Deploying a Highly Available Enterprise Manager 12c Cloud Control
Product Overview................................................................. 2
Introduction ........................................................................... 2
Cloud Control Architecture...................................................... 3
Implementation of a Level 3 MAA Setup................................. 4
  Cloud Control Infrastructure.................................................. 6
  Preparing for Cloud Control Installation ............................. 7
  Step 1: Install Cloud Control on primary OMS server .......... 9
  Step 2: Configure the Server Load Balancer (SLB)............. 13
  Step 3: Add Repository Database Targets to Cloud Control ... 19
  Step 4: Configure Software Library .................................... 20
  Step 5: Add Second OMS .................................................... 20
  Step 6: Add Standby Database ........................................... 26
  Step 7: Convert Standby Database to RAC ......................... 34
Conclusion .............................................................................. 39
Product Overview

Oracle Enterprise Manager is Oracle’s integrated enterprise IT management product line and provides the industry’s first complete cloud lifecycle management solution. Oracle Enterprise Manager’s Business-Driven IT Management capabilities allow you to quickly set up, manage and support enterprise clouds and traditional Oracle IT environments from applications to disk. Enterprise Manager allows customers to achieve:

- **Best service levels for traditional and cloud applications** through management from a business perspective including Oracle Fusion Applications

- **Maximum return on IT management investment** through the best solutions for intelligent management of the Oracle stack and engineered systems

- **Unmatched customer support experience** through real-time integration of Oracle’s knowledgebase with each customer environment

Introduction

With Enterprise Manager Cloud Control, Oracle has taken a unique approach to systems management, allowing organizations to deploy a single tool with a tightly integrated set of features to manage all tiers in the datacenter as well as the entire lifecycle of applications. By using Cloud Control, organizations are able to lower the cost of managing applications while at the same time dramatically improving quality of service.

Because of this unique approach to systems management, Cloud Control is far more critical in the data center than the other management tools that are typically found. As such, High Availability has become a key requirement for many Cloud Control deployments as the impact of a Cloud Control outage is much more significant when compared to loss of a point-solution tool. Without access to Cloud Control, administrators are left unaware of the health of their business critical applications and are also unable to undertake many of their day-to-day tasks.

A Cloud Control deployment includes a wide variety of components, and in a highly-available installation each must be considered. This presents a number of possibilities for the overall deployment architecture for a highly available implementation. This whitepaper will detail the steps required to setup a highly available Cloud Control in a Maximum Availability Architecture (MAA) Level 3 configuration. The configuration outlined ensures that performance and availability are maintained, while keeping costs contained. The steps include a number of best practice recommendations and are sequenced in such a way that the overall number of deployment steps and reconfigurations while building the configuration are kept to a minimum. The setup steps also make
use of automated processes where possible, thus further reducing the time taken to setup Cloud Control and minimizing the risk of human error.

Cloud Control Architecture

Cloud Control provides a central point for monitoring and administration in the data center. To achieve this, it collects information from a variety of distributed components and consolidates it in a centralized repository. These components must all work in harmony for the Cloud Control system to operate correctly. The components and information flows involved in collecting, processing and presenting this information are as follows:

- **Oracle Management Agents (Agents)** – The Oracle Management Agent is a software component that is installed on every monitored host in the enterprise. Agents collect information from the targets running on the host and send this information to the Oracle Management Service (OMS). Agents also perform operations against the targets on behalf of Cloud Control users. There are many different types of targets that Cloud Control can manage. Examples include Host, Database, Listener, ASM, WebLogic Server, Service Bus and Fusion Applications components.

- **Oracle Management Service (OMS)** – The Oracle Management Service is the central component in Cloud Control. It is the component with which all other components interact (see Figure 1). The OMS is deployed on WebLogic Server and must be available in order for the agents to upload data and for administrators to access the Cloud Control console.

- **Oracle Management Repository (Repository)** – The Oracle Management Repository is used as a persistent data store. Examples of the information stored in the repository include user information, job definitions, monitoring and alerting settings and all configuration and monitoring data related to targets. The OMS depends on the repository being available, and as such Cloud Control cannot run if the repository is unavailable.

- **Oracle Software Library** – The Software Library is a filesystem repository that stores software entities such as software patches, virtual appliance images, reference gold images, application software, and their associated directive scripts. The software library is accessed by the OMS and is used extensively by the Cloud Control framework for features such as self-update and agent-push.

- **Console** – The Console is a browser-based web application that is the main user interface for Cloud Control. This console allows the administrator to monitor, manage and report on the Cloud Control targets that have been setup.

- **Enterprise Manager Command Line Interface (EMCLI)** – EMCLI allows users to access Cloud Control functionality either interactively from a command line, or as part of a script. This allows Cloud Control operations to be integrated with complex business processes without user interaction.
Implementation of a Level 3 MAA Setup

There are many different configuration options available and these will determine the availability provided by the Cloud Control system. When architecting a highly available Cloud Control implementation, consideration should be given to each tier as all tiers need to be available and work in concert for the system to function as a whole.

To simplify the design of highly available Cloud Control implementations, 4 basic configuration levels are described in the Cloud Control documentation. These 4 configurations start at Level 1 and progresses through to Level 4, with each level providing increased availability over the previous one. Level 1 provides the least protection against planned or unplanned outages as there is just a single OMS and a single database, and no redundant components are configured. As such, failure of either the OMS or the Repository would result in the system being unavailable until such a time that the component could be recovered. In contrast to this, Level 4 provides a significantly higher level of protection which is made possible with the use of redundant components installed across multiple physical locations. The levels of availability are briefly summarized in the table below:

<table>
<thead>
<tr>
<th>Level</th>
<th>Sites</th>
<th>Description</th>
<th>Load Balancer Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Single Site</td>
<td>OMS and Repository hosts configured on a single site. Each resides on their own host with no failover.</td>
<td>None</td>
</tr>
<tr>
<td>2</td>
<td>Single Site</td>
<td>Pair of OMSs installed in active/passive mode on shared storage with VIP based failover. Repository hosts configured with Local Physical Standby Database.</td>
<td>None</td>
</tr>
<tr>
<td>3</td>
<td>Single Site</td>
<td>Multiple OMSs deployed in Active/Active configuration with a Server</td>
<td>Local Load</td>
</tr>
</tbody>
</table>
This document will describe the implementation of a Level 3 MAA setup. This offers a very high level of protection within a single site.

The following diagram outlines a Level 3 configuration. As illustrated, this configuration comprises multiple Management Services accessed through a local server load-balancer (SLB) and a Repository database that uses Oracle Real Application Clusters. Oracle recommends that the Repository and any active OMSs are located in close proximity to one another as increased latency between the OMS and Repository tiers will impact the overall performance of Cloud Control. As a Level 3 configuration has multiple active OMS and Repository servers it provides continuous availability when either a database host or OMS host fails. Furthermore, a Level 3 configuration utilizes a Data Guard standby database. The standby database offers protection for the database tier in the event that the database storage should fail.

It should be noted that a Level 3 configuration does not protect against site failure. If protection against site failure is required a Level 4 setup should be considered.
Figure 2: Cloud Control Level 3 MAA Deployment

For more information regarding the various levels of availability refer to the Enterprise Manager Cloud Control documentation.

Cloud Control Infrastructure

The hardware used to setup this Cloud Control installation is as follows:

- 2 node Linux Cluster for Primary repository DB
- 2 node Linux Cluster for Standby repository DB
- 2 Linux servers for OMS
- F5 SLB
- NFS storage server for Software Library
The steps for building the above configuration are outlined in the following flowchart:

Preparing for Cloud Control Installation

Prior to implementation, some prerequisite steps must be done on the OMS and Repository nodes. These steps are outlined below.

**Prepare Database Clusters**

The first step that should be taken prior to a Level 3 Cloud Control implementation is the configuration of the Primary and Standby database clusters. The instructions in this document assume that primary and standby clusters have already been configured.

A Level 3 setup provisions a standby cluster on the same site as the primary cluster. The standby cluster provides protection against failure of the entire primary database cluster. By configuring the standby database environment on identical hardware to the primary environment, it is possible to run Cloud Control at full capacity in the event of a failover to the standby.

We configured 2 clusters as follows:

- Primary and Standby clusters both running Oracle Enterprise Linux 5.6 (x86-64)
- Oracle Clusterware 11g Release 2 (11.2) binaries installed and configured on primary and standby clusters
- Oracle Database 11g Release 2 (11.2) binaries installed on primary and standby clusters
• SCAN listeners configured on Primary and Standby clusters

• Primary system names of emrep1 and emrep2 forming cluster emrep-cl

• Standby system names of emreps1 and emreps2 forming cluster emreps-cl

• As per Oracle Best Practice, the Primary and Standby clusters were each configured with ASM disks for shared database storage.

• ‘DATA’ (Data) and ‘FRA’ (Fast Recovery Area) ASM Diskgroups were configured and available on the Primary and Standby clusters. As per Oracle Best Practice, these were configured with EXTERNAL redundancy as the underlying storage hardware supports redundancy.

Oracle database 11gR2 was used as it allows for the use of the Single Client Access Name (SCAN) for client connections. If possible, it is recommended to use a SCAN address as it allows for the addition and removal of database cluster nodes without reconfiguration of the OMS database connections.

For more information on Single Client Access Name refer to the Oracle Database 11g Release 2 Real Application Clusters Administration and Deployment Guide.

Prepare Repository Database

During the installation of Cloud Control the installer will prompt the user to specify a database to be used as the Cloud Control repository. If your planned configuration uses a RAC database as a repository, it is recommended to create your RAC database prior to the installation of Cloud Control. This approach - as opposed to installing using a single instance database and then converting to RAC - helps to reduce the overall steps and time taken to complete the configuration.

We created a database to be used as the primary Cloud Control repository on the primary database cluster with the name ‘emrep’. This database consisted of the emrep1 and emrep2 instances. The datafiles, redologs and controlfiles were placed on the ‘DATA’ shared ASM diskgroup.

In addition to meeting the requirements specified in the Enterprise Manager Cloud Control Basic Installation Guide, we configured the database in ARCHIVELOG mode and enabled Flashback Database. It is recommended that these options are enabled when the database is created as by doing so it is possible to avoid having to reconfigure the database when creating and managing the Standby Database later on.

For further information and recommendations on the prerequisites for creating the repository database on RAC refer to the following documentation:

• **Clusterware Administration and Deployment Guide**
• **Real Application Clusters Administration and Deployment Guide**
• **Automatic Storage Management Administrator’s Guide**
• **Enterprise Manager Cloud Control Basic Installation Guide**
• **Oracle Enterprise Manager Cloud Control 12c Sizing Guidelines**
Prepare OMS Nodes

The final step that we needed to complete prior to installing the Cloud Control software was to prepare the OMS nodes.

When creating a highly available OMS tier, more than one management service must be configured. We had a pair of Linux servers that would be used as OMS servers. These were configured as follows:

- Both servers running Oracle Enterprise Linux 5.6 (x86-64)
- System names were oms1 and oms2
- An NFS location accessible from both servers was configured and mounted as /cc12_SWLlib

For full details of the requirements for the OMS see the Enterprise Manager Cloud Control Basic Installation Guide

Following the successful preparation of the database clusters and OMS nodes, the installation of Cloud Control could be started using the Oracle Installer.

Step 1: Install Cloud Control on primary OMS server

The Cloud Control installation should be started from the first node that will be configured as an OMS server.

In our example, we staged the installation media and started the installer on the host named oms1. We chose to create a new Enterprise Manager System using Advanced installation. The Middleware Home was specified as /u01/Middleware which was an empty directory on the OMS host:

In Step 5 in the installer, we chose not to install any additional plug-ins at this time.

In Step 6 we specified passwords for the WebLogic Domain and Node Manager
In Step 7 we provided the login credentials for the repository database that we created. We specified one of the database hosts and specified the service/SID as 'emrep'.

Tip: Entering a cluster database instance in Step 7 will prompt for the modification of the database connect string. If using Oracle Database 11gR2, it is recommended to specify a connect string that uses the SCAN address when prompted. By using the SCAN address to connect to the database it is possible for nodes to be added and removed from the RAC cluster without having to subsequently change the connect string that the OMSs use.

As we are using a cluster database on Oracle Database 11g Release 2, we modified the connect string to specify the SCAN address as follows:

```sql
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = emrep-cl-scan.example.com)(PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = emrep)
  )
)
```

For the repository configuration details in Step 9, we specified the previously created ASM diskgroups as the location for the Management Tablespace, Configuration Data Tablespace and JVM Diagnostics Data Tablespace.
When prompted to specify the ports for this OMS, it is recommended to specify ports that are also free on other servers that will be used as OMS hosts. This helps to simplify the load balancer setup later on.

In Step 10, we configured ports as follows during the installation process:

Of these ports, the following are relevant for the SLB configuration later on:

- Enterprise Manager Upload HTTP Port: 4889
- Enterprise Manager Upload HTTP SSL Port: 4900
- Enterprise Manager Central Console HTTP Port: 7788
- Enterprise Manager Central Console HTTP SSL Port: 7799

Upon successfully completing the installation a summary screen describing how to access the Cloud Control installation is presented. The information presented on this screen should be noted.
Immediately following the installation the initial OMS configuration should be verified to determine the configuration details such as security setup, ports used and load balancer setup.

This can be done by issuing the “emctl status oms –details” command from the OMS server:

```
$ ./emctl status oms –details
Oracle Enterprise Manager Cloud Control 12c Release 12.1.0.1.0
Copyright (c) 1996, 2011 Oracle Corporation. All rights reserved.
Enter Enterprise Manager Root (SYSMAN) Password :
Console Server Host : oms1.example.com
HTTP Console Port   : 7788
HTTPS Console Port  : 7799
HTTP Upload Port    : 4889
HTTPS Upload Port   : 4900
OMS is not configured with SLB or virtual hostname
Agent Upload is locked.
OMS Console is locked.
Active CA ID: 1
Console URL: https://oms1.example.com:7799/em
Upload URL: https://oms1.example.com:4900/empbs/upload

WLS Domain Information
Domain Name      : GCDomain
Admin Server Host: oms1

Managed Server Information
Managed Server Instance Name: EMGC_OMS1
Managed Server Instance Host: oms1.example.com

The OMS application traffic includes browser-OMS traffic (ie. the browser traffic created by users accessing Cloud Control) and agent-OMS traffic (ie. the traffic created by the agents uploading their data to the OMS). Both browser-OMS traffic and agent-OMS traffic can be configured to use either HTTP or HTTPS.

To ensure secure communication between Cloud Control components, it is recommended to use HTTPS for all agent-OMS and browser-OMS traffic.

The output above shows that Agent Upload and OMS console ports are already locked, and therefore using HTTPS. As this is the case no further action needs to be taken here. It also shows that the OMS is not currently configured with an SLB or virtual hostname. The Console and Upload URLs indicate that the application is accessed directly through the physical host that the OMS was installed on (oms1).

It is recommended that the repository connect string that is used by the OMS to connect to the database server is checked using the “emctl config oms –list_repos_details” command:

```
$ ./emctl config oms –list_repos_details
Oracle Enterprise Manager Cloud Control 12c Release 12.1.0.1.0
Copyright (c) 1996, 2011 Oracle Corporation. All rights reserved.
```
Repository Connect Descriptor:

```plaintext
(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=emrep-cl-scan.example.com)(PORT=1521)))
(CONNECT_DATA=(SERVICE_NAME=emrep)))
```

Repository User: SYSMAN

The above output shows that the SCAN address specified during the initial OMS install is being used. Final verification that the OMS is operating correctly can be done by logging in to Cloud Control. In our case we used the following URL:

[https://oms1.example.com:7799/em](https://oms1.example.com:7799/em)

After Step 1, the Cloud Control topology is as follows:

![Cloud Control Topology After Installation of First OMS](image)

As shown in the diagram above, the Repository tier is protected from node failure with the use of RAC, however if the OMS node is lost then the application will be unavailable until such a time that it can be recovered.

**Step 2: Configure the Server Load Balancer (SLB)**

As shown above, the install of the first OMS configures the system so that the Cloud Control users and agents connect directly to the OMS using its physical hostname. In a highly available Cloud Control configuration, multiple OMS servers are present and users and agent should connect to the OMSs via a load balancer which is able to direct traffic to available management services.

SLB configuration should be done immediately after installing the first OMS.

Our SLB was an F5 BIG-IP Local Traffic Manager running 11.1.0, Build 1943.0 Final. As our OMS is configured only for secure console and upload traffic, we only needed to configure the Secure Upload and Secure Console services on the SLB.
SLB setup consisted of configuring:

- Health Monitors
- TCP Profiles
- Pools
- Persistence Profile (Console service only)
- Virtual Servers

We also registered the virtual server IP address and hostname (oms.example.com) in the DNS servers, so that client requests could reference the hostname rather than the IP address.

The table below summarizes the F5 objects that were created during the SLB setup:

<table>
<thead>
<tr>
<th>CLOUD CONTROL SERVICE</th>
<th>TCP PORT</th>
<th>MONITOR NAME</th>
<th>PERSISTENCE</th>
<th>POOL NAME</th>
<th>LOAD BALANCING</th>
<th>VIRTUAL SERVER NAME</th>
<th>VIRTUAL SERVER PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Upload</td>
<td>4900</td>
<td>mon_ccsu4900</td>
<td>None</td>
<td>pool_ccsu4900</td>
<td>Round Robin</td>
<td>vs_ccsu4900</td>
<td>4900</td>
</tr>
<tr>
<td>Secure Console</td>
<td>7799</td>
<td>mon_csc7799</td>
<td>Source IP</td>
<td>pool_csc7799</td>
<td>Round Robin</td>
<td>vs_csc443</td>
<td>443</td>
</tr>
</tbody>
</table>

**Step 2.1: Create Health Monitors**

Health Monitors check the status of a service on an ongoing basis, at a set interval. If the service being checked does not respond within a specified timeout period, or the status of the service indicates that the performance has degraded, the system automatically takes it out of the pool and will choose other members of the pool.

The Health Monitors were configured using the settings in the table below:

<table>
<thead>
<tr>
<th>CLOUD CONTROL SERVICE</th>
<th>TCP PORT</th>
<th>MONITOR NAME</th>
<th>TYPE</th>
<th>INTERVAL</th>
<th>TIMEOUT</th>
<th>SEND STRING</th>
<th>RECEIVE STRING</th>
</tr>
</thead>
</table>
| Secure Console        | 7799     | mon_csc7799  | https   | 5        | 16      | GET /em/console/home HTTP/1.1
HTTP/1.1 Host: /em/login.jsp
Connection: Close

| Secure Upload         | 4900     | mon_ccsu4900 | https   | 60       | 181     | GET /empbs/upload 'n' /empbs/upload 'n'          | Http Receiver Servlet active!  |

**Step 2.2 Create TCP Profiles**

TCP Profiles are created to control the behavior of Cloud Control traffic.

We created two TCP Profiles, one for the Secure Console service and one for the Secure Upload service.

These were created with the default settings for a TCP Profile.
Step 2.3 Create Pools

A pool is a set of servers grouped together to receive traffic on a specific TCP port using a load balancing method. Each pool can have its own unique characteristic for a persistence definition and the load-balancing algorithm used. The preferred setting of the load balance algorithm for all Cloud Control pools is Least Connections (Member).

We created pools on the load balancer as follows:

<table>
<thead>
<tr>
<th>CLOUD CONTROL SERVICE</th>
<th>POOL NAME</th>
<th>ASSOCIATED HEALTH MONITOR</th>
<th>Load Balancing</th>
<th>MEMBERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Console</td>
<td>pool_ccsc7799</td>
<td>mon_ccsc7799</td>
<td>Least Connections (member)</td>
<td>oms1.example.com:7799</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>oms2.example.com:7799</td>
</tr>
<tr>
<td>Secure Upload</td>
<td>pool_ccsu4900</td>
<td>mon_ccsu4900</td>
<td>Least Connections (member)</td>
<td>oms1.example.com:4900</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>oms2.example.com:4900</td>
</tr>
</tbody>
</table>

Even though oms2 has not been configured yet, it is recommended to add the second OMS host to the server pools now as it means modifying the SLB configuration subsequent to the installation of the second OMS can be avoided.

Step 2.4 Create Console Persistence Profile

A console persistence profile is required to ensure that all Cloud Control user requests for a given session are directed to the same management service for the entire session. Without a Persistence Profile such as this, user sessions could span multiple OMSs, and require the Cloud Control user to login multiple times.

We created a Persistence Profile with the following attributes:

<table>
<thead>
<tr>
<th>CLOUD CONTROL SERVICE</th>
<th>FS PERSISTENCE PROFILE NAME</th>
<th>TYPE</th>
<th>TIMEOUT</th>
<th>EXPIRATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Console</td>
<td>sourceip_ccsc7799</td>
<td>Source Address Affinity</td>
<td>3600</td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>
Figure 5: Creating Persistence Profile for Secure Console

Step 2.5 Create Virtual Servers

The final load balancer configuration step was to define our virtual servers. A virtual server, with its virtual IP Address and port number, is the client addressable hostname or IP address through which members of a load balancing pool are made available to a client. After a virtual server receives a request, it directs the request to a member of the pool based on a chosen load balancing method.

We created virtual servers for the Secure Console and Secure Upload services using the settings in the table below:

<table>
<thead>
<tr>
<th>CLOUD CONTROL SERVICE</th>
<th>VIRTUAL SERVER NAME</th>
<th>VIRTUAL IP AND PORT</th>
<th>PROTOCOL PROFILE (CLIENT)</th>
<th>HTTP PROFILE</th>
<th>SNAT POOL</th>
<th>iRULE</th>
<th>DEFAULT POOL</th>
<th>DEFAULT PERSISTENCE PROFILE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Console</td>
<td>vs_ccsc443</td>
<td>&lt;virtual Host IP&gt;:443</td>
<td>tcp_ccsc7799</td>
<td>None</td>
<td>Auto</td>
<td>None</td>
<td>pool_ccsc</td>
<td>sourceip_ccsc7799</td>
</tr>
<tr>
<td>Secure Upload</td>
<td>vs_cusu4900</td>
<td>&lt;virtual Host IP&gt;:4900</td>
<td>tcp_cusu4900</td>
<td>None</td>
<td>Auto</td>
<td>None</td>
<td>pool_cusu</td>
<td>None</td>
</tr>
</tbody>
</table>

Step 2.6: Update OMS configuration

After the SLB setup was completed we needed to resecure the OMS using the SLB hostname.

emctl secure oms -sysman_pwd <sysman_pwd>
- reg_pwd <agent_reg_password>
- host oms.example.com
- secure_port 4900
- slb_port 4900
- slb_console_port 443
Following this command the OMS was restarted.

“emctl status --details” output now shows that the OMS is configured against an SLB.

$ ./emctl status oms --details
Oracle Enterprise Manager Cloud Control 12c Release 12.1.0.1.0
Copyright (c) 1996, 2011 Oracle Corporation. All rights reserved.
Enter Enterprise Manager Root (SYSMAN) Password:
Console Server Host: oms1.example.com
HTTP Console Port: 7788
HTTPS Console Port: 7799
HTTP Upload Port: 4889
HTTPS Upload Port: 4900
SLB or virtual hostname: oms.example.com
HTTPS SLB Upload Port: 4900
HTTPS SLB Console Port: 443
Agent Upload is locked.
OMS Console is locked.
Active CA ID: 1
Console URL: https://oms.example.com:443/em
Upload URL: https://oms.example.com:4900/empbs/upload

WLS Domain Information
Domain Name: GCDomain
Admin Server Host: oms1.example.com

Managed Server Information
Managed Server Instance Name: EMGC_OMS1
Managed Server Instance Host: oms1.example.com

The above output shows that the Console and Upload URLs now reference the SLB rather than the physical host of the OMS.

Although the SLB has been configured, the agent that was previously deployed on the OMS is still uploading to the physical hostname of the first OMS server. This can be seen from the output of an “emctl status agent” command:

$ ./emctl status agent
Oracle Enterprise Manager 12c Release 1 12.1.0.1.0
Copyright (c) 1996, 2011 Oracle Corporation. All rights reserved.
-----------------------------------------------------------------
Agent Version: 12.1.0.1.0
OMS Version: 12.1.0.1.0
Protocol Version: 12.1.0.1.0
Agent Home: /u01/Middleware/agent/agent_inst
Agent Binaries: /u01/Middleware/agent/core/12.1.0.1.0
Agent Process ID: 15661
Parent Process ID : 15594
Agent URL : https://oms1.example.com:3872/emd/main/
Repository URL : https://oms1.example.com:4900/empbs/upload

Started at : 2012-02-09 06:48:14
Started by user : oraha
Last Reload : (none)
Last successful upload : 2012-02-21 19:58:44
Last attempted upload : 2012-02-21 20:02:04
Total Megabytes of XML files uploaded so far : 182.23
Number of XML files pending upload : 1,488
Size of XML files pending upload(MB) : 24.28
Available disk space on upload filesystem : 57.28%
Collection Status : Collections enabled

Agent is Running and Ready

In order for this agent to start uploading via the SLB, we performed a resecure on the agent using the SLB hostname:

eemctl secure agent -emdWalletSrcUrl https://oms.example.com:4900/em

Following the resecure of the agent, the Repository URL in the “eemctl status agent” output will reflect the SLB hostname instead of the hostname of the first OMS server.

The final step is to inform the clients that they can now connect to Cloud Control from their browsers using the SLB rather than the physical OMS.

https://oms.example.com/em

EMCLI should also be reconfigured to connect via the SLB at this point. This is done with the following command:

eemcli setup -url=https://oms.example.com/em -username=em_user

For more information regarding configuring EMCLI refer to the Command Line Interface Guide.
After the SLB setup has been completed, the Cloud Control topology is as shown below:

![Cloud Control Topology After Load Balancer Configuration](image)

**Figure 6: Cloud Control Topology After Load Balancer Configuration**

As shown in the diagram, the agents and clients now connect communicate with the OMS via the load balancer.

For further details regarding configuration of Cloud Control with F5 Load Balancers, refer to the Oracle/F5 white paper [Configuring OMS High Availability with F5 Big-IP Local Traffic Manager](https://www.oracle.com/technetwork/articles/omsi-configure-oms-high-availability-f5-big-ip-ltm-191751.html).

**Step 3: Add Repository Database Targets to Cloud Control**

Following the installation of Cloud Control, the RAC hosts that are used for the repository database will not be visible as Cloud Control targets. In order to add them to the Cloud Control environment they should each have an agent installed. We installed the agents by navigating to Setup | Add Target | Add Targets Manually and adding the Hosts using the Add Host Targets wizard.

After the repository hosts are added, the repository itself can be added as a database target by navigating to Targets | Databases and using the add target wizard. As part of this flow, we were prompted to also add the cluster target for the primary database cluster.
Step 4: Configure Software Library

In a highly available Cloud Control installation, the Software Library needs to be accessible from each host that will be used as a management service. Because the Software Library is a critical part of Cloud Control infrastructure, the filesystem on which it is placed should be highly available. Examples of filesystems that could be used for the Software Library are NFS, OCFS2 and ACFS.

For our installation, the Software Library was placed on a highly available NFS filesystem. This NFS filesystem was then mounted on the first OMS using the mountpoint /cc12_SWLib. At this point we also mounted the filesystem on oms2 using the same mountpoint.

The Software Library was configured from the Cloud Control console by navigating to Setup | Provisioning and Patching | Software Library.

We configured our software library location by adding an OMS Shared Filesystem called cc12_SWLib that was on the NFS filesystem.

Step 5: Add Second OMS

In Cloud Control, OMSs are added through the execution of an out-of-the-box Deployment Procedure. Deployment Procedures automate common provisioning and patching operations and are orchestrated and managed from the Cloud Control console.

The following steps outline the procedure that we followed to execute the “Add Management Service” Deployment Procedure.

Step 5.1: Install agent on second OMS server

To enable the execution of the Deployment Procedure on the new OMS server it is necessary to deploy a Cloud Control agent to the server. This was done using the same procedure as was used for deploying the agents onto the database repository servers in Step 3.

Go to Setup | Add Target | Add Targets Manually and use the guided workflow for Add Host Targets to complete the installation of the Enterprise Manager agent.

Step 5.2: Add second OMS using “Add Management Service” Deployment Procedure
The “Add Management Service” Deployment Procedure is provided out of the box. It runs a series of pre-requisite checks on the target Management Service host and performs a clone of the primary Management Service to add a second OMS.

Prior to running the Deployment Procedure, we ensured we were running the latest version of the Deployment Procedure by going to Self Update and checking the Provisioning Bundle Updates (Setup | Extensibility | Self Update). This showed that there were no pending updates to apply and we were therefore running the latest version of the Deployment Procedure.

After ensuring that the latest version is being used, the Deployment Procedure can be accessed by navigating to Enterprise | Provisioning and Patching | Procedure Library and selecting the Add Management Service Deployment Procedure.

We ran the Deployment Procedure by clicking on the Launch button.
The Deployment Procedure provided a guided workflow for adding the new Management Service. The Deployment Procedure asked for confirmation that prerequisites such as configuration of the Software Library and Load Balancer were completed. As all of these tasks were done in prior steps, we checked the boxes to acknowledge we had performed the setup.

The next step of the installer wizard prompts for the destination host and install location for the second OMS. At this step it was also necessary to provide login credentials for both the source server (first OMS host) and the target server (new OMS host). We setup and used some Host Named Credentials for this step.¹

¹ Named Credentials allow users to store and share credentials within Enterprise Manager. The stored credentials could be username/password combinations or public/private key pairs.
On the next page we specified the method to use to transfer the files from the source to the destination server. As we had a shared filesystem configured for NFS access we chose to use the Shared Directory option and specified our NFS location as the Shared Directory Path. As we used a shared directory we didn’t need to specify source and target staging locations on this step.

We were also asked to provide ports for the new OMS. It is recommended to keep the ports on the second OMS the same as those configured on the first OMS as this simplifies SLB configuration.
Finally, we were prompted with some instructions for steps that need to be completed on the SLB after the second OMS is added. These steps need to be followed if the second OMS details are not present in the SLB configuration. As we added the details of this OMS to the SLB Pool when we configured the OMS these steps did not need to be followed.
It is possible to optionally specify an email address where the steps can be sent (note: Email Notification Method must have been configured from the Setup | Notifications | Notification Methods page for this).

When submitted, the procedure completes the process of adding the second OMS by cloning the software homes from the source server to the target server.

The diagram below shows the Cloud Control topology following installation of the second OMS:

![Diagram of Cloud Control topology after installation of second OMS]

**Figure 7: Cloud Control Topology After Installation of Second OMS**

As can be seen, the Load Balancer is now able to direct traffic to either OMS. In the event that we should lose one of the OMSs, the Load Balancer will stop directing traffic to it and application availability will be maintained. We are now protected from the loss of a database node in the Repository tier or a Management Service node in the OMS tier.
Step 6: Add Standby Database

Adding a Standby Database ensures that the repository database is protected from complete failure. Standby Databases provide a copy of the data in a separate environment which can be activated in the event of a primary failure.

Step 6.1: Install agents on Standby nodes

Before the standby database can be configured, the standby database servers must be added to the cloud control environment. This was done using the agent deployment wizard as seen in previous steps.

Goto Setup | Add Target | Add Host Targets and use the guided workflow for Add Host Targets to complete the installation of the Cloud Control agent.

Step 6.2: Add standby cluster targets

After the agent has been deployed to the standby node, the cluster target for the standby nodes should also be added.

This was done by navigating to Setup | Add Targets | Add Targets Manually. Then select “Add Non-Host Targets Using Guided Process” and select “Oracle Cluster and High Availability Service” as the Target Type.

When prompted for a Cluster Target Host, we provided the hostname of one of the standby database hosts. We then selected the hosts that were cluster members and completed the required fields for adding the cluster target.
Following the addition of the standby cluster, the ASM instances and listeners were added. This was done by again navigating to Setup | Add Targets | Add Targets Manually and this time selecting “Add Non-Host Targets Using Guided Processes” for Target Types “Oracle Database, Listener and Automatic Storage Management”

When prompted for a Host, we provided the hostname of one of the standby cluster nodes.
We specified to look for databases on all hosts in the cluster and then configured and added all of the targets that were discovered.

The Database Oracle Homes on the standby nodes will also need to be promoted to Managed Target status. To do this, we navigated to Setup | Add Target | Auto Discovery Results, clicked on the Non-Host Targets tab and promoted the database Oracle Homes that were discovered (note: you can customize the columns that are displayed by clicking on the View option):

**Step 6.3: Create Standby Database using Cloud Control “Add Standby Database” feature**

With the Cloud Control management repository running on Real Application Clusters and the addition of a second management service, the Cloud Control installation is protected from component failure at the primary site. In the event a database or OMS server is unavailable, the application can still continue to function.

Adding Data Guard protects the database from a failure should the database storage fail. It does this by continuously shipping database updates to a standby database that is configured on separate hardware.

The first step in configuring the standby site is to create a single instance standby database for the repository database. This step can be done from within Cloud Control by navigating to the target homepage of the Cloud Control repository database and selecting Availability | Add Standby Database... from the menu.
This started the Add Standby Database Wizard which we used to create a single instance Physical Standby Database.

We selected Online Backup using RMAN

The wizard will prompt for the creation of Standby Redo Logs on the Primary database.
We provided an Instance Name of ‘emdr’. When specifying Standby Database Location, we clicked on the magnifying glass and selected the Database Oracle_Home of the first Standby Database host.

For File Locations we left the defaults provided by the wizard.
We specified Database Unique Name and Target Name as ‘emdr’. We also ensured that we checked the box to monitor the standby database using SYSDBA credentials. This is because SYSDBA credentials are required for complete monitoring of a mounted standby database.

We then clicked Finish which submitted the job to create the standby database.
After the Standby Database has been successfully created, we navigated to the DataGuard homepage and verified that everything was working. This can be done by navigating to the repository database homepage and selecting Availability | Data Guard Administration. This should indicate that everything is in a normal state as per the screenshot below.

Additionally, a Data Guard verification can be run from the Data Guard administration homepage.
Step 6.4: Switch standby database to ‘Maximum Availability’ mode

Maximum Performance, Maximum Availability and Maximum Protection modes offer varying levels of protection with trade-offs in performance, availability and cost. The Add Standby Database wizard creates the Physical Standby database in ‘Maximum Performance’ mode, resulting in asynchronous writes to the standby site. As our standby database is on the same site as the primary database, we will chose to reconfigure our database in Maximum Availability mode, which will perform synchronous writes to the standby database, thus ensuring a higher level of data protection should the primary database fail.

We switched to Maximum Availability mode by clicking on the Protection Mode from the Data Guard Administration Page.

At the next screen, we selected Maximum Availability and then Continue.

We selected our ‘emdr’ standby database and continued.
Following the change in protection mode, we checked the status of the Data Guard configuration from the Data Guard Administration page again. This showed that we were now running in Maximum Availability mode.

The following diagram shows the Cloud Control topology following the creation of the Standby Database:

As shown above, we have now protected the installation against complete failure of the Primary RAC cluster with the use of a standby database. This standby database, however, is only running on a single node, and therefore has less capacity available than the primary.

**Step 7: Convert Standby Database to RAC**

To ensure that Cloud Control performance is maintained when the standby database becomes active, we converted this database to RAC so that it mirrored the primary site. To convert the single instance standby database to RAC, we used the Convert Cluster Database feature of Cloud Control.
be accessed by navigating to the homepage of the Standby Database and selecting Availability | Convert to Cluster Database…

We specified the Oracle Home for the cluster database instances, along with credentials for the Cluster and ASM.

As the standby database being converted was not in read-only mode we were informed that the wizard would do this as part of the convert to RAC process.

We then specified a prefix for the new cluster database and selected to additionally configure the database on the second node in our cluster.
We chose to use the existing Database Area and Fast Recovery Area.

After reviewing the details, we submitted the job.
After the job succeeded we checked the Data Guard status from the Data Guard Administration page. This showed that the Standby Database had been converted to RAC.

The diagram below now shows the topology as it was shown in the introduction. We have configured an SLB, installed an additional OMS, created a standby database and subsequently converted it to RAC to arrive at an MAA Level 3 configuration.

As illustrated by the diagram, we are now protected from failure in each tier.

If the primary database is not available, the standby database can be activated and the connect string used by the OMS tier can be modified to connect to the activated standby database.

Switchover or failover of the primary database to the standby can be done using the Data Guard broker through the ‘dgmgrl’ interface. The repository connect string used by the OMS tier can be modified by executing the ‘emctl config oms -store_repos_details’ command from each OMS.

In our example, OMS connect string is reconfigured by executing the following on each OMS server:

```bash
emctl config oms -store_repos_details -repos_conndesc
"'(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=emrep-clscan.example.com)(PORT=1521))) (CONNECT_DATA=(SERVICE_NAME=emdr)))’ -repos_user sysman
```
Following this reconfiguration each OMS can be restarted.

After the restart, the ‘Management Services and Repository’ target will not be monitored. To resume monitoring of this target it should be relocated to a management agent on the standby site. This is achieved with the ‘emctl config emrep’ command. In our example we reconfigured the ‘Management Services and Repository’ target as follows:

```
emctl config emrep -agent emreps1.example.com:3872 -conn_desc
"(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=emreps-cl-scan.example.com)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=emd)))"
```

In some circumstances, it may be desirable to automate the above reconfiguration steps. This can be achieved through the use of a database trigger setup to fire upon detection of a role change on the Repository database.

For full details of switchover or failover to the standby site (including how to automate the failover process) refer to the Cloud Control Administrator’s Guide.
Conclusion

The breadth and depth of features provided by Cloud Control makes it a critical data center application and as such, high availability of the Cloud Control infrastructure is often deemed essential.

The building of a highly available Cloud Control implementation involves the configuration of a number of different technologies and each provides a particular part of the overall solution.

The implementation of a highly available Cloud Control deployment can be greatly simplified and streamlined by planning the optimal order for the deployment and configuration of the various components. Similarly, the time taken and number of errors can be reduced through the use of the automation features that are provided by Cloud Control.