Oracle® Communications Convergent Charging Controller

High Availability Operations Guide for Linux

Release 15.0.0

October 2023

Copyright

Contents

About This Document

Scope

This guide provides an overview of Oracle Communications Convergent Charging Controller. It also introduces the general concepts of Convergent Charging Controller using Oracle Clusterware. This document is not intended as a detailed configuration guide and is not certified on any specific version of Oracle Cluster Server.

Audience

This guide is intended for system administrators and system integrators who have some experience with implementing high-availability services and have an understanding of Convergent Charging Controller.

Related Documents

For more information, see the following document sets:

- Oracle Communications Convergent Charging Controller:
  - *Oracle Communications Convergent Charging Controller Release Notes*
  - *Oracle Communications Convergent Charging Controller Installation Guide*
- Oracle Database:
  - *Oracle Database High Availability Overview 12c Release 1*
  - *Oracle Database High Availability Best Practices 12c Release 1*
  - *Oracle Data Guard Concepts and Administration 12c Release 2 (12.2)*
  - *Oracle Real Application Clusters Administration and Deployment Guide*
- Oracle Linux:
  - *Oracle Linux Administrator's Guide for Release 7*

Document Conventions

Typographical Conventions

The following terms and typographical conventions are used in the Oracle Communications Convergent Charging Controller documentation.

Formatting Convention	Type of Information
Special Bold	Items you must select, such as names of tabs. Names of database tables and fields.
Italics	Name of a document, chapter, topic or other publication. Emphasis within text.
Button	The name of a button to click or a key to press. **Example:** To close the window, either click **Close**, or press **Esc**.
Key+Key	Key combinations for which the user must press and hold down one key and then press another. Example: **Ctrl+P** or **Alt+F4**.
`Monospace`	Examples of code or standard output.
`Monospace Bold`	Text that you must enter.
variable	Used to indicate variables or text that should be replaced with an actual value.
menu option > menu option >	Used to indicate the cascading menu option to be selected. Example: **Operator Functions > Report Functions**
hypertext link	Used to indicate a hypertext link.

Specialized terms and acronyms are defined in the glossary at the end of this guide.

Chapter 1

System Overview

Overview

Introduction

This chapter provides a high-level overview of Convergent Charging Controller high availability (HA). It explains the basic functionality of the system and lists the main components.

In this chapter

This chapter contains the following topics.

High Availability Overview

Introduction

An HA environment should have minimal or no downtime caused by unplanned outages. Outages can be caused by disk drive failures, network failures, system processing unit (SPU) failures, improper system configuration, and application software failures due to application errors or temporarily unavailable system resources.

Additionally, an HA environment should minimize downtime required for planned system and application maintenance and upgrades. Routine system and application upgrades (such as installing kernel or application patch, or new applications) should occur without taking the critical application services off line.

Oracle highly recommends that client applications be configured so that they detect connection problems and automatically attempt to reconnect when a connection is lost.

Key HA features

Convergent Charging Controller can remain available in various failure conditions. Convergent Charging Controller in an HA environment has the following key features:

- Distributed multiprocess, multi-node, multi-system, and multi-site deployment with application resiliency and fault tolerance
- Application service HA with automatic process recycling and failover
- Hardware HA through redundancy and configuration

Disaster recovery

Disaster recovery requires that you set up a remote instance of Convergent Charging Controller that can be activated in the event of a catastrophic failure at the production site. An HA system for Convergent Charging Controller, consisting of multiple clustered servers, is usually limited by the length of the cables connecting the shared data disk devices and the network interfaces. A remote disaster recovery site that is geographically dispersed requires access to the same resources as the production site, including:

- Network connectivity to clients
- Hardware
- Up-to-date Convergent Charging Controller configuration data
- Dynamic provisioning data

An HA environment requires regular system backups and data replication mechanisms. Data backup must be implemented independent of Convergent Charging Controller.

Hardware requirements for HA

You achieve hardware availability by using redundant backup components for each subsystem that may fail:

- Mirrored dual-port data disks to protect the application from loss of critical data
- Redundant network interfaces and networks to ensure that application clients can connect to the network
- Redundant SPUs to guard against entire system failures

Redundancy by Node Setup

Hardware redundancy on its own does not guarantee the HA of application services offered by Convergent Charging Controller. It is achieved by ensuring that all software components included in the entire solution are built and configured for fault tolerance. When you set up an HA environment, you must eliminate single points of failure that prevent Convergent Charging Controller from processing orders for an extended period of time.

Each Convergent Charging Controller node type has a different redundant architecture to ensure continuous service availability beyond the availability of each hardware element.

Redundancy by SMS Node

The default installation of SMS nodes does not provide HA. The redundant SMS node setup includes the following deployment types:

- **Small deployments:** In small deployments, SMS is deployed as a single node, where SMS is not redundant and its service availability is based on the availability of the underlying server. In such a setup, operational integrity is maintained through the use of a secure backup mechanism. When the SMS node is offline, the network routing offered by the SLC and VWS nodes i.e. the end subscriber handset based services, will be unaffected.
- **Highly available deployment:** : More typically, the SMS is deployed in HA setup, which has two separate servers, each able to host the SMS node arranged in an active/active topology. To achieve this, the SMS is installed on each node with a common disk array for shared application file data, and access to the HA servers is via network switch that accesses one of the SMS servers.The disk array hosts a common application file system. A common SMS database instance (RAC One Node) is installed remotely from the SMS servers. RAC One Node provides HA resilience on the remote database system. With this configuration, a planned or unplanned shutdown of one active SMS node results in clients being directed by the switch to the surviving active SMS node. The failover time is mostly taken up by the time taken to restart the SMS screens and related daemons on the

surviving node. Other processes already running on the active node such as ccsCDRLoader do not need to be restarted. The remote SMS database is not affected and continues to provide service to the surviving SMS node.

These options are based around a single site. To overcome this constraint, introduce Oracle Data Guard:

- Provide an additional SMS disaster recovery option at a second site. The SMS disaster recovery option receives database transaction updates directly from the primary node through Oracle Data Guard, which maintains the SMS disaster recovery database up to date with the primary SMS in near real time.

Note: Activation of the disaster recovery site is typically undertaken following a total and catastrophic loss of the primary site. Activation would typically take 30 to 60 minutes, depending upon the number of connected nodes and the familiarity of the operations staff with the necessary procedures

Redundancy by SLC Nodes

The SLC hosts the service logic and network interfaces and integrates with an external online charging system (OCS) for rating and charging services. The SLCs rely on the connected network elements to manage the distribution of traffic between nodes. These might be on a load-share or active/standby basis to one or several nodes.

Some service providers dedicate particular groups of SLC nodes to specific traffic types or to some other grouping. Other service providers configure all SLCs to handle all types of traffic. If subscribers are provisioned into Convergent Charging Controller, all SLCs will host all provisioned subscribers.

Transactions started on one SLC node continue to be serviced by that node, that is the transaction data remains local to each SLC node and is not shared between the SLC nodes. In the event of either a planned or unplanned outage of an SLC node, all active transactions on that node are lost. New transactions would then be targeted to one of the other available SLC nodes.

Planned outages (for example, maintenance activities) are typically scheduled during quiet traffic periods. In this situation, it is normal for the network operator to reduce traffic for the selected SLC node so that all the new transactions target other SLC nodes. By allowing the existing transactions to complete on the chosen SLC node, maintenance activity can be undertaken with minimum service interruption.

Voice calls and data sessions have periodic commits, which further minimize the opportunity for revenue loss from a planned or unplanned outage of an SLC node. This is achieved since the revenue loss is limited to the amounts reserved but not committed which, through configuration, will be only part of a session and limited to the most recent reservation chunk within those sessions that remain active.

Geographic redundancy of the SLC nodes is achieved by locating SLC nodes on different sites. The total number of required SLC nodes depends on the number of nodes for the required traffic level, the number of sites, and whether complete site failure in the busy hour is a required scenario, or a maintenance outage of a single site.

The worst case would typically be a dual site setup, where long-term catastrophic failure of one site is a required scenario. In this case, each site would need N+1 nodes, requiring a total of 2(N+1) SLC nodes.

Redundancy by VWS Node

The VWS is exclusively deployed in a 2N mated pair architecture, where one node is active and the second node is a hot standby. Each node has its own separate database, with transaction data copied from the active to the standby at the application layer, such that the active service transactions can be started on one node, and in case a failover occurs, the service transactions can complete on the second node.

The client systems of the VWS are the SMS and SLC nodes. A mated pair of VWS nodes forms a logical construct termed a 'Domain'. Each VWS Domain hosts one or more voucher batches. Through data mastered on the SMS and replicated to the other nodes, the target VWS Domain can be identified. Each node then maintains connections to both VWS nodes within each domain and exclusively uses the connection to the active node within the required domain. The two nodes within a domain are designated the primary and secondary nodes. If the primary node is available, it will be the active node. When the primary node fails, the secondary node becomes the active node. If the primary node becomes unavailable, due to either planned or unplanned outages, the transactions initiated on one node are continued on the previously standby node. As such, failover between the VWS nodes within a domain is seamless and happens in real time.

When the primary VWS node returns to service, it initially needs to catch up with the active secondary node. It first processes the incoming synchronization files before notifying the client systems that it is now active. After it becomes active, it continues to process any in-bound synchronization files. Geographic redundancy of the VWS nodes is achieved by locating the two nodes in any given domain, on separate sites.

Redundancy by Traffic/Service Type

To assess the impact of the loss of any single node, to look at the impact with respect to each traffic or service type, such as:

- Customer care operations that come to the platform through these interfaces:
  - SMS screens
  - PI on the SMS
  - OSD on the SLC nodes
- Customer care operations can come to the platform through these interfaces:
  - WEB 2.0 through the PI to the SMS
  - WEB 2.0 through OSD to the SLC(s)
  - USSD request to the SLC(s)
  - SMS text to the SLC(s)
  - IVR session managed through the SLC(s)
- Session-based traffic services that are categorized by having a back and forth message exchange between the serving network element and the Convergent Charging Controller that is the controlling network element. For each node:
  - Failure of an SMS node will have negligible impact to an active session or a new. If there is no subscriber data, no network-side updates occur.
  - Failure of the primary VWS node has no affect on active or new sessions, that is any transactions initiated on the primary, but not completed, will complete on the secondary.
  - Where the SLC node serving the session is lost, the connection to the adjacent network element will drop. For established bearer sessions, the controlling network element may hold the session up until either it is terminated by one of the parties involved and/or until the controlling element requires further direction from Convergent Charging Controller. This would typically be an additional reservation of funds to continue the session. At this point, for voice services, the session failure would be recognized and the bearer session dropped. For data services, the serving node may maintain the bearer session and attempt a new transaction to one of the other SLCs. The sessions that were being set up at the time of failure may either fail back through the network, or be re-attempted to one of the other SLC nodes, as determined by the serving network element. The result on the end subscriber is that the session attempt might fail or an established session might be dropped.
- Event-based traffic services are categorized by having a request-response transaction, that is the response concludes the message exchange between the serving network element and the Convergent Charging Controller. Failure of the serving SLC node will result in loss of connection to the serving network element and failure of that request, following a short timeout. In that situation,

serving network elements will typically re-attempt the transaction to one of the other serving SLC nodes, that is the end subscriber does not perceive any issue.

Chapter 2

SMS Node Configuration for High Availability

Overview

Introduction

This chapter provides information about configuring for high availability on an SMS node.

In this chapter

This chapter contains the following topics.

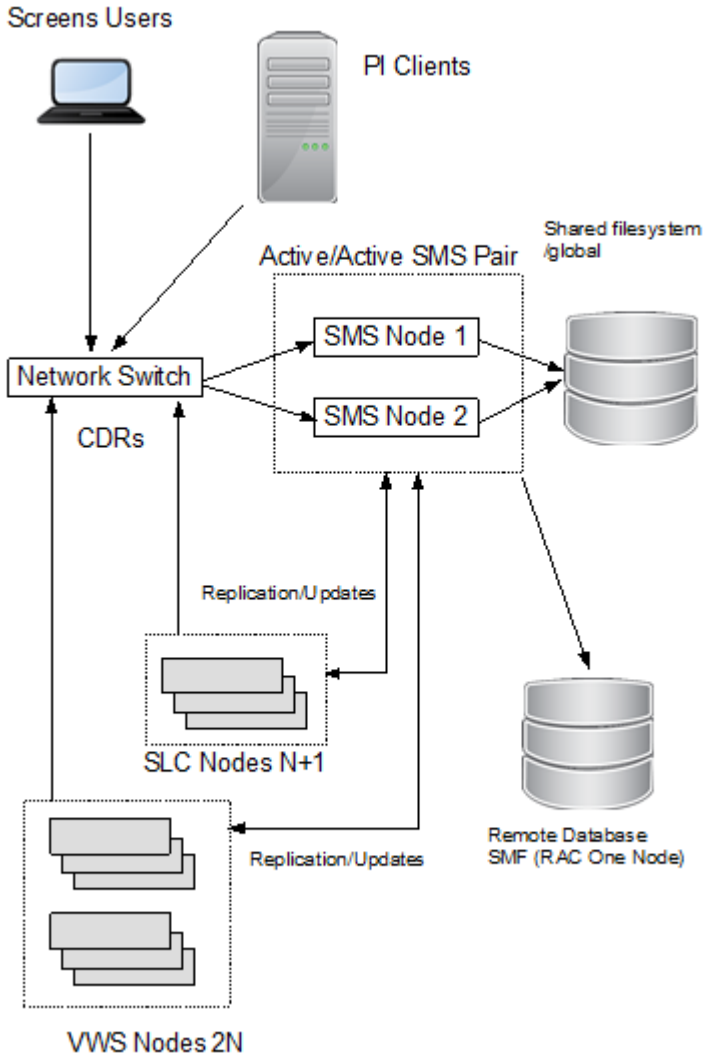
Configuring High Availability on SMS

Introduction

This section describes the HA SMS configuration. HA SMS is provided by two SMS nodes that access the same SMF database instance remotely. The remote database uses Oracle RAC One Node to provide a single resilient SMF database instance to the SMS nodes. This design splits the NCC application layer from the database layer.

A network switch proxies client connections to the HA SMS nodes. See the architecture diagram.

Architecture



Active/Active SMS Node Configuration

On each SMS node in the active/active SMS configuration:

- Install Convergent Charging Controller specifying the same remote database instance (SMF) and same remote database host.
- When defining nodes in the OCNCC screens, define one SMS node as node id 1 and the other SMS node as node id 2.
- Ensure that Unix user IDs and group IDs are aligned across the SMS nodes (for example, user smf_oper should have the same numeric uid on each SMS node, and the group esg should have the same gid on each SMS node) as a pre-install task for Convergent Charging Controller.
- Install the SMF database instance remotely using RAC One Node. Each SMS node connects to the same SMF database instance.

About Configuring Active/Active SMS Nodes

Refer to the following guides to find more general information about active/active setup:

- *Oracle Database High Availability Overview 12c Release 2 (12.2)*
- *Oracle Database High Availability Best Practices 12c Release 1 (12.1)*
- *Oracle Data Guard Concepts and Administration 12c Release 2 (12.2)*
- *Convergent Charging Controller Installation Guide*

Synchronizing System Time

The local system time must be consistent on all SMS nodes. You can properly synchronize the time by enabling the Network Time Protocol (NTP) daemon on each node in the HA SMS. The NTP daemon configuration is in the **/etc/ntp.conf** file.

It is recommended that each SMS node be configured to refer to the other SMS node as its peer.

Configure redundant time services so the loss of any one time service does not affect time synchronization.

For more information, see the discussion about how to configure NTP in *Oracle Linux Administrator's Guide for Release 7*.

Shared Storage Configuration

Both SMS nodes should have access to a common filesystem for shared file data such as CDRs and process lockfiles. The shared volume should itself be resilient by utilizing technologies such as dual-ported disks and RAID.

As an example, each SMS node mounts the shared volume as /global/CDR and this location provides directories

Directory	Description
/global/CDR	Common mountpoint
/global/CDR/CCS	CCS subdirectory
/global/CDR/CCS/logs	Logs subdirectory
/global/CDR/CCS/logs/CDR-in	Receive directory for VWS CDR files
/global/CDR/CCS/logs/CDR-store	CDR files after being processed by ccsCDRLoader
/global/CDR/CCS/logs/ccsNotificationRead	NotificationPlugin CDRs (VWS)
/global/CDR/CCS/logs/expiryMessage	Balance expiry messages (VWS)
/global/CDR/CCS/logs/wallet	Wallet expiry messages (VWS)
/global/CDR/UIS	UIS CDRs (SLC)
/global/CDR/cdr/received	ACS/PIN/LCR CDRs (SLC)

Switch Configuration

Introduction

You perform the following tasks to configure the hardware network switch that interfaces between external clients and the HA SMS.

- Configure the hostname and IP of the network switch, e.g haswitch
- Configure the networks switch so that incoming connections to the OCNCC ports are forwarded to one of the SMS nodes

Configure Network Switch

Use the network switch name (for example, haswitch) when connecting to the HA SMS, and configure NCC components to use the network switch name to connect to the SMS.

See the following section for more port forwarding information.

Step	Action
1	On each SMS in the /IN/html/sms.jnlp file, configure the jnlp.sms.host property to refer to the haswitch name: <property name=”jnlp.sms.host” value=”haswitch”/>
2	On each OCNCC node add an entry to the /etc/hosts file for haswitch
3	On the replication nodes, in eserv.config configure the cmnPushFiles items to specify the haswitch when sending CDR files to the SMS (-h cmnPushFiles option) and configure the destination directory on the SMS to be a location on the common shared filesystem (-r cmnPushFiles option).

Configure Network Switch Ports

This procedure shows a configuration list for the NCC network ports on the network switch for use by SMS clients. SMS clients do not connect to SMS directly. Instead, they connect to the network switch which then forwards the client connection onto one of the SMS nodes. For each HA SMS port that the switch is proxying for, configure the switch to periodically check which of the HA SMS back end ports supporting a service are available. If a back-end port ceases service, the switch can then direct new port connections to the back-end port on the other SMS node. Further, the switch can forward certain connections to one SMS node preferentially (provided it is available) or the switch can round-robin the clients connects over the available SMS nodes.

The following port configuration example uses the default NCC port numbers.

Step	Description
1	Configure the SMS screens jnlp download port 80 (HTTP)
2	Configure the SMS Task Agent port 6332
3	Configure the SMS Naming Service port 5556
4	Configure the CCS Be Orb port 6335
5	Configure the SMS TrigDaemon port 6334
6	Configure the SMS Reports Service port 6333
7	Configure the UIS Receive Files port (SLC UIS CDRs) 2031
8	Configure the VWS Push Files port 2027

9	Configure the PI service port 2999. Configure additional ports if they are defined in the PI screens. Configure the HA switch as a PI Host in the PI screens.
10	Configure ACS Receive Files port (ACS CDRs) 2028

HA SMS Active/Active Process Configuration

Introduction

This section provides information about configuring OCNCC specific processes in an Active/Active deployment.

SmsMergeDaemon

This process is only required for non-HA SMS deployments and should not be run on a HA SMS.

To stop and disable smsMergeDaemon, on each SMS node:

1. Log in as the root user.

2. Stop and disable the smsMergeDaemon service

```
systemctl stop smsMergeDaemon.service
systemctl disable smsMergeDaemon.service
```

3. Open the /IN/bin/OUI_systemctl.sh file in a text editor.

4. Comment out the entry for the smsMergeDaemon.service

5. Save and close the file.

Legacy SMS-based ccsPeriodicCharge

This section applies to the legacy SMS based ccsPeriodicCharge process, not the VWS based Periodic Charge that uses VWS wallets.

If the deployment does not require legacy ccsPeriodicCharge on the SMS, disable this service on each SMS node by commenting out the ccs_oper crontab line.

If legacy ccsPeriodicCharge is required, configure it on each node to use a common lock file that resides on shared storage. For example if shared storage is mounted on /global/CDR on each SMS node, and directory CCS/logs is present under this location, the lock file can be configured in eserv.config on each SMS node by doing the following:

1. Log in as the root user.

2. Edit /IN/service_packages/eserv.config and define the LockFile parameter in the CCS.ccsPeriodicCharge section to be placed on the shared disk volume

```
CCS = {
…
  ccsPeriodicCharge = {
     LockFile = "/global/CDR/CCS/logs/.ccsPeriodicCharge"
     …
  }
  …
}
```

3. Save and close the file

SLC CDR Configuration

On each SLC, configure cmnPushFiles startup scripts to specify *haswitch* as the destination host (-h option) and specify a destination directory in the shared filesystem on the SMS nodes (-r option).

Following is an example:

Step	Action
1	Configure SMS/bin/cmnPushFilesSMSStartup.sh `exec /IN/service_packages/SMS/bin/cmnPushFiles -d /IN/service_packages/SMS/cdr/closed -f /IN/service_packages/SMS/cdr/retry -r /global/CDR/cdr/received -h haswitch -s 10 -p 2028 -S cdr`
2	Configure ACS/bin/cmnPushFilesACSStartup.sh `exec /IN/service_packages/ACS/bin/cmnPushFiles -d /IN/service_packages/SMS/cdr/closed -f /IN/service_packages/SMS/cdr/retry -r /global/CDR/cdr/received -h haswitch -s 10 -p 2028 -P PIN -S txt`
3	Configure UIS/bin/uisCdrPushStartup.sh `exec /IN/service_packages/UIS/bin/cmnPushFiles -d /IN/cdr/ussd -r /global/CDR/UIS/cdr/received -S .cdr -h haswitch -p 2031`
4	Configure NP_SERVICE_PACK/bin/cmnPushFilesNPStartup.sh `exec /IN/service_packages/NP_SERVICE_PACK/bin/cmnPushFiles -d /IN/service_packages/SMS/cdr/closed -f /IN/service_packages/SMS/cdr/retry -r /global/CDR/cdr/received -h haswitch -s 10 -p 2028 -P LCR -S .cdr`

VWS CDR Configuration

On each VWS, configure the cmnPushFiles/CDR sections in the eserv.config file to specify the *haswitch* as the destination host (-h option) and specify a destination directory in the shared filesystem on the SMS nodes (-r option).

Following is an example:

Step	Action

1	Configure item cmnPushFiles in section CCS.ccsVWARSExpiry
	`cmnPushFiles = [`
	`  "-d", "/IN/service_packages/CCS/logs/wallet"`
	`  "-r", "/global/CDR/CCS/logs/wallet"`
	`  "-h", "haswitch"`
	`  "-F"`
	`]`

Step	Action

2	Configure item cmnPushFiles in section CCS.notificationPlugin
	`cmnPushFiles = [`
	`"-d", "/IN/service_packages/CCS/logs/ccsNotificationWrite/"`
	`"-r", "/global/CDR/CCS/logs/ccsNotificationRead/"`
	`"-h", "haswitch"`
	`"-F"`
	`]`

3	Configure item cmnPushFiles in section CCS.ExpiryMessages
	`cmnPushFiles = [`
	`"-d", "/IN/service_packages/CCS/logs/expiryMessage/"`
	`"-r", "/global/CDR/CCS/logs/expiryMessage/"`
	`"-h", "haswitch"`
	`"-p", "2027"`
	`"-F"`
	`]`

4	Configure item CDR in section BE.cmnPushFiles
	`CDR = [`
	`"-d", "IN/service_packages/E2BE/logs/CDR-out"`
	`"-r", "/global/CDR/CCS/logs/CDR-in"`
	`"-h", "haswitch"`
	`"-F"`
	`]`

CcsCDRLoader Configuration

In an active/active configration, instances of ccsCDRLoader can be run on each SMS node. CcsCDRLoaders co-operate using file locking to ensure that only one loader processes an incoming CDR file. It is also possible to dedicate ccsCDRLoader instances to certains VWARS numbers with the --vwars_range option. The --serverID option can be used to restrict a ccsCDRLoader to only handle CDR files produced by a specific VWARS served Id.

SMS Node Failure

In the Active/Active deployment, with two SMS nodes, if one node fails, then the switch will direct clients to the surviving node.

The following manual operations are also required:

1. Restart these services on the surviving node:

   smsTaskAgent, smsReportsAgent, ccsBeOrb

   Connected screens clients should clear their browser cache

2. Login to the screens.

SMS Geo Redudancy

Geographic redundancy is achieved by duplicating the SMS HA configuration at a second site.

Use Oracle Data Guard over a high speed link (for example, 100 GbE) to synchronize the remote SMS primary site remote database with the secondary site remote database.

Oracle Data Guard should be configured with the Max Performance option.

Data Corruption Protection

Oracle Data Guard can also be configured with a 6-hour apply lag at the second site database, this provides protection against data corruption.