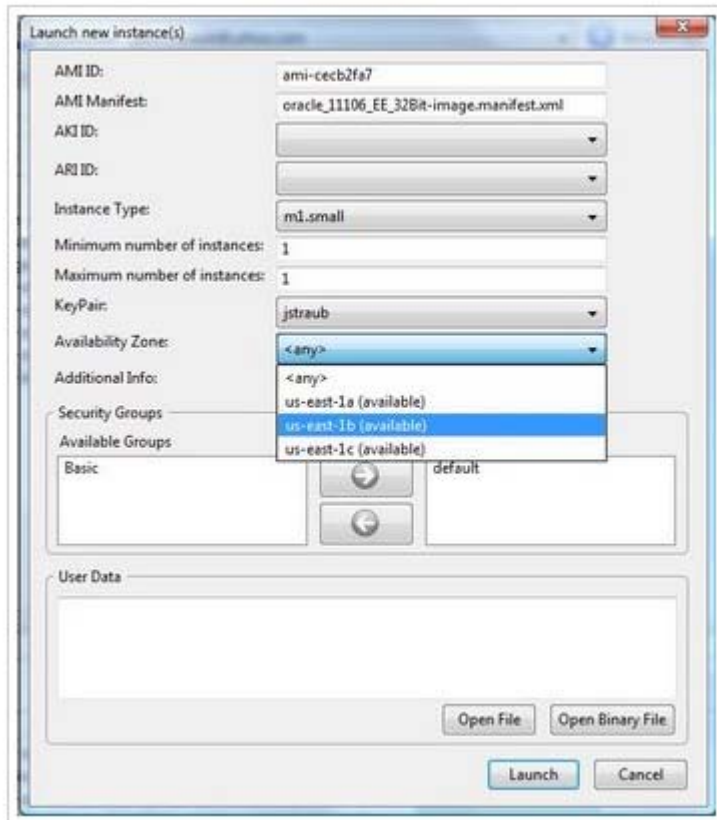
```
$ umount <EBS device name> (e.g /dev/sdf1)
```

Create a snapshot of the volume, which will be stored on Amazon [Simple Storage Service](#) (S3). This snapshot is a point-in-time backup of the EBS volume. This gives the flexibility of shutting down all instances, deleting EBS volumes and pay only for the S3 storage thus reducing the cost as well.



The following steps show how to create an EC2 instance using this snapshot:

1. Start a new AMI and configure the instance using the same steps as given earlier.
2. Choose the same availability zone as your EBS volume.



3. Connect to the instance using Putty and choose 'No' when asked to create a database.
4. Now create another directory e.g. /u02, attach the EBS volume under this instance, and then mount it to /u02. To start the database set the ORACLE_SID environment variable and create symlinks in \$ORACLE_HOME/dbs to the spfile and password file in the admin directory on the /u02, as shown below:

```
$ mkdir /u02
$ mount <EBS device name> /u02 (e.g /dev/sdf1)
$ su - oracle
$ export ORACLE_SID=orcl
$ cd $ORACLE_HOME/dbs
$ ln -s /u02/admin/orcl/dbs/spfileorcl.ora spfileorcl.ora
$ ln -s /u02/admin/orcl/dbs/orapworcl orapworcl
$ sqlplus / as sysdba
SQL> startup
```

5. This starts your database on the new smaller machine.

The total cost for configuring Oracle APEX on Amazon should be less than \$1.00. Depending on the level of service, amount of storage, and bandwidth used ongoing costs will vary.

CONCLUSION

Since APEX has a simple architecture and is completely browser based it makes the transition to a cloud instance for developers and end-users seamless - simply provide the URL for the cloud environment. Using S3 for storage allows the administrator to easily manage and change the EC2 level of service as required. The admin defines the service and then mounts the Oracle Database on EBS from S3.

This allows maximum flexibility, for example:

- Turn off development instance over the weekend when developers not working
- Change from Large Instance to Small Instance for Production applications after hours
- As application workload increases upgrade from Medium Instance to Large Instance

If you wish to migrate any applications developed on the cloud onto your internal systems, ensuring that the correct version of APEX is installed on the target server and export the APEX application, data schema and data from Amazon and then import them on the target server.



Oracle Application Express and
Amazon Cloud
April 2009
Author: Priyanka Sharma
Contributing Authors: Jason Straub

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
www.oracle.com

Oracle Corporation provides the software
that powers the internet.

Oracle is a registered trademark of Oracle Corporation. Various
product and service names referenced herein may be trademarks
of Oracle Corporation. All other product and service names
mentioned may be trademarks of their respective owners.

Copyright © 2002 Oracle Corporation
All rights reserved.