



## **Configuration Guide for High Availability Distributed System on Microsoft SQL Server**

### **Americas 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: 408 527-0883

Text Part Number: OL-23277-01

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks). Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

*Configuration Guide for High Availability Distributed System on Microsoft SQL Server*  
© 2010 Cisco Systems, Inc. All rights reserved.

# Table of Contents

<b>Preface .....</b>	<b>5</b>
Document Conventions .....	5
Obtaining Documentation, Obtaining Support, and Security Guidelines .....	6
<b>Chapter 1: Getting Started .....</b>	<b>7</b>
Terminology .....	7
Overview .....	8
Architecture .....	10
What is Horizontal Scalability? .....	11
NCM Internal Monitoring: SQL Server Merge Replication .....	12
NCM Internal Monitoring: Inter-NCM Core Communication .....	13
<b>Chapter 2: Installation, Setup, and Upgrading ...</b>	<b>15</b>
System Requirements.....	16
Installation Planning.....	16
Protocols, Databases, and Ports .....	19
NCM Server Installation Steps .....	20
Installation and Removal.....	20
Unpacking the Replication Script Installation Bundle .....	21
Creating a Two NCM Core SQL Server Replication Environment .....	21
Preparation Requirements .....	21
Installation Steps.....	22
Deleting a Subscriber from a NCM Mesh.....	25
System Requirements.....	25
Deletion Steps.....	25
Removing Replication .....	27
Upgrading from NCM 1.3/1.3.1 to NCM 1.4.....	27
Upgrading from NCM 1.4 to NCM 1.5 .....	30
Upgrading from NCM 1.5 to NCM 1.6 .....	33
<b>Chapter 3: System Administration .....</b>	<b>37</b>
Getting Started.....	37
NCM Generated Events .....	38
Distributed System - Uniqueness Conflict .....	38
Distributed System - Time Synchronization Warning .....	39
Distributed System - RMI Error .....	39
Distributed System - Stopped Merge Agent Job.....	40

Distributed System - Data Synchronization Delay Warning .....	40
Using the NCM Distributed System Pages .....	41
Navigating to Distributed Systems Pages .....	41
Distributed Monitor Results Page.....	42
Distributed Conflict List.....	44
View Distributed Conflict Page .....	45
Site Reassignment Page .....	46
List Cores Page.....	46
Edit Core Page.....	47
Device Password Rule Priority Reset Page.....	48
Renew Configuration Options Page .....	48
<b>Chapter 4: Failover and Recovery .....</b>	<b>49</b>
Failover .....	49
Recovery .....	49
Loss of Network Connectivity.....	50
Loss of a NCM Server .....	51
Loss of a Database Server.....	52
<b>Chapter 5: Troubleshooting .....</b>	<b>55</b>
SQL Server Replication Setup .....	55
Removing In-Memory and Database Information.....	56
<b>Index .....</b>	<b>57</b>

# Preface

This document contains information on installing, configuring, and administering the CiscoWorks Network Compliance Manager (NCM) High Availability Distributed System on Microsoft SQL Server 2005 and SQL Server 2008.

**Note:** The NCM High Availability Distributed System on Microsoft SQL Server software requires SQL Server 2005, Service Pack 2 (Standard Edition or Enterprise Edition) or SQL Server 2008 (Standard Edition or Enterprise Edition.) 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
<i>Italic</i>	Used for system messages, paths, file names, and Web URLs. For example, <i>C:\ncm\client\sdk\docs</i>
<u>Link</u>	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.

## Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, 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>

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS Version 2.0.

# Chapter 1: Getting Started

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

Topic	Refer to:
Terminology	<a href="#">"Terminology" on page 7</a>
Overview	<a href="#">"Overview" on page 8</a>

## 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 CiscoWorks Network Compliance Manager 1.6* 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 can send and receive replication data, but do not control the replication process.

**Note:** Both the Publisher and Subscribers will have the same data (subject only to replication time lags).

## Overview

The NCM High Availability Distributed System on SQL Server 2005 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.

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 Partition:
  - A device is associated with a single Partition.
  - A Partition is associated with a single NCM Core.
  - A task is associated with a specific NCM Core.
- Merge replication on SQL Server 2005:
  - 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 SQL Server 2005 replication conflicts and view merge agent 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.

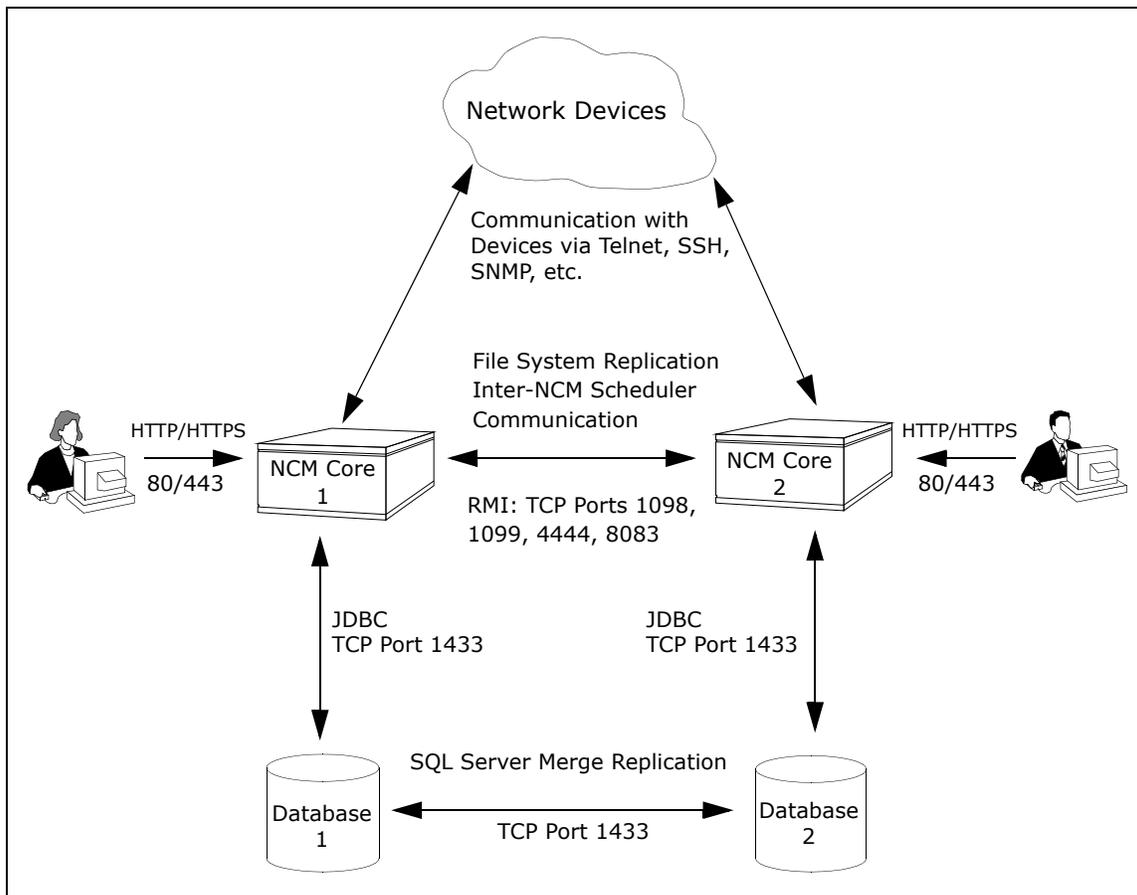
System setup requires a thorough understanding of SQL Server 2005 and NCM. Installation includes number of steps that must be performed on the various servers.

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 Partitions to ensure proper NCM Core access to devices. Refer to the *User Guide for Network Compliance Manager 1.6* for information on segmenting devices.

**Note:** The NCM High Availability Distributed System on Microsoft SQL Server requires SQL Server 2005, Service Pack 2 (Standard Edition or Enterprise Edition) or SQL Server 2008 (Standard Edition or Enterprise Edition).

## Architecture

The following figure shows a typical two-NCM Core High Availability Distributed System 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 SQL Server Merge Replication. For details on how replication works, refer to Chapter 3: System Administration.

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)

## What is Horizontal Scalability?

Horizontal Scalability is the ability to combine multiple NCM Cores with a single database so that they work as a single logical unit to improve the performance of the overall system. NCM Horizontal Scalability can be configured in different ways, depending on how you use your system and where bottlenecks are apt to occur. Refer to the *Horizontal Scalability User Guide for CiscoWorks Network Compliance Manager 1.6* for installation and configuration information.

## NCM Internal Monitoring: SQL Server Merge Replication

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

When it is not possible to use SQL Server's default handler, NCM typically has 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 CiscoWorks Network Compliance Manager 1.6* 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 "[Distributed Monitor Results Page](#)" on page 42 for detailed information on NCM monitors.

In addition, in SQL Server 2005, NCM monitors on a regular schedule the following on each NCM Core for the database transport (i.e., replication):

- Merge agent status — If the check fails and the merge agent is found to have stopped, NCM will attempt to restart it.
- Merge agent delays — If the merge agent synchronization falls behind, NCM will alert 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.

**Note:** Refer to the *User Guide for CiscoWorks Network Compliance Manager 1.6* 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 SQL Server DBA using a standard event rule in NCM. Refer to the *User Guide for CiscoWorks Network Compliance Manager 1.6* for information on configuring email notification. Refer to **“NCM Generated Events” on page 38** 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.

Topic	Refer to:
System Requirements	<a href="#">"System Requirements" on page 16</a>
Installation Planning	<a href="#">"Installation Planning" on page 16</a>
Installation and Removal	<a href="#">"Installation and Removal" on page 20</a>
Creating a Two NCM Core SQL Server Replication Environment	<a href="#">"Creating a Two NCM Core SQL Server Replication Environment" on page 21</a>
Deleting a Subscriber from a NCM Mesh	<a href="#">"Deleting a Subscriber from a NCM Mesh" on page 25</a>
Removing Replication	<a href="#">"Removing Replication" on page 27</a>
Upgrading from NCM 1.3/1.3.1 to NCM 1.4	<a href="#">"Upgrading from NCM 1.3/1.3.1 to NCM 1.4" on page 27</a>
Upgrading from NCM 1.4 to NCM 1.5	<a href="#">"Upgrading from NCM 1.4 to NCM 1.5" on page 30</a>
Upgrading from NCM 1.5 to NCM 1.6	<a href="#">"Upgrading from NCM 1.5 to NCM 1.6" on page 33</a>

---

## System Requirements

The NCM High Availability Distributed System on SQL Server software requires SQL Server 2005, Service Pack 2 (Standard Edition or Enterprise Edition) or SQL Server 2008 (Standard Edition or Enterprise Edition) Refer to the CiscoWorks Network Compliance Manager 1.6 installation documentation for information on database server hardware requirements.

## 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 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.

- 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 SQL Server 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.
- SQL Server setup planning. Access to SQL Server Management Studio and the *sqlcmd* command line tool are needed for initial database creation.

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 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 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 High Availability Distributed System installation

To assist in planning, please note the following limitations and suggestions concerning the NCM High Availability 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 and Subscriber server, you must reconfigure replication by using existing NCM data. Refer to ["Loss of a Database Server" on page 52](#) for information on restoring databases.

## 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 the figure on page 11 for information on the needed open ports.

Protocol/Database/Port	From/To
<b>NCM server (running the Mgmt Engine, Syslog, TFTP) and network devices</b>	
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.
<b>Between the database servers</b>	
SQL Server (port 1433)	In a Distributed System configuration, the SQL Server processes connect to each other on port 1433.
<b>Between the NCM servers</b>	
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.
<b>Between the NCM server and the database server</b>	
SQL Server (port 1433)	From the NCM server to an SQL Server database server.
<b>NCM server and NCM users</b>	
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.

---

## NCM Server Installation Steps

For complete information on installing NCM, refer to the CiscoWorks Network Compliance Manager 1.6 installation documentation. Keep in mind that you will have to:

- Verify time synchronization between Distributed Systems.
- Create a second database using the same database user account credentials. You do not need to use the same database name, however, for setting up replication.

**Note:** NCM service accounts do not have to have different usernames for setting up replication.

## 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 55](#) for information on Troubleshooting the installation.

## Unpacking the Replication Script Installation Bundle

The setup files for the NCM High Availability Distributed System are the standard setup files for a normal NCM installation, with the addition of the *SQLServerReplicationScriptToolBundle.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.

## 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 Distributed Systems must be at SQL Server 2005, Service Pack 2.
- A current NCM 1.4 database on NCM Core 1.
- A NCM server connected to the database on NCM Core 1.
- The SQL Sever agent service running on NCM Core 1 (Publisher) database server.
- An empty database (with no data or NCM schema setup) on NCM Core 2.
- The empty database on NCM Core 2 needs a 'dbowner' login account.
- 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. Turn off 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. Enter the "Windows username" value as "machine name\administrator" or "domain\domainUser". If you use "machine name\administrator", make sure the administrator can access the shared directory setup in Step 1 above.

5. Set the java path on NCM Core 1 and run the script tool:
  - `C:\Program Files\Microsoft SQL Server\90> set JAVA_HOME = C:\Rendition\jre`
  - `C:\Program Files\Microsoft SQL Server\90>set PATH = %PATH% ; C:\Rendition\jre\bin`
  - Enter `java` at the prompt to verify that value returned is java 1.4.2 or higher.
  - Run the script tool. This will output two files. Enter:  
`java -classpath . SQLServerReplicationScriptTool`
6. Copy the *SQLServerPreSnapshotScript.sql* 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 *SQLServerReplicationScript.sql* file using the 'sqlcmd' command: `sqlcmd -S <PublisherServer> -U <USER> -P <PASSWORD> -i SQLServerReplicationScript.sql -o log.txt`

Where <PublisherServer> is the Publisher's server, and <USER> and <PASSWORD> are the credentials for a login that is a member of the sysadmin role. (Refer to the eighth bullet in "Preparation Requirements" on page 21.)

8. Using sqlcmd, enter the following t-sql query on both the NCM Core 1 database and the NCM Core 2 database.

NCM Core 1:

```
> sqlcmd -S PublisherServer -U sa -P sa password
1> use <DatabaseName>
2> Select count(*) from Information_Schema.Tables where
Table_Type = 'Base Table' and Table_Name like 'RN_%'
3> go
```

NCM Core 2:

```
> sqlcmd -S SubscriberServer -U sa -P sa password
1> use <Databasename>
2> Select count(*) from Information_Schema.Tables where
Table_Type = 'Base Table' and Table_Name like 'RN_%'
3> go
```

Once the results of the queries are the same on both NCM Cores, the replication has been setup between the NCM Cores.

9. Go to RN\_CORE within SQL database on each NCM Core and verify that the file sizes are the same. Note that <DatabaseName> is the appropriate database name for each server.)
10. Install a NCM server. Be sure it points to an empty database on NCM Core 2. When prompted, have NCM connect to the database using the login from the sixth bullet in the ["Preparation Requirements" on page 21](#) section.
11. Verify that the replication is working correctly. Make sure:
  - The log 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 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:

On the Publisher, enter:

```
UPDATE RN_CORE SET COMMENTS = 'Test from db1 to db2' WHERE  
CoreID=1;
```

wait at least 60 seconds

On the Subscriber, enter:

```
SELECT Comments from RN_CORE where CoreID=1;
```

You should then reverse this, executing the update on the Subscriber with a new comment and the select on the Publisher. Be sure to execute this from every NCM Core in the NCM Mesh. Check the values each time on all of the other NCM Cores to ensure that the replication between each database in the system are working correctly.

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

13. Start the NCM server on NCM Core 1.

14. Start the NCM server on NCM Core 2.

## 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.

### Deletion 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 partitions on NCM Cores that are not going to be removed.
3. Modify all partitions to point to a NCM Core that is not being removed. Alternatively, remove those partitions.
4. Turn off the NCM server from the NCM Core that is being removed.

5. Delete the RN\_CORE entry that was removed using the following commands using sqlcmd:

**Note:** To identify the Core ID of the NCM Core to delete, enter:

```
SELECT * FROM RN_CORE;
```

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

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 -cp . SQLServerReplicationScriptTool`

8. Run the first output file using sqlcmd with a login that is a member of the sysadmin role on NCM Core 1. Enter:

```
sqlcmd -S <PublisherServer> -U <USER> -P <PASSWORD> -i  
SQLServerReplicationScript.sql -o log.txt
```

Where <PublisherServer> is the Publisher's server, and <USER> and <PASSWORD> were collected in step 1.

9. Delete the *distributed.rcx* file on the NCM server if it is the only NCM Core in the NCM Mesh. Restart the NCM server on NCM Core 1.

## Removing Replication

To remove replication:

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

```
use [master]
exec sp_dropdistributor @no_checks = 1
GO
```

**Note:** After removing replication, edit the RN\_CORE entry. Refer to ["Deletion Steps" on page 25](#) for information.

## Upgrading from NCM 1.3/1.3.1 to NCM 1.4

During the NCM upgrade process, the NCM Cores must be completely off-line. Keep in mind that the upgrade instructions only work on a current, up-to-date NCM 1.3 or NCM 1.3.1 database running on all servers.

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.

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

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

To upgrade a NCM Mesh:

1. Turn off all NCM servers in the NCM Mesh.
2. Collect the following information:
  - a) Login name and password of a SQL Server login that is a member of sysadmin on NCM Core 1 and NCM Core 2.
  - b) 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\_1\_3" or "upgrade\_from\_1\_3\_1". These properties are described in detail in the file.

4. Run the script tool. Enter:

```
java -cp . SQLServerReplicationScriptTool
```

5. Run the first output file using sqlcmd with a login that is a member of the sysadmin role on NCM Core 1. Enter:

```
sqlcmd -S <PublisherServer> -U <user> -P <password> -i  
SQLServerReplicationScript.sql -o log.txt
```

Where <PublisherServer> is the server of Publisher, <user> and <password> were collected in Step 2.

6. Verify that the replication is working correctly. Make sure:
  - The log 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 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:

On the Publisher, enter:

```
UPDATE RN_CORE SET COMMENTS = 'Test from db1 to db2' WHERE  
CoreID=1;
```

wait at least 60 seconds

On the Subscriber, enter:

```
SELECT Comments from RN_CORE where CoreID=1;
```

You should then reverse this, executing the update on the Subscriber with a new comment and the select on the Publisher. Be sure to execute this from every NCM Core in the NCM Mesh. Check the values each time on all of the other NCM Cores to ensure that the replication between each database in the system are working correctly.

7. Upgrade all NCM servers using the NCM 1.4 Service Pack Installer. For information on running the NCM 1.4 Service Pack Installer, refer to the *Upgrade and Installation User Guide for Network Compliance Manager 1.4*.
8. Restart the NCM servers in the NCM Mesh.

## Upgrading from NCM 1.4 to NCM 1.5

To update a system to NCM 1.5, you must first upgrade to NCM 1.4.

During the NCM upgrade process, the NCM Cores must be completely off-line. Keep in mind that the upgrade instructions only work on a current, up-to-date NCM 1.4 database running on all servers.

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.

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

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

To upgrade a NCM Mesh:

1. Turn off all NCM servers in the NCM Mesh.
2. Collect the following information:
  - a) Login name and password of a SQL Server login that is a member of sysadmin on NCM Core 1 and NCM Core 2.
  - b) Database name, NCM server hostname, NCMNCM 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\_1\_4". These properties are described in detail in the file.

4. Run the script tool. Note that you must run this on a system that has Java 1.6.x installed. Enter:

```
java -cp . SQLServerReplicationScriptTool
```

5. Run the first output file using sqlcmd with a login that is a member of the sysadmin role on NCM Core 1. Enter:

```
sqlcmd -S <PublisherServer> -U <user> -P <password> -i  
SQLServerReplicationScript.sql -o log.txt
```

Where <PublisherServer> is the server of Publisher, <user> and <password> were collected in Step 2.

6. Verify that the replication is working correctly. Make sure:
  - The log 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 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:

On the Publisher, enter:

```
UPDATE RN_CORE SET COMMENTS = 'Test from db1 to db2' WHERE  
CoreID=1;
```

wait at least 60 seconds

On the Subscriber, enter:

```
SELECT Comments from RN_CORE where CoreID=1;
```

You should then reverse this, executing the update on the Subscriber with a new comment and the select on the Publisher. Be sure to execute this from every NCM Core in the NCM Mesh. Check the values each time on all of the other NCM Cores to ensure that the replication between each database in the system are working correctly.

7. Upgrade all NCM servers using the NCM 1.5 Service Pack Installer. For information on running the NCM 1.5 Service Pack Installer, refer to the CiscoWorks Network Compliance Manager 1.5 installation documentaton. Restart the NCM servers in the NCM Mesh.

## Upgrading from NCM 1.5 to NCM 1.6

During the NCM upgrade process, the NCM Cores must be completely off-line. Keep in mind that the upgrade instructions only work on a current, up-to-date NCM 1.5 database running on all servers.

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.

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

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

To upgrade a NCM Mesh:

1. Turn off all NCM servers in the NCM Mesh.
2. Collect the following information:
  - a) Login name and password of a SQL Server login that is a member of sysadmin on NCM Core 1 and NCM Core 2.
  - b) 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\_7\_6". These properties are described in detail in the file.

4. Run the script tool. Note that you must run this on a system that has Java 1.6.x installed. Enter:

```
java -cp . SQLServerReplicationScriptTool
```

5. Run the first output file using sqlcmd with a login that is a member of the sysadmin role on NCM Core 1. Enter:

```
sqlcmd -S <PublisherServer> -U <user> -P <password> -i  
SQLServerReplicationScript.sql -o log.txt
```

Where <PublisherServer> is the server of Publisher, <user> and <password> were collected in Step 2.

6. Verify that the replication is working correctly. Make sure:
  - The log 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 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:

On the Publisher, enter:

```
UPDATE RN_CORE SET COMMENTS = 'Test from db1 to db2' WHERE  
CoreID=1;
```

wait at least 60 seconds

On the Subscriber, enter:

```
SELECT Comments from RN_CORE where CoreID=1;
```

You should then reverse this, executing the update on the Subscriber with a new comment and the select on the Publisher. Be sure to execute this from every NCM Core in the NCM Mesh. Check the values each time on all of the other NCM Cores to ensure that the replication between each database in the system are working correctly.

7. Upgrade all NCM servers using the NCM 1.6 Service Pack Installer. For information on running the NCM 1.6 Service Pack Installer, refer to the CiscoWorks Network Compliance Manager 1.6 installation documentation.
8. Restart the NCM servers in the NCM Mesh.



# Chapter 3: System Administration

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

Topic	Refer to:
Getting Started	<a href="#">"Getting Started" on page 37</a>
NCM Generated Events	<a href="#">"NCM Generated Events" on page 38</a>
Distributed Monitor Results	<a href="#">"Distributed Monitor Results Page" on page 42</a>
Site Reassignment page	<a href="#">"Site Reassignment Page" on page 46</a>
List Cores page	<a href="#">"List Cores Page" on page 46</a>
Edit Core page	<a href="#">"Edit Core Page" on page 47</a>
Device Password Rule Priority Reset page	<a href="#">"Device Password Rule Priority Reset Page" on page 48</a>
Renew Configuration Options page	<a href="#">"Renew Configuration Options Page" on page 48</a>

## 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 48](#).

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.

## Distributed System - Stopped Merge Agent Job

### Event format:

```
Merge Agent Job Stopped  
<Job Details>  
[Successfully attempted to restart the job.]
```

NCM monitors the replication merge agent and sends the event if it determines that the agent has stopped. If the "Successfully attempted to restart the job" message is displayed, NCM restarted the agent. The agent's status can be reviewed in SQLServer Management Studio.

## Distributed System - Data Synchronization Delay Warning

### Event format:

```
Publisher '<DatabaseName>' has been unable to push changes  
to subscriber '<DatabaseName2>' since <tim>. If you wish to  
avoid potential data loss, please rectify this as soon as  
possible.
```

```
Subscriber '<DatabaseName2>' has been unable to push changes  
to Publisher '<DatabaseName>' since <tim>. If you wish to  
avoid potential data loss, please rectify this as soon as  
possible.
```

NCM has detected that the replication system is having trouble keeping the data and the databases in sync. The databases, merge agents, and network connections between the databases should be examined for issues.

## Using the NCM Distributed System Pages

When you install the NCM High Availability 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

The screenshot shows the CiscoWorks Network Compliance Manager user interface. The top navigation bar includes the Cisco logo, the product name "CiscoWorks Network Compliance Manager", and a "Logout" button. Below the navigation bar are several menu items: "Devices", "Tasks", "Policies", "Reports", "Admin", and "Help". The "Admin" menu is expanded, showing a list of options. An arrow points from the text "Distributed System Pages" to the "Distributed" section of the Admin menu.

Menu Item	Sub-items
Admin	<ul style="list-style-type: none"> <li>Users</li> <li>User Groups</li> <li>New User</li> <li>New User Group</li> <li>Logged on Users</li> </ul>
	User Roles & Permissions
	<ul style="list-style-type: none"> <li>Security Partitions</li> <li>Gateways</li> <li>Device Password Rules</li> <li>Event Notification &amp; Response Rules</li> </ul>
	<ul style="list-style-type: none"> <li>Custom Data Setup</li> <li>LDAP Setup</li> <li>Workflow Setup</li> </ul>
	Administrative Settings ▶
	<b>Distributed</b> ▶
	<ul style="list-style-type: none"> <li><b>Monitor Results</b></li> <li><b>Conflict List</b></li> <li><b>Site Reassignment</b></li> <li><b>Core List</b></li> <li><b>Device Password Rule Priority Reset</b></li> <li><b>Renew Configuration Options</b></li> </ul>
	<ul style="list-style-type: none"> <li>Task Load</li> <li>System Status</li> <li>Start/Stop Services</li> <li>Troubleshooting</li> <li>Drivers</li> </ul>
	<ul style="list-style-type: none"> <li>About CiscoWorks Network Compliance Manager</li> </ul>
	System Tasks ▶

## 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.

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 reports that here. Stopped Merge Agent jobs should be restarted as soon as possible.
- **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.

**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.

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

## 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.

**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 CiscoWorks Network Compliance Manager 1.6* 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: <ul style="list-style-type: none"><li>• Detail — Opens the View Distributed Conflict page, where you can view details on an individual uniqueness constraint. Refer to "<a href="#">View Distributed Conflict Page</a>" on page 45.</li><li>• Delete — Deletes the conflict from the database.</li></ul>

---

## 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.

### 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. 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 Publisher? (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: <ul style="list-style-type: none"><li>• Edit — Open the Edit Core page. Refer to <a href="#">"Edit Core Page"</a> on page 47.</li></ul>

---

## 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 and click the Save Core button:

- Name — Enter the NCM Core name.
- Database Identifier — Enter the database name. This information is needed to make connections to a particular SQL Server instance on a server.
- 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 — Not applicable for SQL Server.
- Confirm Replication Password — Not applicable for SQL Server.
- 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 CiscoWorks Network Compliance Manager 1.6*.

## 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 49
Recovery	"Recovery" on page 49
Loss of Network Connectivity	"Loss of Network Connectivity" on page 50
Loss of a NCM Server	"Loss of a NCM Server" on page 51
Loss of a Database Server	"Loss of a Database Server" on page 52

## 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 46 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 46](#).
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 CiscoWorks Network Compliance Manager 1.6* 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 48](#).
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 have been re-assigned back to their original NCM Core. This can be accomplished in NCM. Refer to ["Site Reassignment Page" on page 46](#).
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 CiscoWorks Network Compliance Manager 1.6* 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 48](#).
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 another working NCM server in the NCM Mesh.
2. If the lost database server is a Subscriber, replication needs to be removed from the NCM Mesh if the lost Subscriber is the only Subscriber in the NCM Mesh. Refer to ["Removing Replication" on page 27](#) for information. Keep in mind that Replication can be reconfigured after removal. Refer to ["Creating a Two NCM Core SQL Server Replication Environment" on page 21](#) for information.
3. If the lost database server is the Publisher, replication needs to be reinstalled.
  - a) Export all RN\_% tables from the Subscriber database.
  - b) Create a new database on the NCM Core 1 database server with data exported from the Subscriber database.
  - c) Reinstall NCM on the NCM Core 1 Server. Be sure NCM points to the existing database on NCM Core 1.
  - d) Edit the RN\_CORE entry. Refer ["Deletion Steps" on page 25](#) (step 5).
  - e) Install replication. Refer to ["Creating a Two NCM Core SQL Server Replication Environment" on page 21](#) for information.
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 46](#) 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 the *User Guide for CiscoWorks Network Compliance Manager 1.6* for information.



# 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 55
Removing In-Memory and Database Information	"Removing In-Memory and Database Information" on page 56

## 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 if 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

## B

- Buttons
  - Renew Config Options 48
  - Reset Priority 48

## C

- Conflict table 45

## D

- Deleting a Subscriber 25
- Deleting a subscriber 25
- Device Password Rule Reset page 48
- Distributed Conflict List 44
- Distributed Monitor Results page 42
- Distributed systems
  - Failover 49
  - Installation 16
  - Overview 8
  - Recovery 50
  - Terminology 7
  - Troubleshooting 55

## E

- Edit Core page 47
- Events
  - RMI error 39
  - Status 44, 45
  - Time synchronization warnings 39

## F

- Failover 51
- Files
  - appserver.rcx 51
  - distributed.rcx 24

## H

- Horizontal scalability 11

## I

- Installation
  - Creating two NCM Cores 21
  - Planning 16
  - SQL Server setup 16
  - Unpacking the Replication script 21

## L

- List Cores page 46

## M

- Monitor results 42

## N

- NCM
  - Core 7
  - Customer Support 6
  - Device Password Rule Reset page 48
  - Distributed Conflict List 44
  - Distributed Monitor Results page 42
  - Edit Core page 47
  - Generated events 38
  - List Cores page 46
  - Mesh 7
  - Renew Configuration Options 48
  - Site Reassignment page 46
  - View Distributed Conflict page 45
- NCM scheduler 8

## P

- Ports
  - JNDI 19
  - SNMP 19
  - Telnet 19

## R

- Recovery
  - Loss of network connectivity 49
  - Replication data 50
- Removing database information 56
- Removing replication 27
- Renew Configuration Options 48

**Replication**

- Monitoring 8
- Removing 27
- Script tool 23
- Two NCM Cores 21
- Unpacking 21

RMI connections 42

Rowguid 44, 45

**S**

Site Reassignment page 46

**SQL Server**

- Deleting a subscriber 25
- Multimaster replication 8
- Publisher 21
- Subscriber 21

sqlcmd 23

**Subscriber**

- Definition 7
- Deleting 25

**System administration**

- List Cores 46
- Monitor results 42
- NCM generated events 38
- NCM UI pages 39, 41
- Site reassignment 46
- Uniqueness conflicts 44

**T**

Terminology 7

Timezone offset 46, 47

**Troubleshooting**

- Removing information 56
- Replication setup 55

**U**

Uniqueness conflicts 38, 42

**V**

View Distributed Conflict page 45