High Availability Distributed System Configuration Guide for Network Compliance Manager 1.4

CiscoWorks

Table of Contents

Chapter 4: Failover and Recovery 65

Chapter 5: Troubleshooting 69

Appendix A: Restoring NCM Replication After Upgrading to Oracle 10g 73

Preface

This document contains information on installing, configuring, and administering the High Availability Distributed System on Oracle for Network Compliance Manager 1.4 on Oracle 10g (10.2.0.4).

Document Conventions

The following table explains the conventions used in this guide.

Convention	Description/Action
Italic	Used for system messages, paths, file names, and Web URLs. For example, *C:\ncm\client\sdk\docs*
[Link](#)	Moves you from one location to another within the document, opens Web pages, or opens a new email message. In the guide, cross-references are contained within quotation marks and include a page number, while links to URLs and email addresses appear as underlined text.
< >	Indicates variable information, such as a name or folder that you must supply. Do not include the angle brackets when replacing the placeholder.

NCM Documentation

The core NCM 1.4 documentation includes:

- *User Guide for Network Compliance Manager 1.4* — To view the PDF version, after logging in to NCM, on the menu bar click Docs. The CiscoWorks Network Compliance Manager Documentation page opens.

- Online HTML Help Files — To view the Online HTML Help files, after logging in to NCM, click the Help icon at the top of any NCM page.

Obtaining Documentation, Obtaining Support, and Security Guidelines

For information on obtaining documentation, obtaining support, providing documentation feedback, security guidelines, and also recommended aliases and general Cisco documents, see the monthly What's New in Cisco Product Documentation, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Chapter 1: Getting Started

Use the following table to quickly locate information in this chapter.

Terminology

The following terms are used throughout this guide:

- **NCM Core** — A single NCM Management Engine, associated services (Syslog and TFTP), and a single database. A NCM Core can manage multiple Partitions.

- **Partition —** A set of devices with unique IP addresses. A Partition is managed by one (and only one) NCM Core. Multiple Partitions can be managed by a single NCM Core. Refer to the *User Guide for Network Compliance Manager 1.4* for information on segmenting devices.

- **NCM Mesh** — Multiple NCM Cores connected via high availability replication.

- **High availability replication** — Replication where all data is kept consistent across all database servers.

- **Master Definition Site** — Oracle replication requires that one database in a high availability replication NCM Mesh hold the definition of the replication setup. This database is called the Master Definition Site.

Overview

The NCM Distributed System on Oracle is a high availability system where the data from each NCM Core in a NCM Mesh is accessible to all other NCM Cores. This provides a comprehensive view of your data and allows for redundant data and failover in the event of a problem with a single NCM Core. The high availability features include both database data and certain file system data, such as software images and device driver packages. Keep in mind that software images and device driver packages are also replicated across the NCM Mesh.

The following comprises the NCM Distributed System:

- A NCM Core and a Site:

  - A device is associated with a single Site.

  - A Site is associated with a single NCM Core.

  - A task is associated with a specific NCM Core.

- High availability replication on Oracle:

  - Conflict resolution in the event that the same piece of data is modified almost simultaneously on two separate NCM Cores. This conflict is typically resolved using the latest timestamp method.

  - Replication monitoring and conflict notification is built into NCM. You can manage Oracle replication conflicts and view replication job statuses from within the NCM UI.

  - The NCM scheduler is multi-core aware. You can schedule group tasks containing devices that are associated with different NCM Cores. The system will run these tasks on the correct NCM Core. You do not have to schedule tasks on the appropriate NCM Core.

System setup requires a thorough understanding of Oracle and NCM. Installation includes number of steps that must be performed on the various servers. In addition, certain network changes may be necessary to allow connections between the servers. Once setup is complete, you will need to partition your devices into Sites to ensure proper NCM Core access to devices. Refer to the *User Guide for Network Compliance Manager 1.4* for information on segmenting devices.

Note: The NCM High Availability Distributed System on Oracle requires 32-bit or 64-bit versions of Oracle Enterprise Edition 9i or 10g (10.2.0.2/4) Database server.

Architecture

The following figure shows a typical two-NCM Core high availability installation. The installation enables failover from one NCM Core to another, while the remaining NCM Core continues to manage all devices.
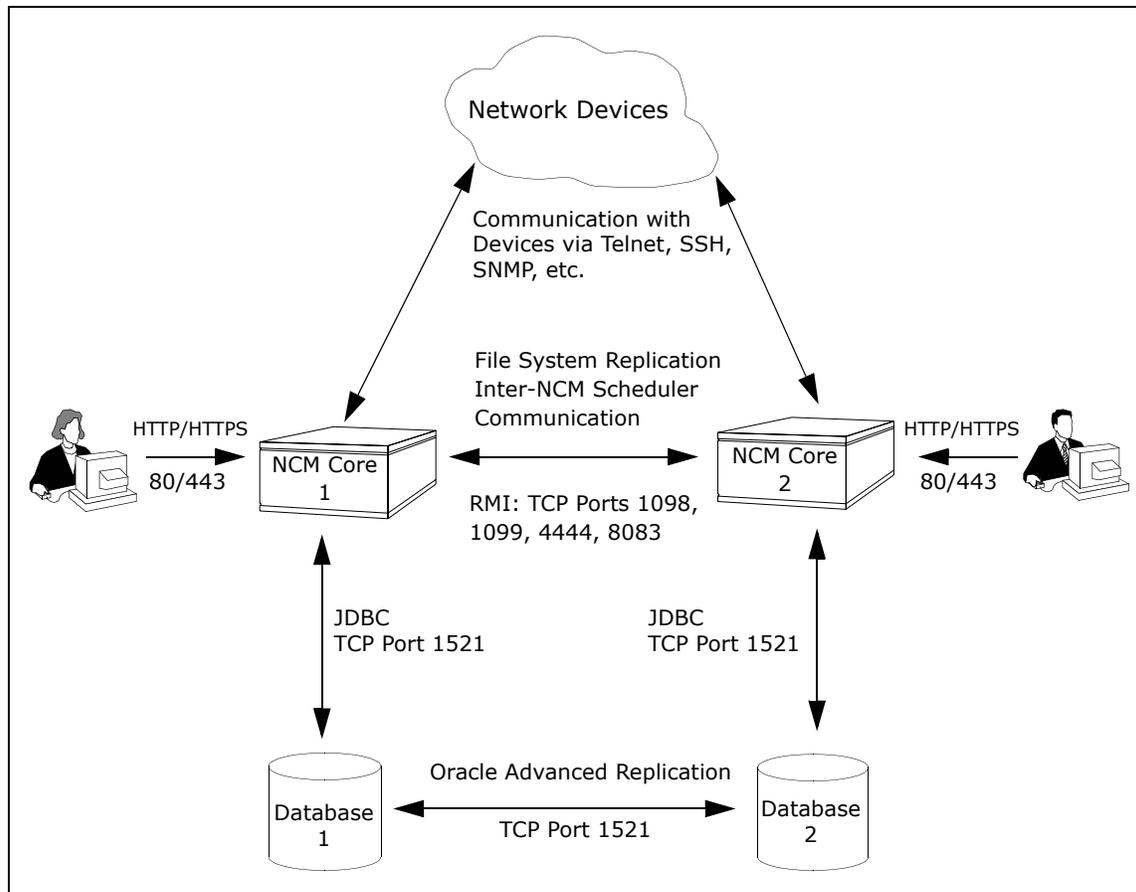
Keep in mind that some installations might have two separate network device collections that are inaccessible from the non-managing NCM Core. These installations still provide data redundancy and scaling benefits, however.

Note that the database data transport uses standard Oracle Advanced Replication. For details on how replication works, refer to "Chapter 3: System Administration" on page 45.

The communication between the NCM Cores is done using Java's Remote Method Invocation (RMI). RMI is used to ensure:

- Certain file system objects, such as software images and driver packages, are in sync.
- NCM tasks are scheduled and run on the correct NCM Core.

NCM does not replicate NCM Core specific options, including:

- TFTP server information
- TACACS server information
- Scripting language settings
- Hop Box definitions
- Driver directory
- Install directory
- Local Gateway information
- SWIM proxy server
- Active Directory settings
- Scheduler (concurrent task limits)

NCM Internal Monitoring: Oracle Advanced Replication

Oracle Advanced Replication can be configured to automatically handle certain types of conflicts. Where possible, NCM uses Oracle's built-in conflict handling software. For example, for most tables with *update/update* conflicts, NCM installs the default Oracle "latest timestamp wins" handler.

When it is not possible to use Oracle's default handler, NCM either adds a stored procedure to handle the conflict (in particular, for *update/delete* conflicts), or adds additional code inside the application with logic that attempts to correct the error. If the error is uncorrected through that logic, NCM alerts the NCM administrator to the issue using NCM' standard event notification emailing functionality. Refer to the *User Guide for Network Compliance Manager 1.4* for information on configuring email notification.

Note: When a second NCM Core is used as a standby for failover and not for normal operation, you should not encounter *update/update* conflicts.

Refer to "Understanding Oracle Replication" on page 46 for detailed information on NCM monitors.

In addition, in Oracle, NCM monitors on a regular schedule the following on each NCM Core for the database transport (i.e., replication):

- Push jobs that handle moving data from one NCM Core to another. If the monitor check fails and the Push job is found to be broken, NCM will attempt to restart it.

- Purge jobs that successfully clear pushed data from the push queue, and attempts to restart the Push queue each time the check fails.

- The deferred transaction queue for any transactions that could not be applied to the local database due to a conflict. NCM applies additional application logic to attempt to correctly apply the transaction. If unsuccessful, NCM alerts the NCM administrator via email notification.

- Unrecoverable uniqueness conflicts. NCM alerts the NCM administrator in the event of unrecoverable uniqueness conflicts. Typically, there is no way to automatically correct these types of issues.

- Both the DEFTRAN and DEFLOB queue size. These queues hold pending data that needs to be pushed to other servers. Typically, if replication is working, but is beginning to backup, these queues will begin to grow. This can occur if the connection between databases is congested. NCM alerts the NCM administrator to this issue via email notification. The system will continue to push data. However, it could take a long time to catch up depending on network conditions.

Note: Refer to the *User Guide for Network Compliance Manager 1.4* for information on configuring email notification.

NCM Internal Monitoring: Inter-NCM Core Communication

In addition to replication monitoring, NCM also monitors the following between each NCM Core:

- RMI connectivity
- NCM server timestamp differences

These monitors generate events during error conditions. The events can be emailed to the NCM administrator or Oracle DBA using a standard event rule in NCM. Refer to the *User Guide for Network Compliance Manager 1.4* for information on configuring email notification. Refer to "NCM Generated Events" on page 49 for detailed information on NCM system events.

Note: There is a standard example event rule shipped with NCM. The event rule can be updated as necessary.

Chapter 2: Installation, Setup, and Upgrading

Use the following table to quickly locate information in this chapter.

System Requirements

The NCM High Availability Distributed System on Oracle software requires Oracle Enterprise Edition, 10g (10.2.0.4). Refer to the NCM 1.4 upgrade and installation documentation for information on database server hardware requirements.

Installation Planning

To properly install the Distributed System software, you must first complete:

- Device partitioning planning across NCM Cores

- Network configuration planning for connectivity between NCM servers and devices. For example, what network connectivity is required to support failover for device access? The ability of a NCM Mesh to failover for device access depends in part on proper network setup to ensure access to devices. In some cases, you might not want to have failover work for complete device access, but instead have it ensure access to data while corrective action is taken to restore the network connectivity to the affected NCM Core.

- Network configuration planning for connectivity and bandwidth between the different servers (NCM and database) that comprise the NCM Mesh and between the NCM Management Engines and devices. The NCM Cores in the NCM Mesh will also need bandwidth between them equal to the bandwidth provided between a single NCM server and its database in a single NCM Core. Refer to for more information.

- Network configuration planning for bandwidth usage by the NCM Mesh. The bandwidth required between a NCM Management Engine and its database depends on the size of the deployment, including the number of devices, average configuration size, and the number of concurrent users. The bandwidth available between the Oracle databases should be as large as between the NCM Management Engine and the database. For a deployment of 14,000 devices with an average 10K configurations that change once a week, the bandwidth requirements could conservatively be on the order of 1Mb/s. Peak bandwidth requirements could be higher, depending on usage.

- Oracle server setup planning. Keep in mind that the database properties required for replication can be set on initial database creation. You do not need to wait until replication setup to set these parameters.

Keep in mind that during setup, data is dumped from an existing NCM server. This could take time and use disk space as the database is copied. In addition, when data is copied from one database to another, there must be ample bandwidth between the two (or more) servers. It will also take time to import the exported database into the other NCM Cores.

You can estimate the time it will take to copy data from server to server given the bandwidth between servers. You can also calculate the disk space requirements for the export (and subsequent import) operations by looking at the size of your database. If you want to export or import data from the same server as the database, the disk space requirement is twice the size of your database.

Note: Estimating time for the import and export operations could be difficult. You should allocate a lengthy time frame to complete this work. In addition, the NCM server(s) must be off during the export and import steps. A sufficiently long maintenance window should be planned for.

You will also have to ensure:

- Time synchronization setup for the NCM servers in the NCM Mesh
- Users are instructed to login to their "closest" NCM Core
- Access to an Oracle DBA to support the NCM Distributed System installation

To assist in planning, please note the following limitations and suggestions concerning the NCM Distributed System:

- NCM currently does not support joining multiple existing NCM installations into a NCM Mesh. You can only create a NCM Mesh from a single existing NCM server, adding new NCM Cores as appropriate. You can also create a NCM Mesh from scratch.

- Users should not share logins. Due to the replication system used to share data across NCM Cores, two users should never use the same login name to connect to two different NCM Cores at the same time. If they do so, the system will likely require additional work by the system administrator to ensure that the effected user's profile is properly synced up across the NCM Mesh.

- NCM currently assumes that all servers in the Mesh (NCM and database) share a single timezone.

- Future NCM upgrades will take longer and require more downtime due to the need to both update the replication setup and to update all servers in the NCM Mesh. NCM does not support rolling upgrades where one part of the NCM Mesh is running a version of NCM while the rest of the NCM Mesh is running a different version.

- The ReplicationScript.sql script updates the RN_CORE table. When the replication script runs, the RN_CORE changes. There is no need for it to UPDATE and INSERT into the RN_CORE table on both Cores because replication is already running. The database changes will be pushed via replication to the second database. The contents of the RN_CORE tables on both databases should match whatever is setup in *ReplicationScript.sql*.

Installation and Removal

This section includes information on:

- Initial setup of a NCM Mesh on two servers

- Addition of a server to an existing NCM Mesh

- Removal of a NCM Core from a NCM Mesh

- Removing replication across the entire NCM Mesh

- Modification of the replication Master Definition Core

Note: Refer to for information on Troubleshooting the installation.

Unpacking the Replication Script Installation Bundle

The setup files for the NCM Distributed System are the standard setup files for a normal NCM installation, with the addition of the *OracleReplicationScriptToolBundle.zip* file. This file should be unzipped onto a standard NCM server. The setup files include a Java program that will be run to generate replication setup SQL scripts. This program will need to use the Java run-time that is available on the NCM server.

Initial Replication Setup

This section includes information for initial replication setup on a Master Definition Site (Master 1) and one additional Master Site (Master 2). Keep in mind that NCM will be off-line during initial replication. It is recommended that you carefully read the following sections so as to streamline the replication process. Keep in mind that a new database instance on Master 2 will need to be manually created when running initial replication setup.

Note: Be sure to save this information. You will need it each time you upgrade in the future.

Planning

When initially setting up replication, complete the following table before starting.

Replication User:_____ (The default is repadmin.)

Password:_____

	Master 1	Master 2
SID		
NCM App Server		
RMI Port		
Database Server		
Database Name		
Database User		
Database Password		
Timezone Offset		

Protocols, Databases, and Ports

NCM communicates with devices using a combination of the following protocols, databases, and ports. If you use a given protocol, NCM requires access to the corresponding port. Specifically, if NCM communicates with devices protected by firewalls, these ports must be open. Refer to "Architecture" on page 11 for more information.

Protocol/Database/Port	From/To
NCM server (running the Mgmt Engine, Syslog, TFTP) and network devices	
Telnet (port 23)	From the NCM server to network devices.
SSH (port 22)	From the NCM server to network devices.
TFTP (port 69/udp)	From network devices to the NCM server.
Syslog (port 514/udp)	From network devices to the NCM server.
SNMP (port 161/udp)	From the NCM server to network devices.
Between the database servers	
Oracle (port 1521)	In a Distributed System configuration, the Oracle processes connect to each other on port 1521.
Between the NCM servers	
JNDI (ports 1098, 1099)	NCM server to NCM server. You can change this by editing the NCM configuration files. Please contact Customer Support for assistance.
RMI (ports 4444 and 8083)	NCM server to NCM server. You can change this by editing the NCM configuration files. Please contact Customer Support for assistance.
Between the NCM server and the database server	
Oracle (port 1521)	From the NCM server to an Oracle database server.
NCM server and NCM users	
HTTPS (port 443)	From the NCM server to NCM users. You can change this by editing the NCM configuration files. Please contact Customer Support for assistance.

Protocol/Database/Port	From/To
Telnet (port 23 - Windows or 8023 - Solaris/Linux)	From the NCM client to the NCM server. This can be changed from the Administrative Settings option.
SSH (port 22 - Windows or 8022 - Solaris/Linux)	From the NCM client to the NCM server. This can be changed from the Administrative Settings option.

Permissions for Non-SYSTEM Users

The following permissions are required for the NCM Distributed System setup process and operation if you are running the NCM servers under a non-SYSTEM user on Oracle.

Note: NCMUSER is the Oracle username that NCM uses to access the database, typically the Oracle schema name. This is required to be the same on all NCM Cores in the NCM Mesh. Be sure to replace NCMUSER with the username your NCM server uses to access the database.

Setup and Upgrade Permissions (Oracle 10g)

For replication setup and upgrade:

```
GRANT EXECUTE ON "SYS"."DBMS_REPCAT_ADMIN" TO "NCMUSER";
GRANT EXECUTE ON "SYS"."DBMS_DEFER_SYS" TO "NCMUSER";
GRANT CREATE PUBLIC DATABASE LINK TO "NCMUSER";
GRANT CREATE DATABASE LINK TO "NCMUSER";
GRANT CREATE USER TO "NCMUSER";
GRANT CREATE ANY PROCEDURE TO "NCMUSER";
```

For normal operation of a NCM Core in a replication environment:

```
GRANT EXECUTE ON "SYS"."DBMS_REPUTIL" TO "NCMUSER";
GRANT SELECT ON "SYS"."DBA_CONSTRAINTS" TO "NCMUSER";
GRANT SELECT ON "SYS"."DBA_CONS_COLUMNS" TO "NCMUSER";
GRANT SELECT ON "SYS"."DBA_REPCOLUMN" TO "NCMUSER";
```

For replication deletion:

```
GRANT DROP USER TO "NCMUSER";
GRANT DROP PUBLIC DATABASE LINK TO "NCMUSER";
GRANT DROP ANY PROCEDURE TO "NCMUSER";
```

For normal NCM operation:

```
GRANT CREATE SEQUENCE TO "NCMUSER";
GRANT CREATE SESSION TO "NCMUSER";
GRANT CREATE TABLE TO "NCMUSER";
GRANT CREATE PROCEDURE TO "NCMUSER";
GRANT SELECT ANY DICTIONARY TO "NCMUSER";
GRANT CONNECT TO "NCMUSER";
```

System Setup

1. Be sure you have the following:

   a) A current, up-to-date 1.4 database on Master 1.

   b) A NCM application server (NCM Core 1) connected to the database (Master 1).

   c) A database with no data or NCM schema setup (Master 2). The database identifier (SID) needs to be different from the database on Master 1.

   d) A network connection from the Master 1 servers to the Master 2 servers (and vice-versa) that enables ports 1521 (Oracle only), 1098, 8083, 1099, and 4444 (or appropriate variations) to be connected between these servers. Port 1521 must be open between the Oracle servers. The remaining ports must be open between the NCM servers.

   e) The *ReplicationScriptTool bundle* files installed on a Java-capable system (such as NCM Core 1).

2. Obtain the following information (refer to "Planning" on page 19):

   a) Login/password/database identifier for NCM into the two databases (on Master 1 and Master 2).

   b) Decide on a replication password. The user "repadmin" will be created and assigned this password by default. You can use a different replication username if needed. This user will be used to manage replication in your system.

   c) NCM application server hostname, NCM application server RMI listening port, database server hostname, and database listening port for Master 1 and Master 2.

   d) The timezone offset (integer from UTC) for the entire NCM Mesh. This must be a constant across the NCM Mesh. Do not consider daylight savings time when setting this value.

3. Set (or verify) the following database parameters on each Master database:

   a) global_names = TRUE

   b) open_links = at least 4 (increase this by 1 for each additional server in the NCM Mesh beyond 2)

   c) open_links_per_instance = at least 4 (same as open_links)

   d) parallel_min_servers = 2

   e) parallel_automatic_tuning = TRUE (for Oracle 9i databases only)

   f) shared_pool_size = increase current value by 40m to support replication

   g) job_queue_processes = at least 2

   Note: The parameters must be set on each database in the NCM Mesh. The parameters could be setup when the database is first created. If not, they must be modified at this point. You can view the current value of these parameters using Oracle Enterprise Manager or SQLPlus with the "SHOW PARAMETERS" or "SELECT NAME, VALUE FROM v$parameters" commands (where NAME = <parameter_name>). For example, <parameter_name> could be global_names from the list above.

4. Ensure that all database server and NCM server systems are set to use the same time and timezone.

5. Unpack the *OracleReplicationScriptToolBundle.zip* file on NCM Core 1 (or any NCM installation).

6. Update the following variables for Master 1 and Master 2 in the *ReplicationScriptTool.properties* file located in the directory where *OracleReplicationScriptToolBundle.zip* was unpacked.

    –   Replication user

    –   Password

    –   Replication group name

    –   Timezone offset

    Ensure that you modify all "REPLACEME" text entries in the *ReplicationScriptTool.properties* file with the appropriate values for your environment. For the app.server values, specify the IP address of the appropriate NCM application server. In addition, make sure the mode property is set to "initial". These properties are described in detail in the *ReplicationScriptTool.properties* file.

7. If you want to run NCM servers under a non-SYSTEM user on Oracle, execute the scripts listed in the .

8. Run the ReplicationScriptTool via the following command:
    ```
    java -classpath . ReplicationScriptTool
    ```

9. Create the database on Master 2. This database should have a different database identifier (SID) from the database on Master 1. It should also have the same schema name and database user login name as the database on Master 1.

10. Add Master 2's database information to the local *TNSNAMES.ora* on Master 1. Verify that Master 1's database information exists in the local *TNSNAMES.ora* on Master 1. If Master 1's database information does not exist, add it same way as other entries.

    Note: You may need to make modifications to the *LISTENER.ora* and *SQLNET.ora* files depending on how your Oracle servers are configured. In particular, in certain environments the NAMES.DEFAULT_DOMAIN property needs to be commented out in *SQLNET.ora*. Consult with your Oracle DBA, if needed.

11. Add Master 1's database information to the local *TNSNAMES.ora* on Master 2. Verify that Master 2's database information exists in the local *TNSNAMES.ora* on Master 2. If Master 2's database information does not exist, add it same way as other entries. (Refer to the note above.)

12. Ensure that you have a server and a directory (typically on one of the database system), where you can run a copy of SQLPlus that can access all NCM Cores in the NCM Mesh. To verify this, connect to one database with SQLPlus and issue the `CONNECT <USER>/<PASSWORD>@<SID>` command for each different SID in the Mesh. This is necessary because the script issues CONNECT commands to change which database it is running procedures against.

13. Copy the newly generated SQL file (typically named *ReplicationScript.sql* depending on what value was set in the *ReplicationScriptTool.properties* file), from the ReplicationScriptTool working directory to the server and directory where you plan to run SQLPlus.

    Note: Make sure that files named **_spool.txt* do not exist in the directory. When run, the script will log its actions to a file with that suffix.

Implementation

1. Stop the NCM Management Engine that is accessing the database on Master 1.

   Note: On Oracle 10g, steps 2 to 4 below could be done more efficiently by using Oracle's Data Pump Utility. Refer to "Appendix C: Data Pump Utility" on page 81 for information.

2. Dump the Master 1 database using the following command on the database server for Master 1:
   ```
   EXP <USER>/<PASSWORD>@<MASTER1_SID> TABLES=RN_% FILE=<filename>
   ```

   where `<USER>`, `<PASSWORD>`, and `<MASTER1_SID>` were collected in Step 2.

   Note: Make sure you log the output of this command to a file for examination later if there are errors.

3. Copy the file from Step 2 to the database server for Master 2.

4. Import into the Master 2 database using the following command on the database server for Master 2:
   ```
   IMP <USER>/<PASSWORD>@<MASTER2_SID> TABLES=RN_% FILE=<filename>
   ```

   where `<USER>`, `<PASSWORD>`, and `<MASTER2_SID>` were collected in Step 2.

5. On either Master 1 or Master 2, ensure that you have two SQLPLus windows open before running the script: One connected as the NCM user and one connected as the SYSTEM user.

6. Run the replication script in the SQLPlus window connected as the NCM user using `@ReplicationScript.sql`.

7. Follow the instructions provided by the script at several points as it runs. This will involve running other commands in the second SQLPlus window to verify that certain operations necessary for later steps in the script have completed.

   Note: If you skipped or misconfigured any of the above steps, and as a result see errors during replication setup, refer to "Removing Replication Across the Entire NCM Mesh" on page 38 before performing initial replication setup again.

8. Exit from SQLPlus once the script completes. This ensures that all actions are logged to the spool file.

9. Verify that the replication is working correctly. Make sure:

   - The spool file generated by running the replication SQL script contains no errors.

   - The RN_CORE table on each database contains the appropriate list of servers in the NCM Mesh. To check the RN_CORE table, using SQLPlus, enter the following command on each server:
     `"SELECT * FROM RN_CORE";`

   - Verify (using Oracle's Enterprise Manager) that each database in the NCM Mesh has replication set up, is in the "Normal" state, and has purge, push, and repcat_admin jobs installed correctly. Consult your Oracle DBA for assistance.

– Verify that the replication of data is working correctly. To verify that replication is working between all NCM Cores in the NCM Mesh, enter the following commands:

```
CONNECT <USER>/<PASSWORD>@<MASTER1_SID>;
UPDATE RN_CORE SET COMMENTS = 'Test from sid1 to sid2'
where CoreID=1;
COMMIT;
```

wait 10 to 30 seconds

```
CONNECT <USER>/<PASSWORD>@<MASTER2_SID>;
SELECT Comments from RN_CORE where CoreID=1;
```

You should enter this from every NCM Core in the NCM Mesh. Be sure to check the values each time on all of the other NCM Cores to ensure that the links between each database in the system are working correctly.

10. Install a NCM server, pointing it to an existing database on Master 2.

11. Stop the NCM Management Engine running on Master 2.

12. Copy the *distributed.rcx* file to each NCM server in the NCM Mesh (in the same location as *appserver.rcx*).

13. Start the NCM Management Engine on NCM Core 1.

14. Start the NCM Management Engine on NCM Core 2.

15. Verify the NCM installation:

– Edit an object on one NCM server (for example, a Comments field for a device).

– Wait a minute and then verify that the updated comment exists on the second server.

– Check the status of the Oracle Distributed Monitor in the NCM UI to ensure that no problems are being reported. (Refer to "Distributed Monitor Results Page" on page 56 for information.) It could take up to five minutes for this monitor to initially run.

Adding a New NCM Core to an Existing NCM Mesh

Master 1 is the existing Master Definition Site and Master 3 is the new NCM Core being added.

1. System Setup:

   a) A current up-to-date 1.4 database on Master 1.

   b) A NCM server of NCM Core 1 connected to the database on Master 1.

   c) A database with no data or NCM schema setup on Master 3. The database identifier (SID) needs to be different from the existing databases in the NCM Mesh.

   d) A network connection from the Master 1 servers to the Master 3 servers (and vice-versa) that enables ports 1521 (Oracle only), 1098, 8083, 1099, and 4444 (or appropriate variations) to be connected between these servers. Port 1521 must be open between the Oracle servers. The remaining ports must be open between the NCM servers.

   e) The *ReplicationScriptTool bundle* files installed on a Java-capable system.

2. Obtain the following information:

   a) Login/password/database identifier for NCM into the two databases (on Master 1 and Master 3). This is necessary for Steps 5, 6, 11, 13, and 14 below.

   b) The replication password for the existing repadmin user. This is necessary for Steps 9 and 14 below.

   c) NCM server hostname, NCM server RMI listening port, database hostname, and database listening port for Master 3. This is necessary for Step 14 below.

   d) The timezone offset (integer from UTC) for the entire NCM Mesh. This must be a constant across the NCM Mesh. This is necessary for Step 14 below.

   e) The existing replication group name. This is necessary for Steps 9 and 14 below.

3. Set (or verify) the following database parameters for Master 3:

Note: These parameters could have been be set up when the database is first created. If not, they must be modified at this point.

a) global_names = TRUE

b) open_links = at least 5 (increase this by 1 for each additional server in the NCM Mesh beyond 2)

c) open_links_per_instance = at least 5 (same as open_links)

d) parallel_min_servers = 2

e) parallel_automatic_tuning = TRUE (for Oracle 9i databases only)

f) shared_pool_size = increase current value by 40m to support replication

g) Job_queue_processes = at least 2.

4. Increase the open_links and open_links_per_instance on existing databases in the NCM Mesh.

5. Add Master 3's database information to the local *TNSNAMES.ora* on Master 1. Verify that Master 1's database information exists in the local *TNSNAMES.ora* on Master 1. If Master 1's database information does not exist, add it same way as other entries.

Note: You may need to make modifications to the *LISTENER.ora* and *SQLNET.ora* files depending on how your Oracle servers are configured. In particular, in certain environments, the NAMES.DEFAUL_DOMAIN property needs to be commented out in *SQLNET.ora*. Consult with your Oracle DBA, if needed.

6. Add Master 1's database information to the local *TNSNAMES.ora* on Master 3. Verify that Master 3's database information exists in the local *TNSNAMES.ora* on Master 3. If Master 3's database information does not exist, add it same way as other entries. (Refer to the note above.)

Note: You may need to make modifications to the *LISTENER.ora* and *SQLNET.ora* files depending on how your Oracle servers are configured. Consult with your Oracle DBA.

7. Ensure that all database server and NCM server system are set to use the same time and timezone.

8. Turn off all NCM servers in the existing NCM Mesh.

9. As the **repadmin** user, enter the following command using SQLPlus on Master 1 to quiesce the replication group:

    ```
    EXEC DBMS_REPCAT.SUSPEND_MASTER_ACTIVITY('<replication_group_name>');
    ```

    where `<replication_group_name>` is from Step 2.

10. Wait for the replication group to quiesce. Do this by executing the `SELECT GNAME,STATUS FROM DBA_REPGROUP` command to check on the status. Wait until the status is listed as `QUIESCED`.

    Note: On Oracle 10i, steps 11 to 13 below could be done more efficiently by using Oracle's Data Pump Utility. Refer to "Appendix C: Data Pump Utility" on page 81 for information.

11. Dump the Master 1 database using the following command on the database server for Master 1:

    ```
    EXP <USER>/<PASSWORD>@<MASTER1_SID> TABLES=RN_% FILE=<filename>
    ```

    where `<USER>`, `<PASSWORD>`, and `<MASTER1_SID>` were collected in Step 2.

12. Copy the file <filename> to the database server for Master 3.

13. Import into the Master 3 database using the following command on the database server for Master 3:

    ```
    IMP <USER>/<PASSWORD>@<MASTER3_SID> TABLES=RN_% FILE=<filename>
    ```

    where `<USER>`, `<PASSWORD>, and <MASTER3_SID>` were collected in step 2.

14. Update the variables for all masters in the *ReplicationScriptTool.properties* file. Ensure that you modify all "REPLACEME" text entries in the *ReplicationScriptTool.properties* file with the appropriate values for your environment.

    Note: The master being added must be the last master entry and Master Definition Site must be the first entry. In addition, be sure to update the replication user, password, replication group name, and timezone offset in the *ReplicationScriptTool.properties* file if needed. Make sure the "mode" property is set to "add_server". All other masters should also be listed.

15. Run the ReplicationScriptTool via the following command:

    ```
    java -classpath . ReplicationScriptTool
    ```

16. Ensure that you have a server and a directory (typically on one of the database system), where you can run a copy of SQLPlus that can access all NCM Cores in the NCM Mesh. To verify this, connect to one database with SQLPlus and issue the `CONNECT <USER>/<PASSWORD>@<SID>` command for each different SID in the NCM Mesh. This is necessary because the script issues CONNECT commands to change which database it is running procedures against.

17. Copy the newly generated SQL file (typically named *ReplicationScript.sql* depending on what value was set in the *ReplicationScriptTool.properties* file), from the ReplicationScriptTool working directory to the server and directory where you plan to run SQLPlus.

    Note: Make sure that files named **_spool.txt* do not exist in the directory. When run, the script will log its actions to a file with that suffix

18. Ensure that you have two SQLPLus windows open before running the script: One connected as the NCM user and one connected as the SYSTEM user.

19. Run the replication script copied in the previous step in the SQLPlus window connected as the NCM user using `@ReplicationScript.sql`.

20. Follow the instructions provided by the script at several points as it runs. This will involve running other commands in the second SQLPlus window to verify that certain operations necessary for later steps in the script have completed.

    Note: If you skipped or misconfigured any of the above steps, and as a result see errors during replication setup, refer to "Removing Replication Across the Entire NCM Mesh" on page 38 before performing initial replication setup again.

21. Exit from SQLPlus once the script completes. This ensures that all actions are logged to the spool file.

22. Verify that the replication is working correctly. Make sure:

    – The spool file generated by running the replication SQL script contains no errors, other than the following warning: `'ORA-23326: object group <yourgroup>.<your_replication_group> is quiesced'` that occurs after the DBMS_REPCAT.SUSPEND_MASTER_ACTIVITY command.

- The RN_CORE table on each database contains the appropriate list of servers in the NCM Mesh. To check the RN_CORE table, using SQLPlus, enter the following command on each server:
`SELECT * FROM RN_CORE;`

- Verify (using Oracle's Enterprise Manager) that each database in the NCM Mesh has replication set up and is in the "Normal" state, and has purge, push, and repcat_admin jobs installed correctly. Consult your Oracle DBA for assistance.

- Verify that the replication of data is working correctly. To verify that replication is working between all NCM Cores in the NCM Mesh, enter the following commands:

```
CONNECT <USER>/<PASSWORD>@<MASTER1_SID>;
UPDATE RN_CORE SET COMMENTS = 'Test from sid1 to sid3'
where CoreID=1;
COMMIT;
```

wait 10 to 30 seconds

```
CONNECT <USER>/<PASSWORD>@<MASTER3_SID>;
SELECT Comments from RN_CORE where CoreID=1;
```

You should enter this from every NCM Core in the NCM Mesh. Be sure to check the values each time on all of the other NCM Cores to ensure that the links between each database in the system are working correctly.

23. Install a NCM server. Be sure to point it to an existing database on Master 3.

24. Add the *distributed.rcx* file to the new NCM server in the NCM Mesh (in the same location as *appserver.rcx*).

25. Start all of the NCM servers in the NCM Mesh.

26. Verify the NCM installation. To do this:

- Edit an object on one NCM server (for example, a Comments field for a device).

- Wait a minute and then verify that the updated comment exists on the second server.

– Check the status of the Oracle Distributed Monitor in the NCM UI to ensure that no problems are being reported. (Refer to "Distributed Monitor Results Page" on page 56 for information.)

Removing a NCM Core from an Existing NCM Mesh

Warning: *Once you remove a NCM Core from a NCM Mesh, the data in that NCM Core's database will no longer remain in sync with the data in the rest of the NCM Mesh. The data cannot be merged back into the NCM Mesh.*

To remove a NCM Core from a NCM Mesh:

1. If the NCM core you want to remove is the Master Definition Site, change the Master Definition Site to be another NCM Core. Refer to the instructions below.

2. Install the *ReplicationScriptTool bundle* files on a Java-capable system.

3. Ensure that all NCM devices belong to Sites on NCM Cores that are not going to be removed.

4. Modify all Sites to point to a NCM Core that is not going to be removed. (Alternatively, you can remove those Sites.)

5. Turn off the NCM server from the NCM Core that is being removed.

6. Delete the RN_CORE entry that was removed using the following commands using SQLPlus:

   Note: To identify the Core ID of the NCM Core to delete, enter the `SELECT*FROM RN_CORE` command.

   ```
   UPDATE RN_SITE SET OwningCoreID = 1 WHERE OwningCoreID = <coreID>;
   UPDATE RN_SITE SET ManagingCoreID = 1 WHERE ManagingCoreID = <coreID>;
   UPDATE RN_SCHEDULE_TASK SET CoreID = 1 WHERE CoreID = <coreID>;
   DELETE FROM RN_CORE WHERE CoreID = <coreID>;
   COMMIT;
   ```

7. Update the variables for all masters in the
   ReplicationScriptTool.properties file. Ensure that you modify all
   "REPLACEME" text entries in the *ReplicationScriptTool.properties* file
   with the appropriate values for your environment.

   Note: The master being deleted must be the last master entry and the Master
   Definition Site must be the first entry. In addition, update the replication user,
   password, and replication group name in the *ReplicationScriptTool.properties*
   file if needed. Make sure the "mode" property is set to "delete_server".

8. Run the ReplicationScriptTool via the following command:
   ```
   java -classpath . ReplicationScriptTool
   ```

9. Ensure that you have a server and directory (typically on one of the
   database systems) where you can run a copy of SQLPlus that can access
   all NCM Cores in the NCM Mesh. To verify this, connect to one database
   with SQLPlus and issue the `CONNECT <USER>/<PASSWORD>@<SID>`
   command for each different SID in the NCM Mesh. This is necessary
   because the script issues `CONNECT` commands to change which database
   it is running procedures against.

10. Copy the newly generated SQL file (typically named *ReplicationScript.sql*
    depending on what value was set in the *ReplicationScriptTool.properties*
    file), from the ReplicationScriptTool working directory to the server and
    directory where you plan to run SQLPlus.

    Note: Make sure that files named "*_spool.txt" do not exist in that directory.
    When run, the script will log its actions to a file of that suffix.

11. Ensure that you have two SQLPLus windows open before running the
    script: One connected as the NCM user and one connected as the
    SYSTEM user.

12. Run the replication script copied in the previous step in the SQLPlus
    window connected as the NCM user using `@ReplicationScript.sql`.

13. Follow the instructions provided by the script at several points as it runs.
    This will involve running other commands in the second SQLPlus window
    to verify that certain operations necessary for later steps in the script
    have completed.

14. Exit from SQLPlus once the script completes. This ensures that all
    actions are logged to the spool file.

15.Verify that the replication is working correctly. Make sure:

- The spool file generated by running the replication SQL script contains no errors.

- The RN_CORE table on each database contains the appropriate list of servers in the NCM Mesh. To check the RN_CORE table, using SQLPlus, enter the following command on each server:
  ```
  SELECT * FROM RN_CORE;
  ```

- Verify (using Oracle's Enterprise Manager) that each database in the NCM Mesh has replication set up, is in the "Normal" state, and has purge, push, and repcat_admin jobs installed correctly. Consult your Oracle DBA for assistance.

- Verify that the replication of data is working correctly. To verify that replication is working between all NCM Cores in the NCM Mesh, enter the following commands:

  ```
  CONNECT <USER>/<PASSWORD>@<MASTER1_SID>;
  UPDATE RN_CORE SET COMMENTS = 'Test from sid1 to sid2'
  where CoreID=1;
  COMMIT;
  ```

  wait 10 to 30 seconds

  ```
  CONNECT <USER>/<PASSWORD>@<MASTER2_SID>;
  SELECT Comments from RN_CORE where CoreID=1;
  ```

  You should enter this from every NCM Core in the NCM Mesh. Be sure to check the values each time on all of the other NCM Cores to ensure that the links between each database in the system are working correctly.

16.Uninstall the NCM server from the NCM Core that was removed.

17.Restart the NCM servers in the NCM Mesh.

Changing the Master Definition Site

It may be necessary to change the Master Definition Site in the event of failure of that Site or when you want to remove a NCM Core from the NCM Mesh. To change the Master Definition Site:

1. If the old Master Definition Site is accessible (for example in the case where it is being removed from the NCM Mesh), as the "repadmin" user on the existing Master Definition Site, enter the following command using SQLPlus:

```
BEGIN
    DBMS_REPCAT.RELOCATE_MASTERDEF (
                                     gname => '<REPLICATION GROUP NAME>',
                                     old_masterdef => '<OLD MASTERDEF SID>',
                                     new_masterdef => '<NEW MASTERDEF SID>',
                                     notify_masters => TRUE,
                                     include_old_masterdef => TRUE);
END;
/
COMMIT;
```

2. If the old Master Definition Site is not accessible (for example in the case of failure of the Master Definition database server), as the "repadmin" user on the existing Master Definition Site, enter the following command using SQLPlus on any other NCM Core in the NCM Mesh:

```
BEGIN
    DBMS_REPCAT.RELOCATE_MASTERDEF (
                                     gname => '<REPLICATION GROUP NAME>',
                                     old_masterdef => '<OLD MASTERDEF SID>',
                                     new_masterdef => '<NEW MASTERDEF SID>',
                                     notify_masters => TRUE,
                                     include_old_masterdef => FALSE);
END
/
COMMIT;
```

3. For both cases, as the NCM database login user, modify the RN_CORE table entries:

```
UPDATE RN_CORE SET IsMasterDef = 0 WHERE CoreID = <ID OF OLD MASTERDEF>;
COMMIT;

UPDATE RN_CORE SET IsMasterDef = 1 WHERE CoreID = <ID OF NEW MASTERDEF>;
COMMIT;
```

Removing Replication Across the Entire NCM Mesh

Warning: *All NCM servers participating in the NCM Mesh will now operate as independent NCM installations. The data cannot be merged once they are no longer in sync.*

To remove replication across the entire NCM Mesh:

1. Make sure that the *ReplicationScriptTool bundle* files are installed on a Java-capable machine.

2. Turn off the NCM server on all NCM Cores.

3. Update the variables for all masters in the *ReplicationScriptTool.properties* file. Ensure that you modify all "REPLACEME" text entries in the *ReplicationScriptTool.properties* file with the appropriate values for your environment.

   Note: The Master Definition Site must be the first entry. In addition, update the replication user, password, and replication group name in that file if needed. Make sure the "mode" property is set to "delete".

4. Run the ReplicationScriptTool via the following command:
   `java –classpath . ReplicationScriptTool`

5. Ensure that you have a server and directory (typically on one of the database systems) where you can run a copy of SQLPlus that can access all NCM Cores in the NCM Mesh. To verify this, connect to one database with SQLPlus and issue the `CONNECT <USER>/<PASSWORD>@<SID>` command for each different SID in the NCM Mesh. This is necessary because the script issues `CONNECT` commands to change which database it is running procedures against.

6. Copy the newly generated SQL file (typically named *ReplicationScript.sql* depending on what value was set in the *ReplicationScriptTool.properties* file), from the ReplicationScriptTool working directory to the server and directory where you plan to run SQLPlus.

   Note: Make sure that files named "*_spool.txt" do not exist in that directory. When run, the script will log its actions to a file of that suffix.

7. Ensure that you have two SQLPLus windows open before running the script: One connected as the NCM user and one connected as the SYSTEM user.

8. Run the replication script copied in the previous step in the SQLPlus window connected as the NCM user using `@ReplicationScript.sql`.

9. Follow the instructions provided by the script at several points as it runs. This will involve running other commands in the second SQLPlus window to verify that certain operations necessary for later steps in the script have completed.

10. Exit SQLPlus once the script completes (this ensures that all actions are logged to the spool file).

11. Verify that the replication has been removed correctly. Make sure:

   – The spool file generated by running the replication SQL script contains no errors.

   – Verify (using Oracle's Enterprise Manager) that each database in the NCM Mesh does not have replication setup. Consult your Oracle DBA for assistance.

12. For each server, ensure that devices are deleted or associated with Sites that map to the local NCM Core as needed.

13. Remove the *distributed.rcx* file from each NCM server machine.

14. Restart the NCM servers.

15. For each database, remove all entries from RN_CORE except for the local NCM Core system's entry.

16. Restart each NCM server to enable the NCM Core changes to take effect.

Post Installation Setup

Once you have a functioning replication system, there are additional steps you can take to complete setup:

1. Add new Sites — This will enable you to partition your devices across the different NCM Cores in the NCM Mesh.

2. Add new Realm definitions — A Realm is a network segment. A Site is not required to be in the same Realm as its managing NCM Core. Keep in mind that a Realm is a large area that can include many Sites. However, a Realm does not have to include any NCM Cores. Typically, a Realm is identified by a set of unique IP addresses. For example, a Realm cannot contain two devices numbered as 10.255.111.128. Instead, the devices must be broken out into separate Realms. (Refer to the *User Guide for Network Compliance Manager 1.4* for information.)

Upgrading from NCM 1.3/1.3.1 to NCM 1.4

During the NCM upgrade process, the NCM Cores need to be completely offline. You will be upgrading the:

- NCM application servers
- NCM database schema
- Replication between the database cores

Note: You will need to update each NCM Core in your system.

Keep in mind that these upgrade instructions only work on a current, up-to-date NCM 1.3/1.3.1 database running on all servers. Refer to the NCM 1.3 documentation for instructions.

To upgrade from NCM 1.3/1.3.1 to NCM 1.4, be sure that:

- A current, up-to-date NCM 1.3/1.3.1 database is installed on all Masters.
- The *ReplicationScriptTool* bundle files are installed on a Java-capable system (on one of the NCM systems).
- You have the following information:
  - Login/password/DatabaseIdentifier (SID) for NCM into all databases in the NCM Mesh
  - Replication password for the existing repadmin user
  - DatabaseIdentifier (SID), NCM server hostname, NCM server RMI listening port, database server hostname, and database listening port for all Masters
  - The timezone offset (integer from UTC) for the entire NCM Mesh (This must be a constant across the NCM Mesh.)
  - The existing replication group name

1. Shutdown all NCM Management Engines in the NCM Mesh.

2. Update the variables for all Masters in the
   R*eplicationScriptTool.properties* file. Ensure that you modify all
   "REPLACEME" text entries in the *ReplicationScriptTool.properties* file
   with the appropriate values for your environment.

   Note: All servers should be listed in the *ReplicationScriptTool.properties* file.
   Make sure the mode property is set to the appropriate value for the upgrade
   you want to perform. For example, if you are upgrading from NCM 1.3 to NCM
   1.4, this value should be set to "upgrade_from_1_3". If you are upgrading from
   NCM 1.3.1 to NCM 1.4, this value should be set to "upgrade_from_1_3_1".

3. Run the ReplicationScriptTool. Enter:
   ```
   java -classpath . ReplicationScriptTool
   ```

4. Ensure that you have a server and a directory (typically on one of the
   database systems), where you can run a copy of SQLPlus that can
   access all NCM Cores in the NCM Mesh. To verify this, connect to one
   database with SQLPlus and issue the CONNECT <USER>/
   <PASSWORD>@<SID> command for each unique SID in the NCM Mesh.
   This is necessary because the script issues CONNECT commands to
   change which database it is running procedures against.

5. Copy the newly generated SQL file (typically named
   ReplicationScript.sql, depending on what value was set to in the
   ReplicationScriptTool.properties file), from the ReplicationScriptTool
   working directory to the server and directory where you plan to run
   SQLPlus.

   Note: Make sure that files named *_spool.txt do not exist in the directory.
   When run, the script will log its actions to a file with that suffix.

6. Ensure you have two SQLPLus windows open before running the script:
   One window connected as the NCM user and the other window
   connected as the SYSTEM user.

7. Run the replication script copied in the previous step in the SQLPlus
   window connected as the NCM user using @ReplicationScript.sql.

8. Follow the instructions provided by the script at several points as it runs. This will include running other commands in the second SQLPlus window to verify that certain operations necessary for later steps in the script have completed.

   Note: If you skipped or miss-configured any of the above steps, and as a result see errors during replication setup, refer to "Removing Replication Across the Entire NCM Mesh" on page 38 before performing the upgrade.

9. Exit from SQLPlus once the script completes. This ensures that all actions are logged to the spool file.

10.Verify that the replication is working correctly. Make sure:

   – The spool file generated by running the replication SQL script contains no errors, other than the following warning: 'ORA-23326: object group <yourgroup>.<your_replication_ group> is quiesced' that occurs after the DBMS_REPCAT.SUSPEND_MASTER_ACTIVITY command.

   – Verify (using Oracle's Enterprise Manager) that each database in the NCM Mesh has replication set up and is in the "Normal" state, and has purge, push, and repcat_admin jobs installed correctly. Consult your Oracle DBA for assistance.

   – Verify that the replication of data is working correctly. To verify that replication is working between all NCM Cores in the NCM Mesh, enter the following commands:

   ```
   CONNECT <USER>/<PASSWORD>@<MASTER1_SID>;
   UPDATE RN_CORE SET COMMENTS = 'Test from sid1 to sid2'
   where CoreID=1;
   COMMIT;
   ```

   wait 10 to 30 seconds

   ```
   CONNECT <USER>/<PASSWORD>@<MASTER2_SID>;
   SELECT Comments from RN_CORE where CoreID=1;
   ```

   You should enter this from every NCM Core in the NCM Mesh. Be sure to check the values each time on all of the other NCM Cores to ensure that the links between each database in the system are working correctly.

11. Upgrade each NCM server using the NCM 1.4 Service Pack Installer. The NCM 1.4 Service Pack Installer upgrades the NCM server to NCM 1.4. The scripts for the database upgrade have already been handled by running the replication script in Step 7.

12. Verify the NCM installation:

   a) Edit an object on one NCM server (for example, a Comments field for a device).

   b) Wait a minute and then verify that the updated comment exists on the second server.

   c) Check the status of the Oracle Distributed Monitor in the NCM UI to ensure that no problems are being reported. (Refer to "Distributed Error List" on page 58 for information.)

Chapter 3: System Administration

Use the following table to quickly locate information in this chapter.

Getting Started

In general, a NCM server that is part of a High Availability Distributed System NCM Mesh should be transparent to users. However, there are a number of operations that the system administrator may need to do to keep the High Availability Distributed System NCM Mesh functioning properly.

Understanding Oracle Replication

Oracle replication uses three types of jobs:

- **Transaction Push Job** — Each NCM Core will have a Transaction Push job for each of the other NCM Cores in the NCM Mesh. For example, if you have two NCM Cores, each NCM Core has one Transaction Push job. If you have three NCM Cores, each NCM Core has two Transaction Push jobs for the two destination NCM Cores. These jobs are identified in Oracle's Enterprise Manager or from DBA_JOBS by the `DBMS_DEFER_SYS.PUSH()` command string.

- **Transaction Purge Job** — Each NCM Core will have one Transaction Purge job for cleaning out successfully pushed transactions. These jobs can be identified in Oracle's Enterprise Manager or from DBA_JOBS by the `DBMS_DEFER_SYS.PURGE ()` command string.

- **Replication Administration Job** — Each NCM Core will have one Replication Administration job. These jobs are not used in normal NCM operation. It is used if the replication setup is changed. These jobs can be identified in Oracle's Enterprise Manager or from DBA_JOBS by the `DO_DEFERRED_REPCAT_ADMIN()` command string.

NCM automatically monitors all three jobs and attempts to restart any job that fails. You can monitor the jobs using Oracle's Enterprise Manager or by viewing DBA_JOBS. Using SQLPlus, enter: `SELECT * FROM DBA_JOBS;`

To view specific job IDs and if they are broken, enter: `SELECT JOB, BROKEN, WHAT FROM DBA_JOBS;`

On Oracle, NCM monitors on a regular schedule the following on each NCM Core for the database transport (i.e., replication):

- Push jobs that handle moving data from one NCM Core to another. If the monitor check fails and the Push job is found to be broken, NCM will attempt to restart it.

- Purge jobs that successfully clear pushed data from the push queue, and attempts to restart the Push queue each time the check fails.

- The deferred transaction queue for any transactions that could not be applied to the local database due to a conflict. NCM applies additional application logic to attempt to correctly apply the transaction. If unsuccessful, NCM alerts the NCM administrator via email notification.

- Unrecoverable uniqueness conflicts. NCM alerts the NCM administrator in the event of unrecoverable uniqueness conflicts. Typically, there is no way to automatically correct these types of issues.

- Both the DEFTRAN and DEFLOB queue size. These queues hold pending data that needs to be pushed to other servers. Typically, if replication is working, but is beginning to backup, these queues will begin to grow. This can occur if the connection between databases is congested. NCM alerts the NCM administrator to this issue via email notification. The system will continue to push data. However, it could take a long time to catch up depending on network conditions.

Note: Refer to the *User Guide for Network Compliance Manager 1.4* for information on configuring email notification.

In addition to replication monitoring, NCM also monitors the following for each of the NCM Cores:

- RMI connectivity between each NCM Core
- NCM server timestamp differences between each NCM Core

These monitors will generate events during error conditions. The events can be emailed to the NCM administrator or Oracle DBA using a standard event rule in NCM.

Note: There is a standard example event rule shipped with NCM. The event rule can be updated as necessary.

How NCM Removes In-Memory and Database Information

There could be a delay in removing or cancelling a task because NCM must communicate with the appropriate owning NCM Core to perform this operation. To ensure proper removal of all in-memory and database information, and to avoid replication conflicts due to matching timestamps, NCM active tasks are automatically deleted on the NCM Core with which they are associated. An active task is any task that does not have the "Succeeded," "Failed," "Duplicate", "Skipped", or "Warning" status.

Keep in mind that deleted tasks could be displayed in task lists for a few moments while the replication process pushes the deletes to other NCM Cores in the system. In addition, if the NCM Core from which the task originated is not accessible, the delete will fail.

Suspending Replication for Database Maintenance

To suspend replication because you are bringing down one or both database servers for maintenance:

1. Turn off all NCM servers accessing the databases.
2. CONNECT repadmin/<PWD>@<MASTER1_SID>;
3. EXEC DBMS_REPCAT.SUSPEND_MASTER_ACTIVITY(<REPGROUP>);

   Note: This must be done as repadmin from the masterdef database site.

NCM Generated Events

By default, NCM generates system events. Event rules can alert you to certain error conditions requiring attention. Each event is listed below, along with an explanation and required action to be taken. You should examine the default "Distributed System" event rule to ensure all of the events are included in the event rule and that the event rule is configured to send the email notification to the appropriate administrator.

Note: Refer to the *User Guide for Network Compliance Manager 1.4* for information on configuring email notification.

Broken Replication Job

Event format:

```
Job ID: <ID>
Job Type: <push to <host>, purge, or repcat admin>
Database Host: <hostname>
Database SID: <SID>
Stalled: <true or false>
```

Broken jobs can occur because an Oracle server cannot push transactions to another server due to that server being down or in the event of a network connectivity loss. Replication jobs can also become broken because they stall. Stalled replication jobs represent a hang in an internal Oracle process. This event is sent only once for each broken replication job (unless the NCM server is restarted, in which case the event will be resent if the job is still broken). If a stalled replication job is detected, it is best to restart the Oracle server.

For broken replication jobs that are not stalled, the course of action depends on the job type. If the job type is "purge" or "repcat admin," consult with your Oracle DBA. Failure of these jobs represents an internal problem or resource limit in the Oracle server that will need to be investigated.

For "push to <host>" job types, the typical failure will be loss of connectivity to the other host, either through a network problem, outage of the Oracle server, or outage of the database host system itself. NCM will restart this job automatically once the connectivity problem has been remedied.

To troubleshoot this problem, make sure:

1. The host that the server cannot connect to is up and running.

2. The Oracle instance on that host is running.

3. From a command line enter `ping <host>` to ensure that network connectivity exists between servers.

4. From a command line enter `tnsping <OTHER_SID>` to ensure that Oracle is accepting connections.

5. Ensure that database links are working:

   - Using SQLPlus, enter `SELECT CURRENT_TIMESTAMP FROM DUAL@<OTHER_SID>;`

   - Using SQLPlus, enter `SELECT DBLINK FROM DBA_REPSITES;`

Failures of any of these steps will point to corrective actions needed.

Fixed Replication Job

Event format:

```
Job ID: <ID>
Job Type: <push to host, purge, or repcat admin>
Database Host: <hostname>
Database SID: <SID>
```

This event is sent when the system detects that a previously broken job is now working correctly. This could happen when network connectivity is restored between two database servers, allowing data to be replicated. No action is required when this happens.

Note: This event is only sent when the correction occurs while NCM is running. If NCM is stopped while the problem is corrected, when NCM is restarted no event is sent.

Replication Errors

Event format:

```
TransactionID: <id>
Call Number: <n>
Table Name: <TABLE_NAME>
Transaction Type: UPDATE
NCM Source Host: <hostname>
DB Destination: <SID>
```

NCM sends this event when it encounters a replication error that it cannot resolve on its own. In general, NCM attempts to resolve conflicts automatically. Certain conflict types cannot be resolved using Oracle's built-in methods, but can be resolved by NCM after they occur. In particular, UPDATE conflicts due to out-of-order transactions, causing a primary key or foreign key constraint violation, can be resolved by re-executing the transaction after a short wait.

Additionally, conflicts due to UPDATES of items that have already been deleted can by resolved by removing the failed transaction. These two actions are taken by NCM during its regular replication monitoring run, which happens at several minute intervals. If these corrective actions do not fix the problem, an event is sent.

Typically, to fix an UPDATE error requires editing the object in question to set its values to those expected by the transaction, and then re-executing the transaction (also known as retrying the error). To do this:

1. Open the View Distributed Error page and click the Details option in the Actions column. All of the database attributes of the affected object are displayed. Refer to "Distributed Conflict List" on page 60.

2. Locate the object in the NCM system (or simply update the database directly using SQLPlus or some other tool), and change all of its attributes to those listed in the Original Value column on the View Distributed Error Details page.

3. Retry the error.

Uniqueness Conflict

```
Event format:
Server: <host>
Object: <Object>
Conflicting Value: <value>
Refer to the replication documentation for instructions on
correcting this conflict.
```

This event is sent when NCM detects a conflict in a uniqueness constraint. You will receive an event per NCM Core, since the conflicts are local to each NCM Core. To correct a naming conflict, go to one NCM Core and update the names for the affected objects. Both the renamed <NAME>.<SID> and <NAME> should be edited to force an update on the other NCM Cores.

To correct a rule priority conflict, go to the Device Password Rule Priority Reset page click the Reset Priority button. Refer to "Device Password Rule Priority Reset Page" on page 64.

If this does not solve the problem, you will need to manually edit the rules on each NCM Core, setting the priority order correctly and verifying existence of correct rules. When finished, return to the Device Password Rule Priority Reset page and click the Reset Priority button.

Time Synchronization Warning

Event format:

```
  Time difference: <N> seconds
  Local Core: <hostname>
  Remote Core: <hostname>
```

NCM replication conflict resolution depends on a latest timestamp method. To work correctly, this requires different NCM servers' clocks to differ by only a small amount. To correct this problem, make sure that the time is synchronized on the NCM server systems across the NCM Mesh.

RMI Error

Event format:

```
Local Core: <hostname>
Remote Core: <hostname>
Error: <Exception text>
```

This error typically occurs when there are network problems between the NCM servers. To troubleshoot this problem, make sure:

1. The host that the server cannot connect to is up and running.

2. The NCM instance on that host is running.

3. From a command line, enter `ping <host>` to ensure that network connectivity exists between servers.

4. From a command line, enter `telnet <host>` to port 1099 (or whatever your RMI listen port is set to) to ensure that RMI connections are being accepted. If working correctly, you should get back some data that includes the text string "java.rmi.MarshalledObject".

Failures of any of these steps will point to corrective actions needed, such as updating the RMI port being used in the Edit NCM Core page, or restarting NCM to make sure that the RMI port has been bound correctly and is not being used by another application.

Deferred Transactions Exceed Threshold

Event format:

```
Exceeded number of deferred transactions in threshold.
```

This event occurs when NCM determines that the deferred transaction queue, as reported by the DEFTRAN view, exceeds a certain threshold. This typically means that the NCM Core reporting this event cannot push transactions quickly enough to other NCM Cores.

To troubleshoot this issue:

1. Verify that network connectivity exists between the database servers in the NCM Mesh.

2. Verify that the network connections between the database servers in the NCM Mesh are not experiencing bandwidth congestion or other network issues.

Correcting these issues should enable the databases to catch up with their transaction pushes. In addition, reducing the amount of tasks or other operations on the NCM Cores can help the system recover more quickly.

Deferred LOBs Exceed Threshold

Event format:

```
Exceeded number of deferred LOBs in threshold.
```

This event is identical to Deferred Transactions Exceed Threshold event, except the view being monitored is the DEFLOB view.

Using the NCM Distributed System Pages

When you install the High Availability Distributed System software, the NCM user interface includes specific High Availability Distributed System pages to help you monitor and administer the system.

Navigating to Distributed Systems Pages

Distributed Monitor Results Page

The Distributed Monitor Results page displays the overall health of the Distributed System. By default, the Distributed monitor runs every five minutes.

To open the Distributed Monitor Results page, on the menu bar under Admin select Distributed and click Monitor Results. The Distributed Monitor Results page opens.

NCM monitor several properties necessary for proper functioning of the Distributed System, including:

- **RMI Connections** — RMI (Remote Method Invocation) is Java's remote procedure call protocol. The distributed system makes RMI calls between NCM servers in the NCM Mesh to transfer information about scheduled tasks, system settings, software images, and so on.

- **Scheduled Jobs** — Oracle contains a job scheduler. Replication requires certain jobs be setup using that job scheduler. In particular, there are jobs on each NCM Core database that:
    - Push transactions
    - Purge previously pushed transactions from the log
    - Handle administrative commands for replication

- **Error Transactions** — Error transactions are transaction that have been pushed from a remote server and for some reason were not able to execute correctly on the local server.

- **Uniqueness Conflicts** — Certain NCM database constraints restrict columns to unique values. In a distributed environment, these constraints can be violated when updates are made on two different NCM Cores where the unique column is set to the same value. These conditions are captured by the Replication Conflict Resolution System and logged. NCM cannot automatically resolve these conflicts. They must be resolved manually.

- **Local NCM Core Definition** — The local NCM Core must be able to determine which entry in the RN_CORE table it is. If the "The *local core for this system is undefined.*" error message is displayed, the CoreHostname property needs to be updated for the NCM Core. This can be done using the Edit Core page. Refer to for information.

  Note: When this condition occurs, the NCM Management Engine's log file will contain the following text: "*Fatal error - could not assign local core.*"

  The CoreHostname value can be either the DNS, *etc/hosts* value, or an IP address. If you are using a NCM server with multiple IP addresses, you might need to tell NCM which IP address to use. This is done by adding the following setting to the *distributed.rcx* file:

  ```
  <option name="distributed/NCM_server_local_ip">A.B.C.D</option>
  ```

  Note: The *distributed.rcx* file is located in the same location as the *appserver.rcx* file. Refer to for the location of the files.

  The value A.B.C.D should be replaced with the appropriate NCMT IP address for the NCM server and should match the RN_CORE table's CoreHostname value for that NCM Core.

Distributed Error List

The Distributed Error List page displays distributed transaction errors on the current NCM Core. This information is useful for finding error conditions that require administrative action. Keep in mind that all conflicts are displayed until they are:

- Manually resolved
- Automatically resolved by NCM
- Manually deleted without resolution (not recommended)

To open the Distributed Error List, on the menu bar under Admin select Distributed and click Error List. The Distributed Error List opens.

Distributed Error List Page Fields

Field	Description
Transaction ID	The ID of the transaction that was placed in the deferred error queue due to the conflict.
Call No.	A transaction can update more than one row in the database. Each row will have a separate call number. For example, for a transaction that updates seven rows, there will be seven entries with the same Transaction ID, but different call numbers.
Package Name	The name of the table on which the conflict occurred.
Proc Name	The type of transaction, for example UPDATE or DELETE.
Source	The database from which the transaction originated.
Status	Statuses can include:
	• new — This is a new error. The system has not attempted to deal with it yet. Keep in mind that certain types of errors require waiting for other transactions to complete before they can be retried.
	• alert_sent — The system has sent an alert.
	• to_retry — The system has noted this error and is waiting before retrying.
	• old — The system noted the error and retried it. However, the retry did not work.

Field	Description
Actions	You can select the following options: • Detail — Opens the View Distributed Error page, where you can view details about the transaction in that row. • Delete — Deletes the error from the database. • Retry — Re-execute the transaction.

View Distributed Error Page

The View Distributed Error page displays the detail for a particular distributed transaction error. This information is useful for providing the details necessary to correct the error.

To open the View Distributed Error page:

1. On the menu bar under Admin select Distributed and click Error List. The Distributed Error List opens.

2. In the Actions column, click the Detail option. The View Distributed Error page opens. The Details section displays the column(s) from the table named in the Package Name field that contains conflicting data.

Distributed Conflict List

The Distributed Conflict List page displays the uniqueness constraint conflict list. This provides information about uniqueness conflicts that will need to be manually corrected to ensure that the databases in the NCM Mesh are in sync.

To open the Distributed Conflict List, on the menu bar under Admin select Distributed and click Conflict List. The Distributed Conflict List opens.

Distributed Conflict List Page Fields

Field	Description
Object	The database table on which the conflict occurred.
Site Name	The core in the NCM Mesh for which the conflict occurred (actually the database SID).
Value	The value of the column for which there was a uniqueness conflict. Keep in mind that NCM does not automatically resolve uniqueness conflicts. NCM simply appends the name of the NCM Core from which the transaction originated to the conflicting value so that the transaction can complete.
Create Date	The Date the conflict occurred.
Status	Statuses include: • new — This is a new conflict. The system has not yet alerted the system administrator. • alert_sent — The system has sent an alert.
Actions	You can select the following options: • Detail — Opens the View Distributed Conflict page, where you can view details on an individual uniqueness constraint. Refer to "View Distributed Conflict Page" on page 61. • Delete — Deletes the conflict from the database.

View Distributed Conflict Page

The View Distributed Conflict page provides details on a specific uniqueness constraint.

To open the View Distributed Conflict page:

1. On the menu bar under Admin select Distributed and click Conflict List. The Distributed Conflict List opens.

2. In the Actions column, click the Detail option. The View Distributed Conflict page opens.

Table Name is the name of the database table on which the uniqueness conflict occurred. Object IDs are the primary keys of the objects in the database.

If a key has only one column, it is displayed in Object ID 1 field. Some keys are formed from multiple columns. Consequently, those values are displayed in the Object ID 1, Object ID 2, and Object ID 3 fields.

Site Reassignment Page

The Site Reassignment page allows the Site-to-NCM Core mapping to be modified. This is useful for failover of Sites from one NCM Core to another and for restoring Sites back to their original NCM Core.

To open the Site Reassignment page, on the menu bar under Admin select Distributed and click Site Reassignment. The Site Reassignment opens. You can select NCM Cores from the drop-down menu.

List Cores Page

The List Cores page lists all NCM Cores in the NCM Mesh. This page provides information to properly manage the Distributed System.

To open the List Cores page, on the menu bar under Admin select Distributed and click Core List. The List Cores page opens.

List Cores Page Fields

Field	Description
Name	The NCM Core's name.
Core Hostname	The hostname of the NCM Core's NCM server.
Is Master?	Is the NCM Core the Master Definition? (Yes or No)
Timezone Offset	The timezone offset of the actual NCM Core server.
Status	Currently, there is only Normal status.
Realm	The default Realm for the NCM Core.
Actions	You can select the following option: • Edit — Open the Edit Core page. Refer to "Edit Core Page" on page 63.

Edit Core Page

The Edit Core page enables you to edit the NCM Core definition.

To open the Edit Core page:

1. On the menu bar under Admin select Distributed and click Core List. The List Cores page opens.

2. In the Actions column, click the Edit option. The Edit Core page opens.

You can complete the following fields:

- Name — Enter the NCM Core name.

- Database Identifier — Enter the Database Identifier. This is the name of the NCM Core's database as it appears in the *tnsnames.ora* file. The Database Identifier is corresponds to Oracle's SID (System Identifier). This information is needed to make connections to a particular Oracle instance on a server. Multiple instances may be running on any given server, but each will have different SIDs.

- Core Hostname - Enter the hostname of this NCM Core's server.

- RMI Port — Enter the RMI port. RMI (Remote Method Invocation) is Java's remote procedure call protocol. The distributed system makes RMI calls between NCM servers in the NCM Mesh to transfer information about scheduled tasks, system settings, software images, and so on.

- Database hostname — Enter the Database hostname.

- Database Port — Enter the port on the database server with which NCM communicates with the database.

- Timezone Offset — Select a Timezone offset from the drop-down menu.

- Replication Admin User — Enter the name of the Replication Admin user. The Replication Admin user is created and used by the Oracle database to manage replication.

- Replication Password — Enter the Replication Admin user's password.

- Confirm Replication Password — Re-enter the Replication Admin user's password.

- Comments — Add any comments about the NCM Core.

- Realm Name — Enter the Realm in which the NCM Core resides. For information on segmenting devices, refer to the *User Guide for Network Compliance Manager 1.4*.

Device Password Rule Priority Reset Page

The Device Password Rule Priority Reset page enables you to reset device password rule priorities in the event that a uniqueness constraint conflict occurs for those objects.

To open the Reset Password Priority page, on the menu bar under Admin select Distributed and click Device Password Rule Priority Reset. The Device Password Rule Priority Reset page opens.

Click the Reset Priority button to reset the device password rule priorities.

Renew Configuration Options Page

The Renew Configuration Options page enables you to reset the configuration options when the configuration options on a NCM Core become out-of-sync with other servers in the NCM Mesh.

To open the Renew Configuration Options page, on the menu bar under Admin select Distributed and click Renew Configuration Options. The Renew Configurations Options page opens.

Click the Renew Config Options button to ensure that all options on the NCM Core are in sync with the rest of the NCM Mesh.

Chapter 4: Failover and Recovery

Use the following table to quickly locate information in this chapter.

Topic	Refer to:
Failover	"Failover" on page 65
Recovery	"Recovery" on page 65

Failover

When the network has been configured to failover, if an NCM Core fails, users can continue to access all data in the system using a different NCM Core. All Sites that were originally managed by the failed NCM Core can be pointed to a new NCM Core using the Site Reassignment page. Refer to "Site Reassignment Page" on page 62 for information. Procedures for system recovery will vary depending on how the remote server failed.

Recovery

There are three basic recovery scenarios:

- Loss of network connectivity
- Loss of an NCM server
- Loss of a Database server

Loss of Network Connectivity

In the case of lost network connectivity, failover occurred due to network issues. No problems occurred with the NCM server or with the Oracle database server. Recovery consists of the following steps:

1. Resolve the network issues.

2. Reset Sites that had been reassigned back to their original NCM Core. This can be accomplished in NCM. Refer to "Site Reassignment Page" on page 62.

3. If any drivers have been added to the system during the outage, click the "Reload Drivers" button on the Start/Stop Services page. This action reloads the driver files and pushes them across to other NCM Cores in the NCM Mesh. This action should be performed on the NCM server where the drivers were added. Refer to the *User Guide for Network Compliance Manager 1.4* for information.

4. If any system settings have been modified during the outage, use the "Renew Config Options" page to make sure options are synced across the NCM Mesh. Refer to "Renew Configuration Options Page" on page 64.

5. If any NCM Cores have lost connectivity for a long period of time, restart the NCM Core server that lost connectivity after data sync so as to reload certain Site data and avoid exception errors due to obsolete data.

Once the network issues are resolved, the system should recover as replication syncs data between the databases.

Loss of an NCM Server

In cases where the NCM server suffers a failure that requires re-installation of the server, recovery consists of the following steps:

1. During NCM installation, select the "Use existing database" option. In addition, the database selected should be the one the failed server was previously using.

2. Add the *distributed.rcx* file from the *ReplicationScriptToolBundle* to the directory where the *appserver.rcx* file resides.

3. Re-add any NCM Core-specific special case options for patches and support issues.

4. Restart NCM.

5. Reset Sites that had been re-assigned back to their original NCM Core. This can be accomplished in NCM. Refer to "Site Reassignment Page" on page 62.

6. If any drivers have been added to the system during the outage, click the "Reload Drivers" button on the Start/Stop Services page. This action reloads the driver files and pushes them across to other NCM Cores in the NCM Mesh. This action should be performed on the NCM server where the drivers were added. Refer to the *User Guide for Network Compliance Manager 1.4* for information.

7. If any system settings have been modified during the outage, use the "Renew Config Options" page to make sure options are synced across the NCM Mesh. Refer to "Renew Configuration Options Page" on page 64.

8. Edit the original NCM Core to modify any parameters that may be different (perhaps the installation happened on a new server with a different hostname).

9. Copy the software images repository from a good NCM Core to the recovered NCM Core.

Loss of a Database Server

In the case of a lost database server, the NCM server is still running but cannot access the database. The database will need to be rebuilt and replication setup again on the database. Recovery consists of the following steps:

1. Pause or delete any tasks that appear to be stuck waiting or pending and not running because they are associated with the lost server. You can perform this action on another NCM server in the NCM Mesh.

2. If the server that was lost is the Master Definition server for replication, the procedure described for relocating the Master Definition server should performed. Refer to "Changing the Master Definition Site" on page 37.

3. Refer to "Adding a New NCM Core to an Existing NCM Mesh" on page 29 for instructions on adding a new NCM Core.

4. Reset Sites that had been reassigned back to their original NCM Core. This can be accomplished in NCM. Refer to "Site Reassignment Page" on page 62.

5. In NCM, edit the NCM Core that was recovered to make sure all information is correct for the new setup. Refer to the *User Guide for Network Compliance Manager 1.4* for information.

Chapter 5: Troubleshooting

Use the following table to quickly locate information in this chapter.

Topic	Refer to:
Oracle Replication Setup	"Oracle Replication Setup" on page 69
Replication Tools and Commands	"Replication Tools and Commands" on page 70
Common Errors	"Common Errors" on page 71

Oracle Replication Setup

Oracle's replication setup process is complicated. The following guidelines will help prevent problems with replication setup.

- Ensure that the *tnsnames.ora* file is setup correctly on each system. You can verify correct setup by running SQLPlus on each system to make certain that you can connect to other databases. For example, if you have two systems (A and B), run SQLPlus on A to connect to the database on B using the following command:
  `sqlplus USER/PWD@DATABASE_B`.

- It may be necessary to remove or comment out the `NAMES.DEFAULT_DOMAIN` property entry from the *sqlnet.ora* file for the appropriate connections to work. Consult with your Oracle DBA if you have questions.

- If the replication setup process fails at any step, it is recommended that you follow the process for removing replication from the entire NCM Mesh (refer to "Removing Replication Across the Entire NCM Mesh" on page 38) and then follow the replication setup sets again (refer to "System Setup" on page 23).

Replication Tools and Commands

The following tools and commands can help to isolate problems with replication setup.

To ensure that database links are working:

- Using SQLPlus, enter: `SELECT CURRENT_TIMESTAMP FROM DUAL@<OTHER_SID>;`

- Using SQLPlus, enter: `SELECT DBLINK FROM DBA_REPSITES;`

- From a command line, enter: `tnsping <OTHER_SID>`

- From a command line, enter: `ping <HOST>`

To ensure the replication group is working:

- Using SQLPlus, enter: `SELECT GNAME,STATUS FROM DBA_REPGROUP;`

To ensure there are no errors for replication admin requests:

- Using SQLPlus, enter: `SELECT STATUS,REQUEST,ONAME FROM DBA_REPCATLOG;`

Common Errors

Some errors can occur during replication setup. Ones that have been reported are noted below, with suggested corrective action. In some cases, it might be necessary to remove replication across the NCM Mesh and re-setup replication. Refer to for information.

Error

```
SQL> EXEC DBMS_REPCAT.RESUME_MASTER_ACTIVITY(gname=>'
<REPGROUP_NAME>');BEGIN DBMS_REPCAT.RESUME_MASTER_ACTIVITY
(gname=>'<REPGROUP_NAME>'); END;

*

ERROR at line 1:

ORA-23419: regenerate replication support before resuming master
activity

...
```

Corrective Action

An internal Oracle error has prevented replication support from being generated correctly for one or more objects. To correct this, it will be necessary to remove and re-setup replication for the NCM Mesh.

Error

```
ORA-23374: object group "PUBLIC"."REPLNCM" already exists
ORA-06512: at "SYS.DBMS_SYS_ERROR", line 95
ORA-06512: at "SYS.DBMS_REPCAT_MAS", line 798
ORA-06512: at "SYS.DBMS_REPCAT", line 548
```

Corrective Action

An internal Oracle error has prevented replication support from being generated correctly for one or more objects. To correct this, it will be necessary to remove and re-setup replication for the NCM Mesh. Refer to for information.

Error

```
BEGIN DBMS_REPCAT.DROP_MASTER_REPOBJECT('SYSTEM',
'RN_DEVICE_VIEW','TABLE',FALSE)
; END;

*

ERROR at line 1:

ORA-00060: deadlock detected while waiting for resource

...
```

Corrective Action

Rerun the deletion script you were running when this occurred. You may see other errors due to certain actions already having been done, but that is expected.

Error

```
BEGIN DBMS_REPCAT.DROP_MASTER_REPOBJECT('SYSTEM','
RN_DEVICE_VIEW','TABLE',FALSE)
; END;

*

ERROR at line 1:

ORA-23308: object SYSTEM.RN_DEVICE_VIEW does not exist or is
invalid

...
```

Corrective Action

Rerun the deletion script you were running when this occurred. You may see other errors due to certain actions already having been done, but that is expected.

Appendix A: Restoring NCM Replication After Upgrading to Oracle 10g

This appendix includes the steps for upgrading from Oracle 9i (9.2.0.1) to Oracle 10g (10.2.0.4) and how to restore NCM replication after the upgrade.

Getting Started

Before upgrading from Oracle 9i (9.2.0.1) to Oracle 10g (10.2.0.4), do the following:

1. Make sure that a current NCM 1.3 Oracle 9i (9.2.0.1) database is running on all Masters.

2. Make sure that the *ReplicationScriptTool bundle* files are installed on a Java-capable system.

3. Obtain the following information:

   – Login/password/SID for NCM into all databases in the NCM Mesh.

   – The replication password for the existing 'repadmin' user.

   – DatabaseIdentifier (SID), NCM server hostname, NCM server RMI listening port, database hostname, and database listening port for all Masters.

   – The timezone offset (integer from UTC) for the entire NCM Mesh. This must be a constant across the NCM Mesh. Refer to for information.

   – The existing replication group name.

Removing Replication Before Upgrading the Database

To remove replication across the entire NCM Mesh, do the following:

1. Modify all NCM Sites to point to a NCM Core that is the Master Definition Site.

2. Shut down all NCM management engines so as to remove replication across the entire NCM Mesh.

3. Update the variables for all Masters in the *ReplicationScriptTool.properties* file. Ensure that you modify all "REPLACEME" text entries in the *ReplicationScriptTool.properties* file with the appropriate values for your environment.

   Note: The Master Definition Site must be the first entry. In addition, be sure to update the replication user, password, and the replication group name as needed. Also, make sure the "mode" property is set to "delete".

4. Run the ReplicationScriptTool using the following command:
   `java -classpath . ReplicationScriptTool.`

5. Make sure that you have a server and directory (typically on one of the database systems) where you can run a copy of SQLPlus that can access all NCM Cores in the NCM Mesh. To verify this, connect to a database with SQLPlus and issue the `CONNECT <USER>/<PASSWORD>@<SID>` command for each unique SID in the NCM Mesh. This is necessary because the script issues `CONNECT` commands to change which database it is running procedures against.

6. Copy the newly generated SQLPlus file (typically named *ReplicationScript.sql,* depending on what value was set in the *ReplicationScriptTool.properties* file), from the *ReplicationScriptTool* working directory to the server and directory where you plan to run SQLPlus.

   Note: Make sure any files named "*_spool.txt" do not exist in the directory. When run, the script will log its actions to a file with that suffix.

7. Ensure that you have two SQLPlus windows open before running the script: One SQLPlus window connected as the NCM user and one connected as the SYSTEM user.

8. Run the replication script copied in Step 6 above in the SQLPlus window connected as the NCM user using `@ReplicationScript.sql`.

9. Follow the instructions provided by the script at several points as it runs. This includes running other commands in the second SQLPlus window to verify that certain operations necessary for later steps in the script have completed.

10. Exit SQLPlus once the script completes. This ensures that all actions are logged to the Spool file.

11. Verify that the replication has been removed correctly. Make sure:

    –   The Spool file generated by running the replication SQLPlus script contains no errors.

    –   Verify (using Oracle's Enterprise Manager) that each database in the NCM Mesh does not have replication setup. If needed, consult your Oracle DBA for assistance.

12. Remove the *distributed.rcx* file from each NCM server.

    Note: Only the Master Definition Site database is used for restoring replication.

13. Restart the NCM servers that point to the Master Definition Site. Make sure all of the NCM Core changes are complete. Check to see that the devices are associated with NCM Sites that map to the NCM Core, as needed.

14. Once again, shut down the NCM management engines that point to the Master Definition Site.

15. Upgrade the Oracle database from 9i (9.2.0.1) to 10g (10.2.0.4), including the database for the Master Definition Site. If needed, consult your Oracle DBA for assistance.

Restoring NCM Replication After Upgrading to Oracle 10g

To restore NCM replication, do the following:

1. Dump the Master Definition Site database using the following command:
   `EXP <USER>/<PASSWORD>@<MASTER1_SID> TABLES=RN_% FILE=<filename>`
   where `<USER>`, `<PASSWORD>`, and `<MASTER1_SID>` were collected in Step 3 in the "Getting Started" section. Refer to "Getting Started" on page 73 for information. (Refer to for information on "Appendix C: Data Pump Utility" on page 81 for information on the Data Pump Utility.)

   Note: Make sure you log the output of this command to a file for examination later if needed.

2. Set up replication again with the existing data from Step 1 above. Refer to "Initial Replication Setup" on page 19 for instruction on how to set up initial replication.

   Note: Create a NCM 1.3 database on Oracle 10g (10.2.0.4) so as to have an up-to-date NCM 1.3 database for replication. This is an added step in the "Initial Replication Setup" on page 19. You could receive an error message regarding the *RN_CORE* table during initial replication setup. You can ignore this error message. However, you will need to manually update the *RN_CORE* table after restoring replication.

   Use the following command to create a NCM 1.3 database on Oracle 10g (10.2.0.4):

   `IMP <USER>/<PASSWORD>@<MASTER2_SID> TABLES=RN_% FILE=<filename>`
   where `<USER>`, `<PASSWORD>`, and `<MASTER2_SID>` were collected in the initial replication setup.

3. After finishing initial replication setup, update the RN_CORE table on one of the NCM Cores to make sure it includes the correct information in all columns regarding the new replication system. If needed, consult your Oracle DBA for assistance.

Appendix B: Extracting Data from the NCM Database

This appendix includes the steps for extracting information about the High Availability Mesh from the NCM database.

Note: This information is also available on the appropriate Edit Core page. Refer to for information.

To extract this data from the NCM database, run the following SQL queries against the NCM database. The query will return multiple rows, one row for each database core.

For the NCM application server hostname, run the following:

```
SELECT CoreHostname FROM RN_CORE;
```

For the NCM application server RMI listening port, run the following:

```
SELECT CoreRMIPort FROM RN_CORE;
```

For the Database Identifier (SID), run the following:

```
SELECT DatabaseIdentifier FROM RN_CORE;
```

For the NCM database server hostname, run the following:

```
SELECT DatabaseHostname FROM RN_CORE;
```

For the NCM database server listening port, run the following:

```
SELECT DatabasePort FROM RN_CORE;
```

For the Timezone offset, run the following:

```
SELECT TimezoneOffset FROM RN_CORE;
```

For the Replication User Name, run the following:

```
SELECT ReplicationAdminUser FROM RN_CORE;
```

For example, in a two NCM Core High Availability Mesh, run the following queries:

```
SELECT CoreHostname FROM RN_CORE;
SELECT CoreRMIPort FROM RN_CORE;
SELECT DatabaseIdentifier FROM RN_CORE;
SELECT DatabaseHostname FROM RN_CORE;
```

```
SELECT DatabasePort FROM RN_CORE;
SELECT TimezoneOffset FROM RN_CORE;
SELECT ReplicationAdminUser FROM RN_CORE;
```

You will receive the following:

```
SQL> SELECT CoreHostname FROM RN_CORE;
COREHOSTNAME
----------------------------
10.255.132.177
10.255.136.203
SQL> SELECT CoreRMIPort FROM RN_CORE;
CORERMIPORT
----------------------------
1099
1099
SQL> SELECT DatabaseIdentifier FROM RN_CORE;
DATABASEIDENTIFIER
----------------------------
core1
core2
SQL> SELECT DatabaseHostname FROM RN_CORE;
DATABASEHOSTNAME
----------------------------
10.255.55.30
10.255.56.30
SQL> SELECT DatabasePort FROM RN_CORE;
DATABASEPORT
----------------------------
1521
1521
SQL> SELECT TimezoneOffset FROM RN_CORE;
TIMEZONEOFFSET
----------------------------
-8
-8
SQL> SELECT ReplicationAdminUser FROM RN_CORE;
REPLICATIONADMINUSER
----------------------------
repadmin
repadmin
```

From this data, you can start filling out the *ReplicationScriptTool.properties* file as follows:

```
# This .0 server will be the master definition site, and is the default
# initial core installed.
db.server.0=10.255.55.30
db.port.0=1521
db.name.0=core1
db.username.0=admin
db.password.0=rendition
app.server.0=10.255.132.177
app.rmiport.0=1099
app.corename.0=Core 1

# Second Master site
db.server.1=10.255.56.30
db.port.1=1521
db.name.1=core2
db.username.1=admin
db.password.1=rendition
app.server.1=10.255.136.203
app.rmiport.1=1099
app.corename.1=Core 2

# From UTC
timezone_offset=-8
replication_user=repadmin
replication_password=<YOU WILL NEED TO GET THIS FROM YOUR DBA>
replication_group_name=<YOU WILL NEED TO GET THIS FROM YOUR DBA>
```

Appendix C: Data Pump Utility

For Oracle 10g databases, Oracle supports the Data Pump Utility. You can use the Data Pump Utility to transfer data from one database to another without staging the data on the disk. This improves performance dramatically over the Export/Import Utility.

To transfer data between two NCM Cores and two Oracle 10g databases:

1. Edit the *tnsnames.ora* files for two databases in two NCM Cores to include database information to the other NCM Core. Run the following commands:

   ```
   sqlplus SYSTEM/<SYS_PASSWORD>@<MASTER2_SID>

   CREATE DIRECTORY <DIR_NAME> AS '<DIR_PATH>';
   GRANT ALL ON DIRECTORY <DIR_NAME> TO PUBLIC;
   CREATE DATABASE LINK <DBLINK> CONNECT TO <USER>
   IDENTIFIED BY <PASSWORD> USING '<CONNECTSTRING>';
   ```

   Where:

   - `<DIR_NAME>` and `<DIR_PATH>` is a directory for logs.
   - `<USER>` is the user NCM uses to connect the database.
   - `<MASTER2_SID>` is the SID of the destination NCM Core.
   - `<SYS_PASSWORD>` and `<PASSWORD>` are the appropriate passwords.

   Note: The connect string can be found with Oracle Net Manager.

2. Run the Data Pump Utility on the destination database where you want to import data.

   ```
   $ IMPDP <USER>/<PASSWORD>@<MASTER2_SID> TABLES='<USER>.RN_%'
   DIRECTORY=<DIR_NAME> NETWORK_LINK=<DBLINK>
   ```

   Note: `<DBLINK>` is the name of the remote database link to the source system that you created with `CREATE DATABASE LINK` in Step 1.

Be sure to check the logging information under the directory `<DIR_NAME>` for any Data Pump Utility error messages.

3. Return to Chapter 2 ("Chapter 2: Installation, Setup, and Upgrading" on page 15) to complete the setup.

Index