High Availability Distributed System Configuration Guide on Microsoft SQL Server for Network Compliance Manager 1.3

CiscoWorks

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

Text Part Number: OL-14992-01

High Availability Distributed System Configuration Guide on Microsoft SQL Server for Network Compliance Manager 1.3

Table of Contents

Preface

This document contains information on installing, configuring, and administering the CiscoWorks Network Compliance Manager (NCM) 1.3 High Availability Distributed System on Microsoft SQL Server 2005, Service Pack 2.

Note: The NCM High Availability Distributed System on SQL Server software requires SQL Server 2005, Service Pack 2. Keep in mind that the NCM High Availability Distributed System on SQL Server software can only be run on two NCM Cores and no more than 6,500 devices can be managed.

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.
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.
Enter	Indicates that you should type the text or command that follows, then press the Enter key on the keyboard.
< >	Indicates variable information, such as a name or folder that you must supply. Do not include the angle brackets when replacing the placeholder.

Accessing NCM Documentation

All documentation, including this document and any or all of the parts of the NCM documentation set, might be upgraded over time. Therefore, we recommend you access the NCM documentation set using the Cisco.com URL:

http://www.cisco.com/en/US/products/ps6923 tsd_products_support_series_ho me.html

The Docs tab visible from within Network Compliance Manager might not include links to the latest documents.

To access user documentation:

- *User Guide for Network Compliance Manager 1.3* — To view the PDF version, after logging in, on the menu bar click Docs. The CiscoWorks Network Compliance Manager Documentation page opens.
- Online Help Files — To view Online Help files, after logging in, click the Help icon at the top of any NCM page.
- *Device Drive Reference for Network Compliance Manager 1.3* — To view the PDF version, after logging in, on the menu bar click Docs. The CiscoWorks Network Compliance Manager Documentation page opens.
- Integration API — To view the PDF version of *Java*, *PERL,* and *SOAP API Reference Guides for Network Compliance Manager 1.3*, after logging in, on the menu bar click Docs. The CiscoWorks Network Compliance Manager Documentation page opens.
- CLI Help — To view the command line Help on the server computer, click Start → Programs → CWNCM → CWNCM Client and login. There are two ways to view Help for CLI commands. Enter: `help` to see a list of all commands. Enter `help <command name>` to see detailed help on a specific command.

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 Sites.
- **Site —** A set of devices with unique IP addresses. A Site is managed by one (and only one) NCM Core. Multiple Sites can be managed by a single NCM Core. Refer to the *User Guide for Network Compliance Manager 1.3* for information on segmenting devices.
- **NCM Mesh** — Multiple NCM Cores connected via replication.
- **Publisher** — A SQL Server database that defines what data is replicated and handles transferring changed data to and from the other databases in the NCM Mesh.
- **Subscribers** — Databases in the NCM Mesh that receive replicated data.

Note: The NCM High Availability Distributed System on SQL Server software requires SQL Server 2005, Service Pack 2.

Overview

The NCM High Availability Distributed System on SQL Server is a 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 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 functionality is included in a NCM High Availability Distributed System:

- The concept of 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.
- Replication on SQL Server 2005:
  - Conflict resolution in the event data modification occurs on two different NCM Cores. This conflict is typically resolved using the latest timestamp method.
  - Replication monitoring and conflict notification is built into NCM. You can manage SQL Server 2005 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 SQL Server 2005 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 1.3* for information on segmenting devices.

Chapter 2: Installation, Setup, and Upgrading

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

Installation Planning

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

- Device partitioning planning across NCM Cores
- Network configuration planning for connectivity between NCM servers and SQL Servers in the NCM Mesh
- 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. The different 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.
- SQL 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 replication setup, a snapshot of the initial database is transferred to each database in the NCM Mesh. This requires ample time, disk space, and bandwidth.

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 *N*2*.

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 outage 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 a SQL Server 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 only supports two NCM Cores in a High Availability Distributed System on SQL Server and can support no more than 6,500 nodes.
- NCM currently does not support joining multiple 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 down-time 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 SQLServerReplicationScript.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 the SQLServerReplicationScript.sql script.

- To successfully recover from the loss of the Publisher server, there are four databases on that Publisher server that must be restored to a mutually consistent state. As a result, you must backup these databases. In doing so, take the necessary steps to ensure they can be restored to a mutually consistent state. The four databases include:
  - NCM database
  - `master' database
  - 'msdb' database
  - Distribution database ('NCMdistdb' by default)

  Refer to "Loss of a Database Server" on page 44 for information on restoring databases.

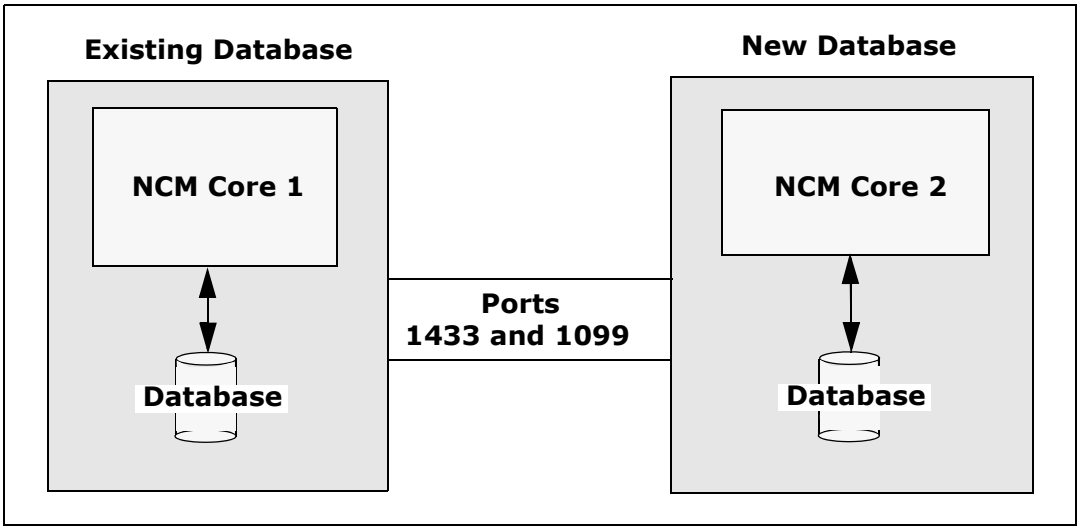
Installation and Removal

This section includes information on:

- Creating a two NCM Core SQL Server replication environment
- Adding an additional NCM Core to the NCM Mesh
- Deleting a Subscriber from a NCM Mesh
- Removing replication

Note: Refer to "Chapter 5: Troubleshooting" on page 49 for information on Troubleshooting the installation.

The following figure provides an overview of the replication process.



Creating a Two NCM Core SQL Server Replication Environment

When creating a two NCM Core SQL Server replication environment, NCM Core 1 is the Publisher and NCM Core 2 is the Subscriber.

Preparation Requirements

You must have the following items configured before creating a SQL Server replication environment:

- Both SQL Server High Availability Distriubted Systems must be at SQL Server 2005, Service Pack 2.
- A current NCM 1.3 database on NCM Core 1.
- A NCM server connected to the database on NCM Core 1.
- The SQL Server agent service running on NCM Core 1 (Publisher) database server.
- An empty database (with no data or NCM schema setup) on NCM Core 2.
- A login account with permission to read and modify the tables in the empty database on NCM Core 2.
- A network connection from the NCM Core 1 servers to the NCM Core 2 servers (and vice-versa) that enable ports 1433 and 1099 (or appropriate variations) to be connected between the servers.
- sqlcmd access to NCM Core 1 and NCM Core 2. (**Note**: You will need to supply credentials for a login that is a member of the sysadmin role when running the scripts.)
- The SQLServerReplicationScriptTool application installed on a Java-capable system.

Installation Steps

Do the following to create a SQL Server replication environment:

1. Set up a shared directory that is accessible from both NCM Core 1 and NCM Core 2.
2. Collect the following information:

- Login name and password of a SQL Server login that is a member of sysadmin on NCM Core 1 and NCM Core 2.
- Login name and password of a Windows account under which SQLServer agents can run.
- Database name, NCM server hostname, NCM server RMI listening port, database hostname, and database listening port for NCM Core 1 and NCM Core 2.
- The time zone 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.
- UNC path for the shared directory you setup in Installation Step 1.

3. Power down the NCM server that is accessing the database on NCM Core 1.
4. Update the variables for NCM Core 1 and NCM Core 2 in the *SQLServerReplicationScriptTool.properties* file. In addition, update the timezone offset in that file. Make sure the "mode" property is set to "initial". These properties are described in detail in the file. Set *replication.data.dir* to be the directory you set up in Installation Step 1.
5. Run the script tool. This will output two files. Enter: **Java -classpath . SQLServerReplicationScriptTool**
6. Copy the second output file to the shared directory that is accessible from both NCM Core 1 and NCM Core 2. The file defaults to *SQLServerPreSnapshotScript.sql*. The name is specified by the "pre-snapshot.file" property in the *SQLServerReplicationScriptTool.properties* file.
7. Run the first output file using **sqlcmd** with a login that is a member of the sysadmin role. The name is specified by the "script.file" property in the *SQLServerReplicationScriptTool.properties* file.
8. Using **sqlcmd**, execute the following t-sql query on both the NCM Core 1 database and the NCM Core 2 database. **Select count(*) from INFORMATION_SCHEMA.TABLES, where TABLE_TYPE = 'BASE TABLE' and TABLE_NAME like 'RN_%'; GO** (Note: Continue executing this query until the result is the same for both the NCM Core 1 and NCM Core 2 databases.)

Deleting a Subscriber from a NCM Mesh

When deleting a subscriber from a NCM Mesh, NCM Core 1 is the Publisher and NCM Core 2 is the Subscriber you want to remove.

System Requirements

You must have the following items configured before deleting a Subscriber from a NCM Mesh:

- sqlcmd access to NCM Core 1 and NCM Core 2.
- The *SQLServerReplicationScriptTool* application installed on a Java-capable system

Installation Steps

To delete a Subscriber from a NCM Mesh:

1. Collect the following information:
   - Login name and password of a SQL Server login that is a member of sysadmin on NCM Core 1 and NCM Core 2.
   - Database name, NCM server hostname, NCM server RMI listening port, database hostname, and database listening port for NCM Core 1 and NCM Core 2.
2. Ensure that all devices in NCM belong to sites on NCM Cores that are not going to be removed.
3. Modify all sites to point to a NCM Core that is not being removed. Alternatively, remove those sites.
4. Power down the NCM server from the NCM Core that is being removed.
5. Delete the RN_CORE entry that was removed. For example, enter: **DELETE FROM RN_CORE,** where **CoreID** = <*NNN*> and then **GO** using sqlcmd at NCM Core 1.
6. Update the variables for NCM Core 2 in the *SQLServerReplicationScriptTool.properties* file. Make sure that the variables for NCM Core 1 are correct. Make sure the “mode” property is set to “delete_server”. These properties are described in detail in the file.

7. Run the script tool. Enter: **java SQLServerReplicationScriptTool**
8. Run the first output file using **sqlcmd** with a login that is a member of the sysadmin role on NCM Core 1. (Refer to the first bullet in "System Requirements" on page 18.)

Removing Replication

To remove replication:

1. Remove each subscriber from the NCM Mesh by following the steps above for each one.
2. Execute the following script on the publisher:

   use [master]
   exec sp_dropdistributor @no_checks = 1
   GO

Upgrading from NCM 1.x to NCM 1.3

When upgrading a subscriber from a NCM Mesh, NCM Core 1 is the Publisher and NCM Core 2 is the Subscriber you want to update.

System Requirements

You must have the following items configured before upgrading a Subscriber from a NCM Mesh:

- sqlcmd access to NCM Core 1 and NCM Core 2.
- The *SQLServerReplicationScriptTool* application installed on a Java-capable system

Upgrading Steps

To upgrade a Subscriber from a NCM Mesh:

1. Turn off all NCM servers in the NCM Mesh.
2. Collect the following information:
   - Login name and password of a SQL Server login that is a member of sysadmin on NCM Core 1 and NCM Core 2.
   - Database name, NCM server hostname, NCM server RMI listening port, database hostname, and database listening port for NCM Core 1 and NCM Core 2.
3. Update the variables for NCM Core 2 in the *SQLServerReplicationScriptTool.properties* file. Make sure that the variables for NCM Core 1 are correct. Make sure the "mode" property is set to "upgrade_from_6_2" or "upgrade _from_6_3". These properties are described in detail in the file.
4. Run the script tool. Enter: **java SQLServerReplicationScriptTool**
5. Run the first output file using **sqlcmd** with a login that is a member of the sysadmin role on NCM Core 1. (Refer to the first bullet in "System Requirements" on page 18.)
6. Upgrade all NCM servers with the NCM Installer. Be sure to select the "Upgrade using an existing database" option.

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 Distributed 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 Distributed NCM Mesh functioning properly.

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.

Distributed System - Uniqueness Conflict

Event format:

```
rowguid: <the guid of the database row that had the conflict>
origin_datasource: <database server>.<database name>
reason_text: <a description of why the conflict occurred>
conflict_type: <type of conflict according to SQLServer>
reason_code: <error message from SQLServer, depends on the type of
conflict>
repl_create_time: <time the conflict was generated by SQLServer>

conflict_table: <where SQLServer stores the conflicting data>
dataTable: <NCM table that contains the conflicting data>
sQLServerConflictID: <ID of the conflict recorded by NCM>
status: <status>
Conflicting Data: <the columns that are conflicting>
Refer to the SQL Server 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 38.

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.

Distributed System - 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.

Distributed System - 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:

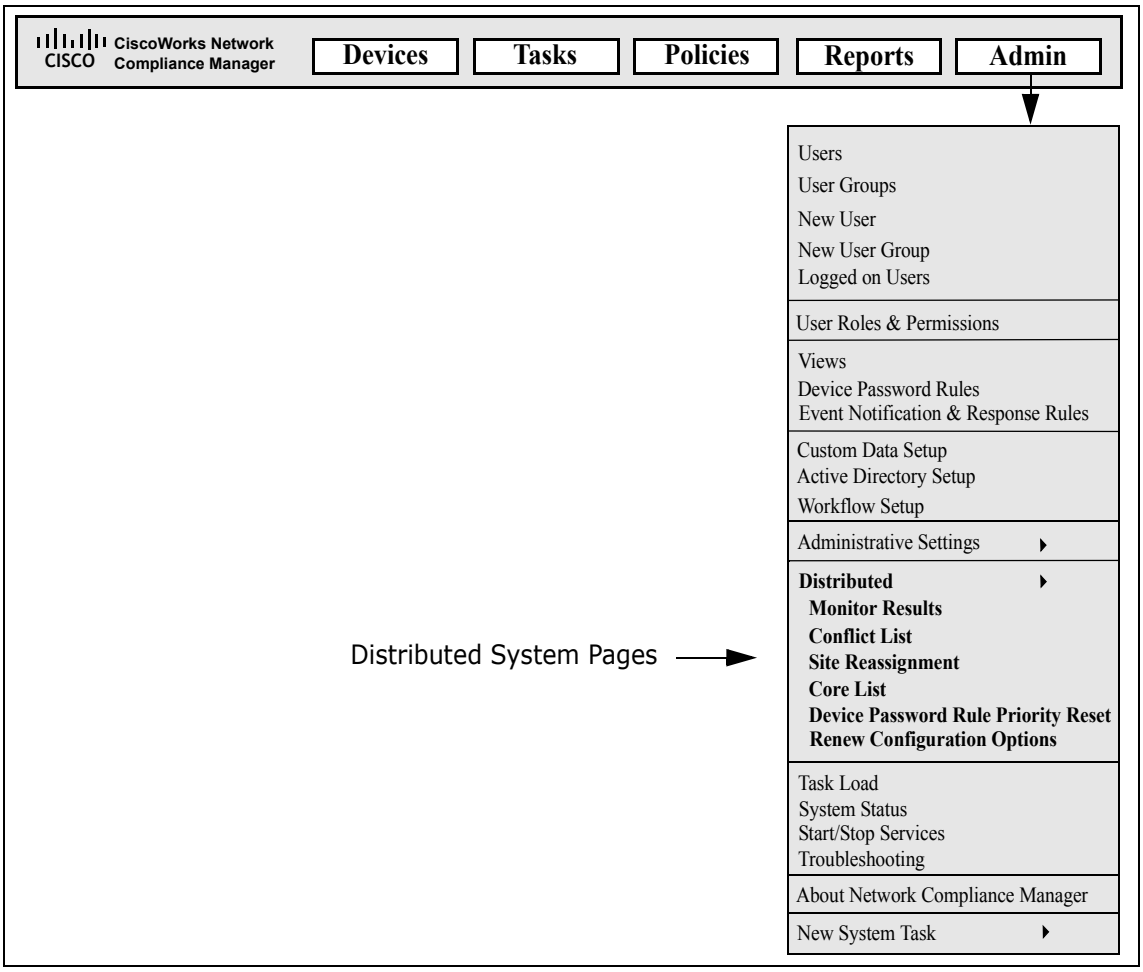
1. The host that the server cannot connect to is up and running.
2. The NCM instance on that host is running.
3. From a command line, enter `ping <host>` to ensure that network connectivity exists between servers.
4. From a command line, enter `telnet <host>` to port 1099 (or whatever your RMI listen 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.

Using the NCM Distributed System Pages

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

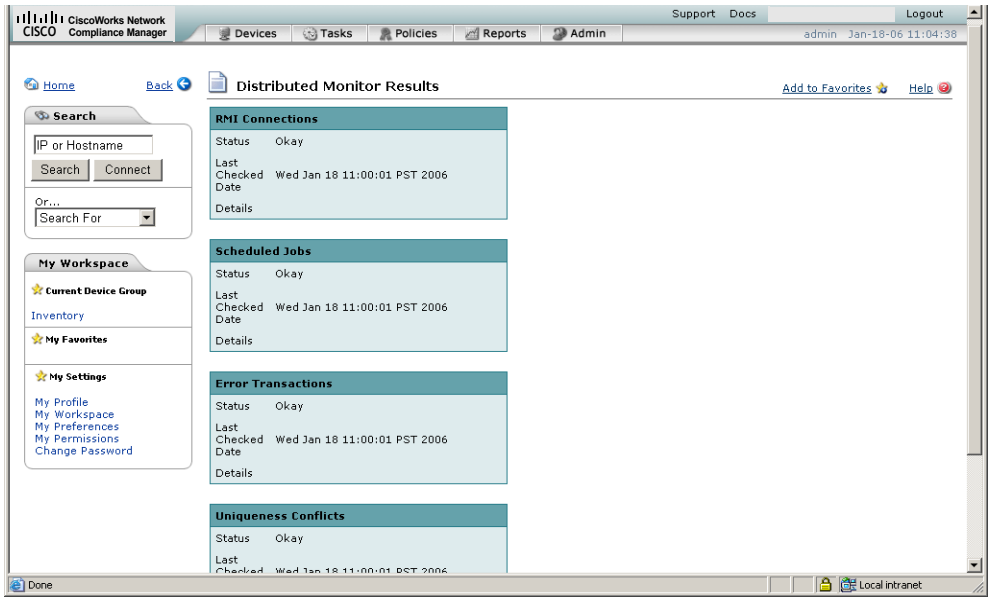
Navigating to Distributed Systems Pages



Distributed Monitor Results Page

The Distributed Monitor Results page displays the overall health of the Distributed System.

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. The following figure shows a sample Distributed Monitor Results page.



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.

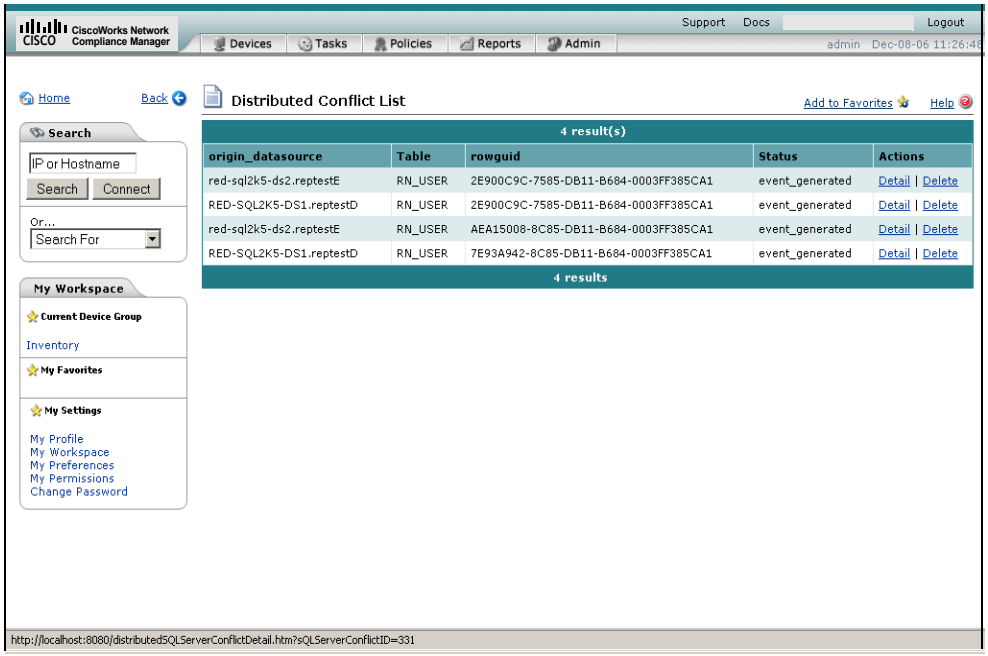
- **Uniqueness Conflicts** — Certain NCM database constraints restrict columns to unique values. In a distributed environment, these constraints can be violated when updates are made on two different NCM Cores where the unique column is set to the same value. These conditions are captured by the Replication Conflict Resolution System and logged. NCM cannot automatically resolve these conflicts. They must be resolved manually.
- **Merge Agents** — Merge Agents are the processes at the Publisher that handle transferring replicated data. NCM monitors the SQL Server jobs that schedule these processes. If for some reason the process stops, NCM will report that here. Stopped Merge Agent jobs should be restarted as soon as possible.

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. The following figure shows a sample Distributed Conflict List.

Note: Conflicts are currently viewable only at the Publisher. It is recommended that you set up an Event Rule to email you whenever a "Distributed System - Uniqueness Conflict" event is generated. Refer to the *User Guide for Network Compliance Manager 1.3* for information on creating Event Rules.



Distributed Conflict List Page Fields

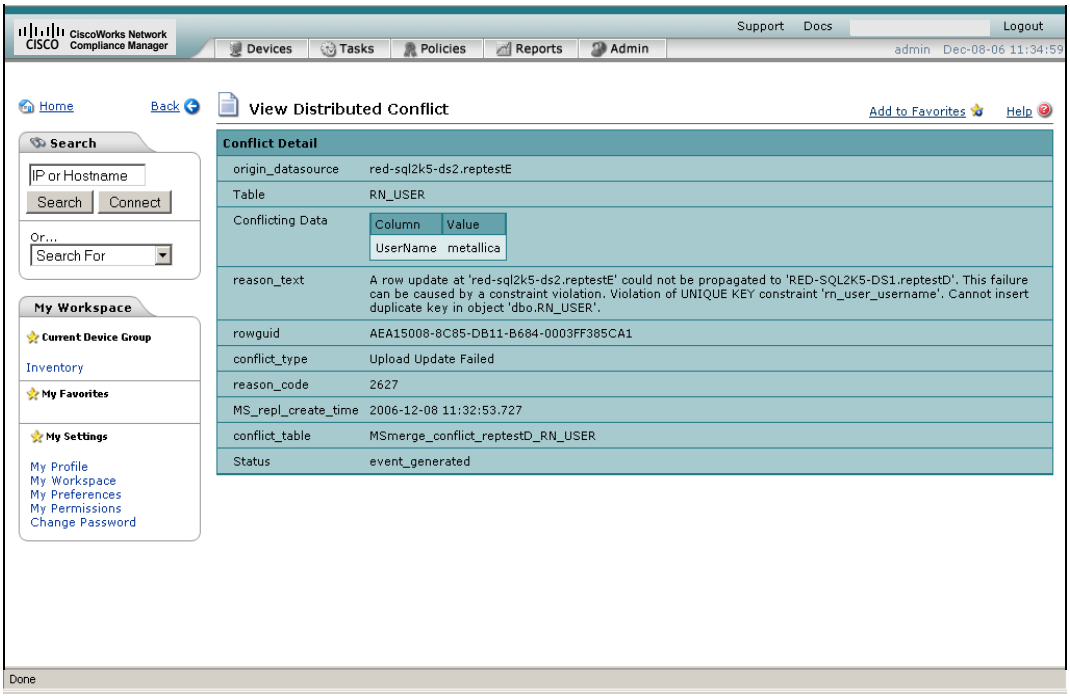
Field	Description
origin_datasource	The database on which the conflict occurred.
Table	The table on which the conflict occurred.
rowguid	The guid of the row on which the conflict occurred.
Status	Status is “event_generated” if the system has sent an alert that this conflict exists.
Actions	You can select 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 31. • 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. The following figure shows a sample View Distributed Conflict page.



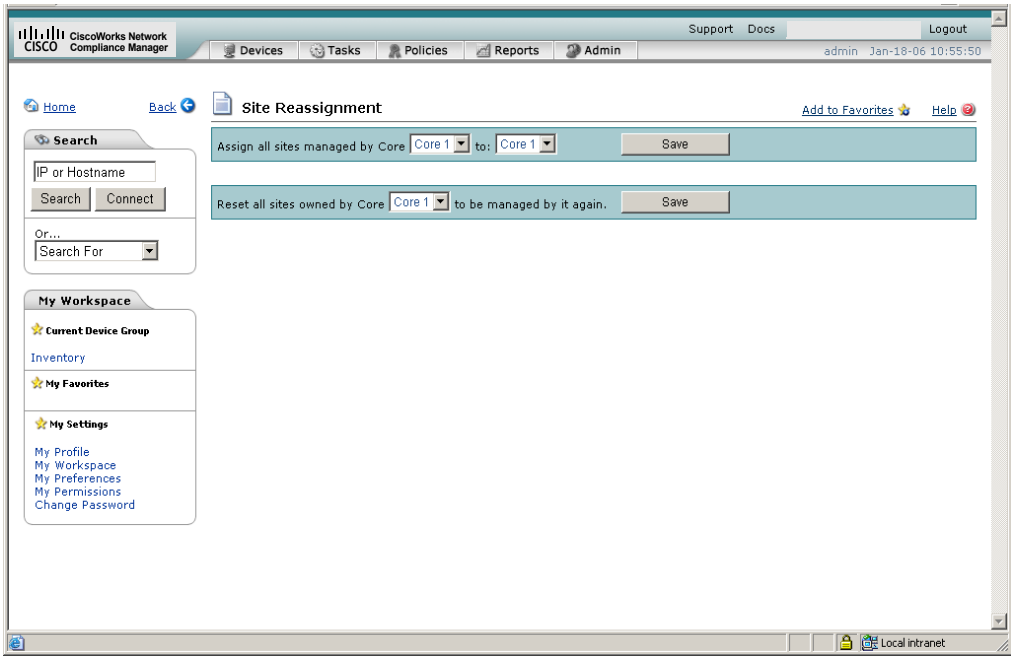
View Distributed Conflict Page Fields

Field	Description
origin_datasource	The database on which the conflict occurred.
Table	The table on which the conflict occurred.
Conflicting Data	The columns and their values that are causing the conflict.
reason_text	A description of why the conflict occurred.
rowguid	The guid of the row on which the conflict occurred.
conflict type	The type of conflict according to the SQL Server.
reason_code	Depending on the type of conflict, the error message from the SQL Server.
MS_repl_create_time	The time the conflict was generated by the SQL Server.
conflict_table	The location where the SQL Server stores the conflicting rows.
Status	Status is “event_generated” if the system has sent an alert that this conflict exists.

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. The following figure shows a sample Site Reassignment page.

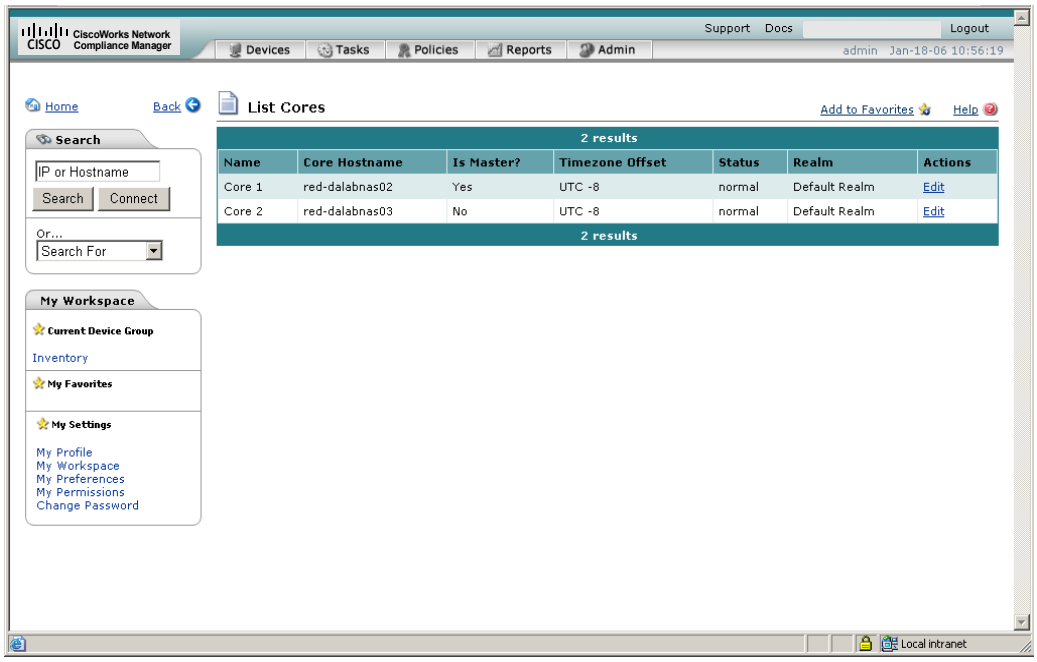


Select NCM Cores from the drop-down menu and click the Save button.

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. The following figure shows a sample List Cores page.



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.

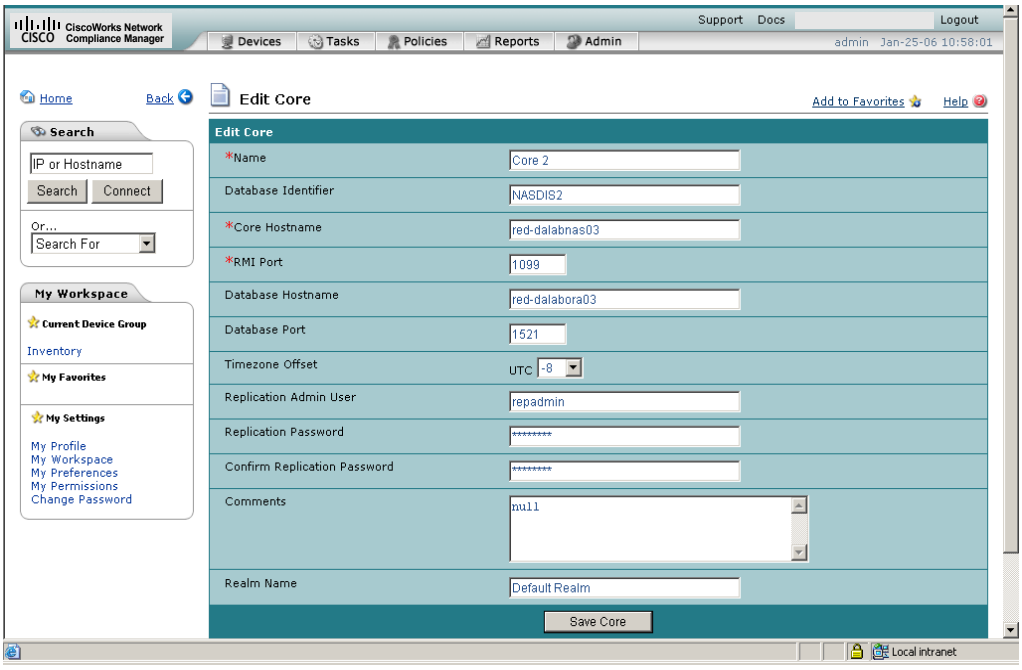
Field	Description
Realm	The default Realm for the NCM Core.
Actions	You can select the following option: • Edit — Open the Edit Core page. Refer to "Edit Core Page" on page 36.

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. The following figure shows a sample Edit Core page.



Complete the following fields and click the Save Core button.

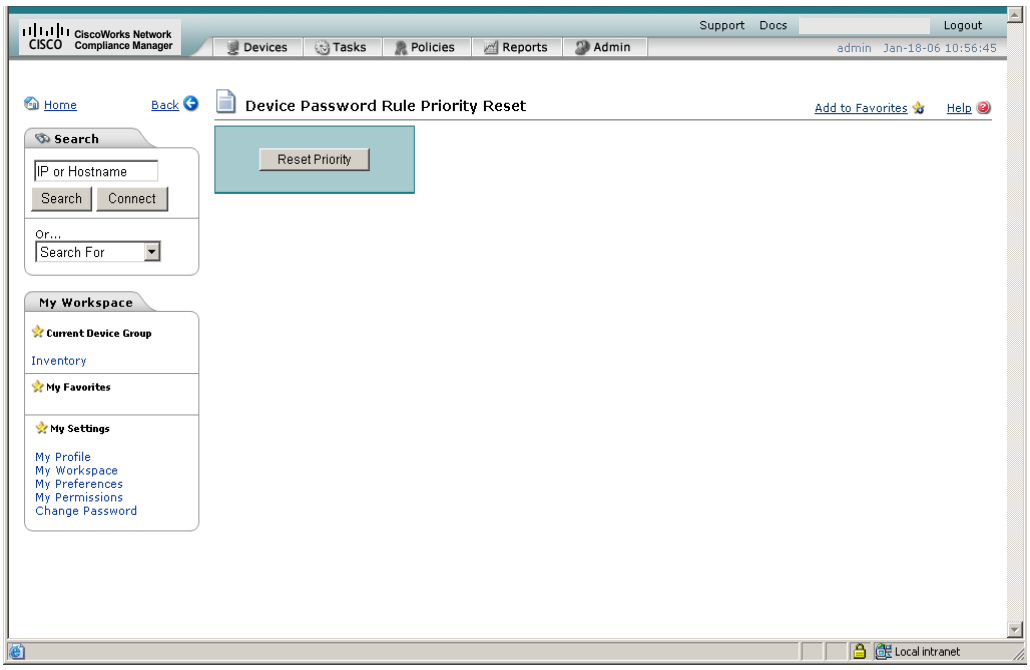
- Name — Enter the NCM Core name.
- Database Identifier — Enter the Database Identifier. This is the name of the NCM Core's database as it appears in the *tnsnames.ora* file. The Database Identifier is corresponds to SQL Server's SID (System Identifier). This information is needed to make connections to a particular SQL Server 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 SQL Server 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 1.3*.

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. The following figure shows a sample Device Password Rule Priority Reset page.

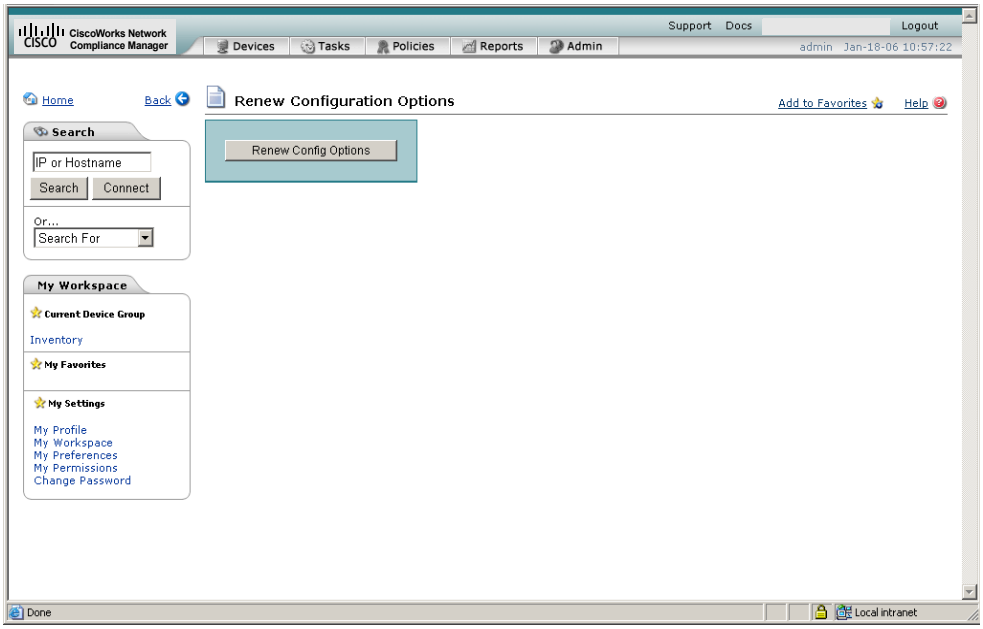


Click the Reset Priority button to reset the device password rule priorities.

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. The following figure shows a sample Renew Configuration Options page.



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.

Failover

When the network has been configured to failover, if a 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 33 for information.

Note: 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 a 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 SQL Server 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 33.
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 1.3* 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 39.
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 a NCM Server

In cases where the NCM server suffers a failure that requires re-installation of the NCM 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 33.
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 1.3* 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 39.
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 server. The database server will need to be rebuilt and replication must be reconfigured on the database server. Recovery consists of the following steps:

1. Pause or delete any tasks that appear to be waiting or pending, or not running because they are associated with the lost database server. You can perform this action on the working NCM server in the NCM Mesh.
2. If the lost database server is a subscriber, refer to "Recovering a Lost Subscriber Database Server" on page 45 for instructions on adding a new NCM Core. Skip to step 4.
3. If the lost database server is the Publisher, do the following:
   a) Pause or delete any tasks that appear to be waiting or pending, or not running because they are associated with the lost database server. You can perform this action on another NCM server in the NCM Mesh.
   b) Restore the 'master' and 'msdb' databases from backup.
   c) Restart SQL Server.
   d) Restore the Distribution database and the NCM database from backup.
4. Make sure the SQL Server Agent is running on the restored database server. Open a command prompt on the restored database server and enter: **net start**. You should see `SQL Server Agent (MSSQLSERVER)` in the output. If you do not, enter: **net start SQL Server Agent <MSSQLSERVER>**.
5. Reset all sites that have been reassigned back to their original NCM Core. Refer to "Site Reassignment Page" on page 33 for information.
6. Login to NCM and edit the NCM Core that was recovered to make sure all information is correct for the new setup. Refer to Chapter 3, "Adding Devices and Device groups," in the *User Guide for Network Compliance Manager 1.3* for information.

Recovering a Lost Subscriber Database Server

When adding a NCM Core to the NCM Mesh, NCM Core 1 is the Publisher and NCM Core 2 is the Subscriber.

System Requirements

You must have the following items configured before creating a SQL Server replication environment:

- A current NCM 1.3 database on NCM Core 1.
- A NCM server connected to the database on NCM Core 1.
- An empty database (with no data or NCM schema setup) on NCM Core 2.
- A login account with permission to read and modify the tables in the empty database on NCM Core 2.
- A network connection from all other NCM Cores the NCM Core 2 servers (and vice-versa) that enable ports 1433 and 1099 (or appropriate variations) to be connected between the servers.
- sqlcmd access to NCM Core 1 and NCM Core 2. You will need to supply credentials for a login that is a member of the sysadmin role when running the scripts.
- The SQLServerReplicationScriptTool application installed on a Java-capable system.

Installation Steps

Do the following to add an additional NCM Core to the NCM Mesh:

1. Set up a shared directory that is accessible from both NCM Core 1 and NCM Core 2.
2. Collect the following information:
   - Login name and password of a SQL Server login that is a member of sysadmin on NCM Core 1 and NCM Core 2.
   - Login name and password of a Windows account under which SQLServer agents can run.
   - Database name, NCM server hostname, NCM server RMI listening port, database hostname, and database listening port for NCM Core 1 and NCM Core 2.
   - The time zone 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. Ensure that all database servers and NCM server system are set to use the same time and time zone.
   - UNC path for the shared directory you setup in Installation Step 1.
3. Ensure that all Database servers and NCM servers are set to use the same time and timezone.
4. Power down all NCM servers in the existing NCM Mesh.
5. Update the variables for NCM Core 2 in the *SQLServerReplicationScriptTool.properties* file. Make sure that the variables for NCM Core 1 are correct. In addition, make sure the time zone offset is correct. Make sure the "mode" property is set to "add_server". These properties are described in detail in the file. Set the *replication.data.dir* to be the shared directory accessible from both NCM Core 1 and NCM Core 2.
6. Run the script tool. This will output two files. Enter: **java SQLServerReplicationScriptTool**

7. Copy the second output file to the shared directory that is accessible from both NCM Core 1 and NCM Core 2. The file defaults to *SQLServerPreSnapshotScript.sql*. The name is specified by the "pre-snapshot.file" property in the *SQLServerReplicationScriptTool.properties* file.
8. Run the first output file using **sqlcmd** with a login that is a member of the sysadmin role. The file defaults to *SQLServerReplicationScript.sql*. The name is specified by the "script.file" property in the *SQLServerReplicationScriptTool.properties* file.
9. Using **sqlcmd**, execute the following t-sql query on both the NCM Core 1 database and the NCM Core 2 database. **Select count(*) from INFORMATION_SCHEMA.TABLES, where TABLE_TYPE = 'BASE TABLE' and TABLE_NAME like 'RN_%'; GO** (Note: Continue executing this query until the result is the same for both the NCM Core 1 and NCM Core 2 databases.)
10. Install a NCM server. Be sure it points to the empty database on NCM Core 2. When prompted, have NCM connect to the database using the login from the fourth bullet on page 16. (Refer to "System Requirements" on page 45.)
11. Add the *distributed.rcx* file to each NCM server in the NCM Mesh (in the same location as *appserver.rcx*).
12. Start all of the NCM servers in the NCM Mesh.

Chapter 5: Troubleshooting

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

Topic	Refer to:
SQL Server Replication Setup	"SQL Server Replication Setup" on page 49
Removing In-Memory and Database Information	"Removing In-Memory and Database Information" on page 50

SQL Server Replication Setup

If the replication setup process fails at any step, it is recommended that you do the following:

1. Delete the RN_CORE entry that was removed:

   DELETE FROM RN_CORE

   where **CoreID** = <*ID*>;

   GO (using sqlcmd)

2. Update the variables for all masters in the *SQLServerReplicationScriptTool.properties* file.

   Note: The NCM Core being deleted must be the second NCM Core entry and the publisher must be the first entry. Update the properties fif needed. Make sure the "mode""property is set to "delete_server".

Removing In-Memory and Database Information

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.

Index

R

S

T

U

V