EMC²
where information lives®

White Paper

Backup and Recovery of EMC Documentum Content Server Using the NetWorker Module for Documentum

Applied Technology

Abstract

This white paper outlines the various configurations supported by the EMC® NetWorker® Module for Documentum®. The paper helps users understand the solution that the NetWorker Module for Documentum provides, make content stored in EMC Documentum Content Server available to enterprises, and protect the content against data loss. The NetWorker Module for Documentum is for fast, bulk admin backup and restore of physical Documentum docbase/repository components.

April 2009

Part Number H4071.1

Table of Contents

Executive summary

The EMC® NetWorker® Module for Documentum® (NMD) is the NetWorker Module that facilitates the backup and recovery of various objects of EMC Documentum Content Server. The solution ensures the consistent backup of the EMC Documentum Content Server by backing up the data in a manner that ensures all the objects belong to the same point in time.

EMC Documentum Content Server is core functionality that provides users with the options to create, capture, manage, deliver, and archive enterprise content.

With NMD, enterprises can leverage NetWorker's industry-proven solution of backup and recovery of applications with features like backup, scheduling, archiving, staging, media management, and recoveries.

Introduction

This white paper provides insight into the different EMC Documentum Content Server objects that can be backed up using NMD. Also, this white paper provides details of specific configurations that include remote full-text indexes (FTIs), distributed storage areas, repository backups, and backups using specific modules like SnapImage™ and database modules (for Oracle, Sybase, SQL, and DB2).

An EMC Documentum Content Server stores the contents in the repositories in the form of objects. Metadata about these objects are stored in the underlying relational database management system, such as an Oracle, Microsoft SQL Server, or other RDBMSs. The content files associated with the objects can be stored in various media, such as file systems, databases, or external storage devices. This paper focuses on the content that is stored in file systems (the most common medium to store the content files).

A FTI enables quick search and retrieval of text strings within content files and attributes. A FTI is an optional component, and a user must install the FTI components that create and maintain the indexes.

Distributed storage areas (DSAs) are configurations supported by EMC Documentum Content Server version 5.3, which enable users to have the content and metadata distributed across sites.

Distributed content can be set up for a Single-Repository or Multiple-Repository model. In a Single-Repository model, content can be distributed across all the participating sites. Multiple repositories enable content and metadata to be distributed across all the participating repositories. NMD supports the backup of the Single-Repository model (discussed later).

NetWorker is the EMC flagship product for backup and restore of enterprise data. Supported types of backups range from simple file system backups to application backups of Documentum, Oracle, Microsoft SQL Server, and other applications. To back up each application, NetWorker provides plug-ins called NetWorker Modules. NMD is one such module. This white paper provides details on how NMD facilitates the backup and recovery of various objects of EMC Documentum Content Server, such as DSAs, SAs, FTIs, and databases.

Audience

This white paper can be used by anyone who uses the NMD solution to back up EMC Documentum Content Server. Basic understanding of the EMC Documentum Content Server and its components such as repositories, FTIs, SAs, DSAs, and databases is a must for understanding the NMD implementation.

Terminology

DSA: Distributed storage areas, which enable the Content Server contents to be distributed across participating sites

FTI: Full-text index, which is an optional component that facilitates the creation and maintenance of indexes used for searching for specific values in a repository

ICF: Documentum installation and configuration files

NMD: NetWorker Module for Documentum, the plug-in for NetWorker software that provides support for backup and recovery of different Documentum objects

RDBMS: Relational database management systems

Repository: Overall content management system, which is a grouping of different Content Server objects

SA: Storage area, which is a logical entity of the Content Server where the user contents can be stored

EMC Documentum Content Server overview

EMC Documentum Content Server is an enterprise content management system that provides a systematic solution for organizing, storing, and delivering unstructured content within an enterprise and beyond. In EMC Documentum Content Server, unstructured content is managed according to predefined business rules, policies, and procedures. A Content Server has a comprehensive, extensible, open, scalable, and secure architecture that meets the need of global and distributed enterprises.

EMC Documentum Content Server enables a user to:

- Create or import files of different formats
- Manage and track the different versions of a content file
- Create and manage virtual documents
- Add and remove renditions for a content file
- Index the content file contents for rapid searching

The Content Server repository stores content in a unified environment. The repository is responsive to the business needs of an organization. This adaptive feature of the repository comes in handy for the enterprise that operates in multiple locations and requires a distributed repository or multiple repositories. The Content Server repository stores contents and its associated metadata by using an object-oriented model.

Documents are composed of content files and document attributes. The content can be in any form: text documents, spreadsheets, pictures, XML files, web pages, full-motion video, streaming audio, instant messages, email messages, and fixed content such as reports, images, and scanned images. The attributes or the metadata, such as document owner, version, and creation date, are used by the repository to organize the content and by the user to search for content. Contents are stored in file systems, and metadata is stored in tables underlying relational databases.

A user accesses contents of the repository through connection brokers. A connection broker is a process that enables clients to connect to EMC Documentum Content Server and, in turn, to a specific repository. A repository must have at least one connection broker so that users can access and manage the contents stored in the repository. A connection broker can project to *n* number of repositories, and vice versa.

A repository may consist of the following three components: SA, FTI server (optional), and RDBMS. All components behave as a single entity from an application point of view.

Figure 1 shows the three main components of a repository. Components are then discussed in more detail.

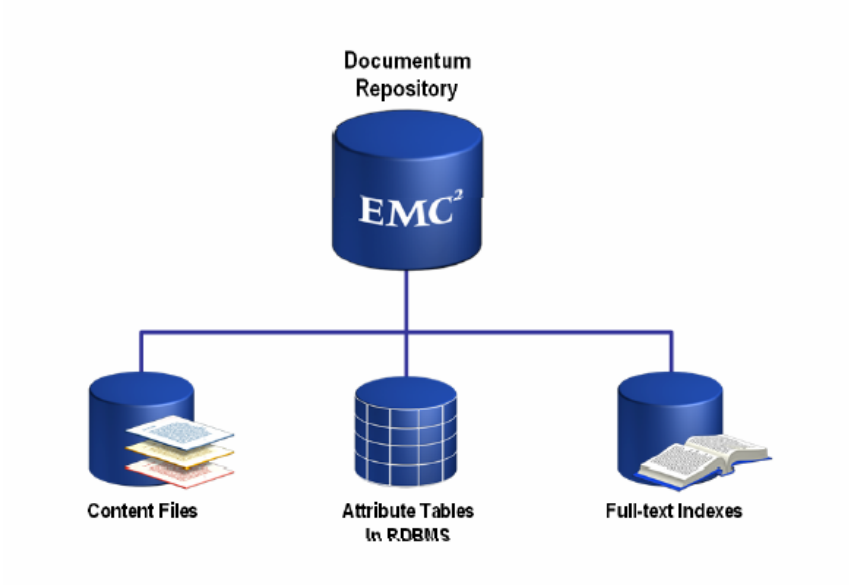

Figure 1. Three main components of a repository

Storage area (SA): A SA contains the content files of all formats. The EMC Documentum Content Server supports different types of SAs. Users can choose any SA option, depending on the contents stored in the repository. The SA options are binary large object (BLOB) storage, turbo storage, content-addressed storage (CAS), external storage, linked storage, distributed storage, and file system directories. Each SA points to a location where the contents are stored. The location may be a file system residing on the same host as the EMC Documentum Content Server or on a remote host, any external storage device, or databases.

EMC Documentum Content Server provides distributed repositories so that companies can access and deploy their content at any time and at any place around the world. Large enterprises require distributed repositories or multiple repositories. For example, a global company might have a Documentum content repository in each geographical region, with a goal of storing data locally to the users who work on it.

Relational Database Management Systems (RDBMS): Content attributes are stored in the tables of a relational database. Storing metadata in attribute tables enables rapid query and retrieval. For operational, performance, and security reasons, all the content can be stored in the RDBMS and the content assets can be stored as a BLOB adjacent to attribute tables.

EMC Documentum Content Server supports both local and remote RDBMS, with Microsoft SQL Server, Oracle, Sybase, or DB2 databases as the RDBMS.

Full-text index (FTI): EMC Documentum Content Server maintains a FTI for all the content stored in the repository. The FTI enables rapid search through a large collection of unstructured information. In addition to searching the text within the content, the FTI also searches content attributes. The indexed content can include documents, text files, XML components, HTML files, and closed-caption tracks of video files.

When a document is added to the repository, the associated content files are added to the SA's index and then the index is updated. The FTI that is automatically created when the content is added to the repository contains all the words of contents stored in the repository, and all keywords and content attributes that describe the content.

The indexing process can run on the same host as that of the EMC Documentum Content Server or on a remote host. As part of the indexing process, the index agent forwards the content to the FTI server, which maintains a FTI database. EMC Documentum Content Server ensures that the query performance and scalability are not affected by the repository size.

Installation and configuration files (ICF): This refers to the EMC Documentum installation and configuration files.

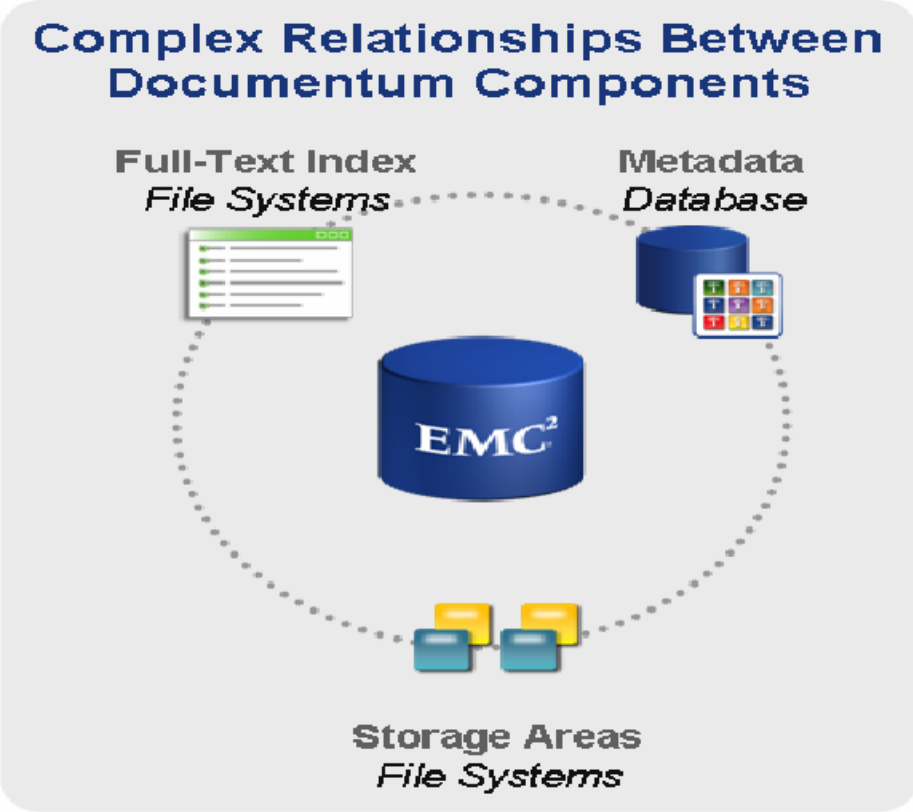

Figure 2. Documentum components

NMD overview

NMD is a solution to back up and restore a Documentum system. NMD can perform hot backups without interrupting Documentum operations. It can back up and restore individual components of the Documentum repository, or the Documentum system as a whole.

NMD supports all features of Documentum 5.3.x. and limited features of Documentum 6.x. It supports the backup and restore of the following repository components:

- Local SA or DSA

  With DSAs, NMD supports only the Single-Repository model. In a DSA configuration, the SA on the remote host is accessible either through the primary host or the remote host through local disks, SAN, shared SCSI, NAS, or NFS.

- Database server

  NMD supports both local and remote databases. The database server may be installed on the primary host or on a remote host.

- Local and remote FTI

The FTI server may be installed on the primary host or on a separate host. NMD supports backup of local and remote FTIs.

NMD supports both manual and scheduled backups of all the components of a repository. By default, a full repository backup includes the underlying database, all FTIs, and all SAs. Optionally, the backup may also include ICF and database logs. Backup of a SA may be performed either by using the traditional NetWorker backup technology or by using snapshot technology. NMD supports snapshot backups only for SAs with local and distributed configurations.

NMD maintains backup indexes and records of all scheduled and manual backups. A user can browse through the indexes and then restore relevant content. NMD supports both file-by-file recovery and complete restore of components of the Documentum content repository. File-by-file recovery can be done only for a SA by using the file reports generated during a SA or DSA backup (explained later in this white paper).

Although NMD can back up and restore individual components of the repository, you should back up the repository as a whole with all the components. Backing up components of a repository independently may lead to unreferenced content because a proper sequence is not followed. Orphaned files can occur during the lag between the database and SA backup.

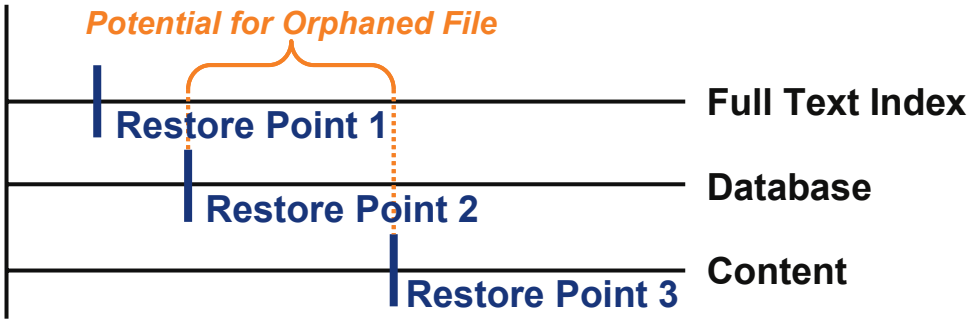

Figure 3. Inconsistent backup of Documentum components

NMD ensures a consistent full backup of a repository so that the backed-up components are restored to a consistent state. NMD achieves consistency by using the following backup workflow:

1. Suspend Documentum jobs of FTIs
2. Suspend Documentum jobs of SAs
3. Back up FTIs
4. Suspend Documentum jobs of the database
5. Back up the database
6. Restart suspended Documentum jobs of the database
7. Back up SAs
8. Restart suspended Documentum jobs of SAs
9. Restart suspended Documentum jobs of FTIs

NMD makes use of a configuration file for backup and restore. All mandatory and optional parameters are defined in the NMD configuration file. On installation of NMD, a sample NMD configuration file is installed in a default location. A user may customize this file according to the user environment and configuration. Also, it is recommended that the customized file be relocated to a different location.

How SA backups work

An NMD SA backup involves the following process interactions:

1. The scheduled or manual backup invokes the **nsrnmdsv** process, which uses the mandatory **-f** option of the **nsrnmdsv** command to locate the NMD configuration file.

2. The **nsrnmdsv** process reads the parameters from the configuration file, and uses those parameters and the **nsrnmdsv** command options to determine the backup to perform.

3. For an incremental backup, the **nsrnmdsv** process obtains the required backup information from the NetWorker client file index.

4. The **nsrnmdde** process or a user-created script (as specified in the configuration file) quiesces the SA.

   The Documentum Server jobs, dm_DMClean and dm_DMFileScan, are disabled.

   Note: If the Documentum dm_DMClean job is disabled *before* this step, the SA quiesce fails during an NMD backup that includes one or more SAs, causing the backup to *fail*.

5. The **nsrnmdsv** process calls the **nsrnmdde** process to obtain the required information (for example, file pathnames) about the components (a single SA component or all the SA components) of the repository from the Documentum Server.

6. The **nsrnmdsv** process creates a lock file (named /nsr/applogs/nsrnmdsv.*repository_name*.lck on UNIX or Linux and *installation_path*\nsr\applogs\nsrnmdsv.*repository_name*.lck on Windows) to prevent concurrent backups for the same repository.

7. To back up a SA:

   a. The **nsrnmdde** process obtains the required information (for example, subdirectory and file pathnames) about the SA from the Documentum Server.

   b. The **nsrnmdsv** process invokes the NetWorker **save** process. If it's a SnapImage backup, **nsrndmp_save** will be invoked.

8. The **nsrnmdde** process or a user-created script (as specified in the configuration file) unquiesces the SA. The Documentum Server jobs, dm_DMClean and dm_DMFileScan, are enabled.

9. The **nsrnmdsv** process stores the backup index records in the online NetWorker indexes.

10. The **nsrnmdsv** process deletes the lock file created in Step 6.

nsrnmdde is a Java binary that sends requests to the Documentum Server for component discovery and details, quiesce and unquiesce of components, and NMD customized file report dumps.

The dm_DMClean and dm_DMFileScan Documentum Server jobs are tools that automate the dmclean and dmfilescan utilities, respectively. The dmclean utility scans a repository and finds all orphaned (deleted or unreferenced) content objects, while the dmfilescan utility scans a specified SA (or all SAs) and finds all orphaned content files. Both utilities generate a script file that may be run to remove the orphaned content objects or orphaned content files. The Documentum documentation provides more information on these utilities.

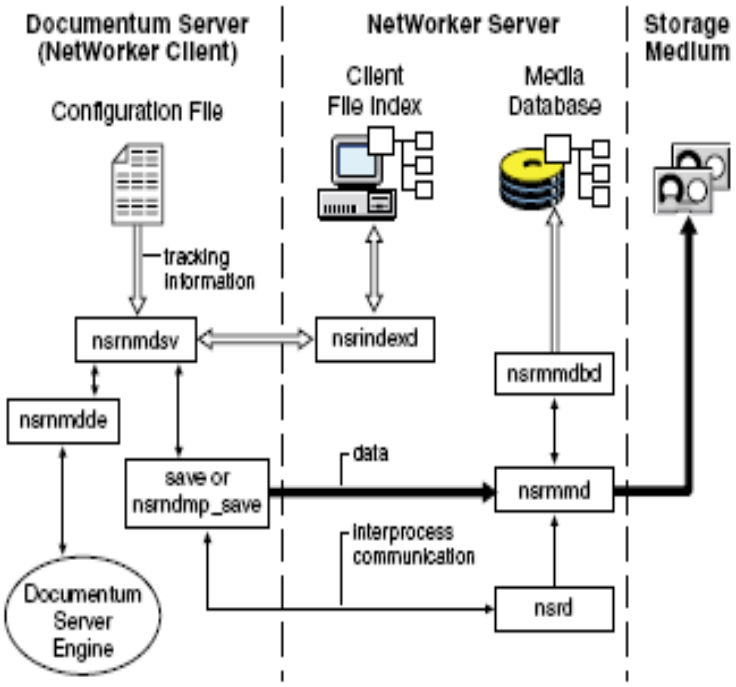

Figure 4. Process interaction in an NMD backup

Backup of a storage area

The backup of a storage area (SA) is a simple process. The proper configuration of the NMD configuration file ensures a successful SA backup. Once the correct versions of the NetWorker software and NMD are in place, you may proceed with the SA backup.

A SA backup may be performed in two ways:

- Through the command line interface (CLI), as a manual backup
- Through the NetWorker Management Console (NMC), as a scheduled backup.

The difference between manual and scheduled backups is the way the **nsrnmdsv** command (which is the NMD backup command) is executed. A comparison between manual and scheduled backups is summarized in Table 1.

Table 1. Comparison between manual and scheduled NMD backups

Factor	Manual backup	Scheduled backup
Command specification	Command line	**Backup Command** attribute of the NetWorker Client resource.
Command execution	Immediately after the backup command is entered	Backup is started at the time specified by the **Start Time** attribute in the Group resource (of which the client is a part), provided that **Autostart** is enabled. **Note:** The backup can also be started manually through the NMC by selecting the appropriate group in the **Monitoring** tab and then clicking the **Run** button.

Preparing for SA backups

This section describes important NMD configuration file parameters and command line options that are used for NMD backups.

NMD configuration file parameters

To ensure successful SA backups, the following parameters must be set in the NMD configuration file:

- DOCUMENTUM
- DOCUMENTUM_SHARED
- DM_HOME
- CLASSPATH
- JAVA_PATH
- LD_LIBRARY_PATH or LIBPATH or SHLIB_PATH
- PATH
- NMDDE_DM_DOCBASE
- NMDDE_DM_USER
- NMDDE_DM_PASSWD
- NMD_SCOPE
- NMD_OBJECT_NAME
- NSR_BACKUP_LEVEL
- NSR_CLIENT
- NSR_DATA_VOLUME_POOL
- NSR_GROUP
- NSR_SERVER

Although all of the preceding parameters must be set to appropriate values, three parameters of utmost importance for SA backups are NMDDE_DM_DOCBASE, NMD_SCOPE, and NMD_OBJECT_NAME, with the NMD_OBJECT_NAME parameter having relevance only in the context of single SA component backups.

- The NMDDE_DM_DOCBASE parameter specifies the name of the repository containing the SA components to be backed up.
- The NMD_SCOPE parameter must be set to SA or SA_ALL, indicating that the NMD backup pertains to backing up of a single SA component or all of the SA components of a repository, respectively.
- The NMD_OBJECT_NAME parameter specifies the name of the SA component to be backed up when NMD_SCOPE is set to SA for a single SA backup. A single repository may consist of several SA components, and hence identifying the particular SA component to be backed up is mandatory for a single SA backup.

Remove (or ideally, comment out) the line containing the NMD_OBJECT_NAME parameter setting when the scope of the backup is SA_ALL. If this parameter is left uncommented, irrespective of the scope of the backup, this parameter will be read and will lead to failure of the NMD backup. NMD_OBJECT_NAME has no relevance when all of the SA components of a particular repository must be backed up. The parameter is not ignored when the backup scope is set to SA_ALL.

NMD command options

The **nsrnmdsv** command options override any corresponding parameters set in the NMD configuration file. Table 2 summarizes the relation between the **nsrnmdsv** command options and the corresponding parameter settings in the NMD configuration file.

Table 2. Comparison of NMD configuration file parameters and nsrnmdsv command options

NMD configuration file parameter	nsrnmdsv command option	Description
NMDDE_DM_DOCBASE =*repository_name*	**–B** *repository_name*	Name of the Documentum repository containing the SA to be backed up.
NMD_SCOPE =SA | SA_ALL	**–M SA | SA_ALL**	Specifies that the backup scope is a single SA component or all SA components of the repository.
NMD_OBJECT_NAME =*SA_name*	**–a** *SA_name*	Specifies the name of the SA component to be backed up. The command option must be immediately preceded by the **–M SA** option. **Note:** This parameter or option has relevance only for the backup of a single SA component.
(Not applicable)	**–f** *config_filename*	Specifies the complete path of the NMD configuration file for the SA backup. There is no parameter setting in the NMD configuration file that corresponds to the **–f** option.
NSR_BACKUP_LEVEL =full | incr	**-l full | incr**	Specifies the level of backup. The backup level may be either full or incr (which stands for incremental). **Note:** Irrespective of the level specified in the NMD configuration file, or in the **Level** attribute of the corresponding Group resource, the level of the first SA backup is always full.

In addition to the preceding options, the **nsrnmdsv** command has a rich set of other options that help the user to perform customized backups:

nsrnmdsv -f *config_file* [**-B** *repos_name*] [**-M ALL**|**SA_ALL**|**FTI_ALL**|**ICF**|**DB**|**DB_LOG**] [**-D** *debug_level*] [*save_options*]
nsrnmdsv -f *config_file* [**-B** *repos_name*] [**-M SA**|**FTI -a** *name*] [**-D** *debug_level*] [*save_options*]
nsrnmdsv -P *PASSWD_text* **-f** *config_file* [**-M FTI**]

save_options:
[**-c** *client*] [**-s** *server*] [**-l** *level*] [**-b** *pool*] [**-g** *group*] [**-p** *parallelism*] [**-N** *saveset_name*] [**-q**] [**-v**] [**-w** *browse_time*] [**-y** *retention_time*] [*savegrp_internal_options**]

**savegrp_internal_options* are internally used by savegrp command.

Irrespective of the parameters set in the NMD configuration file, the **–f** *config_file* option is mandatory for the **nsrnmdsv** command at the command line. (The reason is that, although all of the backup parameters may be specified at the command line, critical path variable values like PATH, CLASSPATH, and others are still read from the NMD configuration file.)

Backing up SAs

NMD allows you to back up a single SA component in a repository or all of the SA components in a repository. This section illustrates different methods of performing SA backups.

Backup of a single SA component

The following examples discuss the backup of a single SA component. The host runs on an HP-UX platform. The repository in use is repos01 and the SA component to be backed up is filestore_01.

Example 1

Assuming that all the mandatory parameters are set in the NMD configuration file, the following **nsrnmdsv** command (specified at the command line) performs the backup of a single SA component:

nsrnmdsv –f /space10/home/dmuser/nmd.cfg

The preceding command requires the following parameters to be set appropriately in the NMD configuration file (which is specified by the **–f** option of the **nsrnmdsv** command):

- NMDDE_DM_DOCBASE=repos01
- NMD_SCOPE=SA
- NMD_OBJECT_NAME=filestore_01

Example 2

Assuming that all the pathnames and other NetWorker parameters are appropriately set in the NMD configuration file, the following **nsrnmdsv** command (specified at the command line) performs the backup of a single SA component (with these command options overriding the corresponding parameter settings in the NMD configuration file):

nsrnmdsv –f /space10/home/dmuser/nmd.cfg –B repos01 –M SA –a filestore_01

Example 3

Assuming that all the mandatory parameters are set in the NMD configuration file, the following **nsrnmdsv** command (specified at the command line) performs the incremental backup of a single SA component (assuming that a previous full backup of the SA exists):

nsrnmdsv –f /space10/home/dmuser/nmd.cfg –l incr

Backup of all SA components in a repository

Example 1

Assuming that all the mandatory parameters are set in the NMD configuration file, the following **nsrnmdsv** command (specified at the command line) performs the backup of all the SA components that are part of the repository repos01:

nsrnmdsv –f /space10/home/dmuser/nmd.cfg

For the preceding command to work properly, the following parameters must be set in the NMD configuration file:

- NMDDE_DM_DOCBASE=repos01
- NMD_SCOPE=SA_ALL
- NMD_OBJECT_NAME must be commented out

Example 2

Assuming that all the pathnames and other NetWorker parameters are appropriately set in the NMD configuration file, the following **nsrnmdsv** command (specified at the command line) performs the backup of all the SA components that are part of the repository repos01 (with these command options overriding the corresponding parameter settings in the NMD configuration file):

nsrnmdsv –f /space10/home/dmuser/nmd.cfg –M SA_ALL

The **–a** *SA_name* option is *not* used in the preceding command, as it has no relevance in the context of backing up all the SA components in a repository.

Backup of a distributed storage area

The "Backup of a storage area" section describes backups of SAs on the primary site. The primary site is the host where the EMC Documentum Content Server and the repository both reside. NMD supports a Single-Repository model. A DSA is a collection of component SAs that reside on local and remote hosts.

Preparing for DSA backups

This section discusses setting up the NMD configuration file that holds the key to successful NMD backups (and restores). Setting up this configuration file correctly clears away all obstacles that could cause DSA backups to fail.

NMD configuration file parameters

As described in the "NMD configuration file parameters" section**,** all of the parameters required for successful SA backups are also mandatory for backing up DSA. Additionally, the important credential information must remain the same throughout the DSA setup. NMDDE_DM_USER and NMDDE_DM_PASSWD must be the same at the primary and remote hosts. All of this configuration information should be drafted during installation time to avoid any issues during the configuration of a DSA.

One more parameter that is of utmost importance for DSA backups is NMD_RCS_CFG_FILE. The NMD_RCS_CFG_FILE parameter specifies the name of the NMD configuration file on each remote host that hosts an SA that is part of the same repository. This parameter specifies only the name of the file, and not the full path. This is because during a DSA backup, this file is searched for in the NetWorker res directory (/nsr/res on UNIX/Linux and *installation_path*\nsr\res on Windows). Furthermore, this parameter is specified only at the primary site (because the backup process is initiated from the primary host), and the configuration file on each remote host must share the same name as specified by this parameter.

The NMD_SCOPE variable can be set to **SA,** with NMD_OBJECT_NAME set to the name of the remote SA to be backed up. Alternatively, the scope can be set to SA_ALL to back up all the SAs in the repository (which also causes the backup of all the remote SAs).

The remote configuration file that is specified with NMD_RCS_CFG_FILE has minimal yet critical configuration information. The following parameters are mandatory in the remote configuration file:

- DOCUMENTUM
- DOCUMENTUM_SHARED
- DM_HOME
- CLASSPATH
- JAVA_PATH
- LD_LIBRARY_PATH or LIBPATH or SHLIB_PATH
- PATH
- NMDDE_DM_DOCBASE
- NMDDE_DM_USER
- NMDDE_DM_PASSWD

The parameters that must be specified in the remote configuration file form a small subset of the parameters that must be specified in the primary configuration file. The parameters that are specified in the remote configuration file mostly describe the pathnames, the location of the EMC Documentum Content Server installation on the remote machine, and a reference to the repository to which the configuration file pertains.

NMD command options

The command options for DSA backups are the same as described in the "NMD command options" section under "Backup of a storage area." The command line options **–B** *repository_name*, **–M SA** | **SA_ALL**, **–a** *SA_name*, **–f** *config_filename*, and **–l incr** | **full** still have the same significance in the context of backing up remote SAs.

Far-store lists

It is important to properly configure the far-store lists. A far-store list identifies a SA that is remote to a particular SA but shares the same repository. For each SA in the DSA setup, the rest of the SAs that are part of the DSA setup become remote and must be added to the far-store list. This configuration setting may be done in the Documentum Administrator by logging in to the corresponding repository. This helps the NMD backup process to correctly discover the remote SAs and associate them with the correct hosts.

Backing up remote SAs

In a DSA setup, backing up of remote SAs involves either backing up of a single SA component or all of the SA components in a particular repository (which, of course, involves the backup of the remote SAs too). However, the backup process may be initiated from the primary host *only*. This is because all of the configuration settings of the DSA setup (such as the backup scope, backup level, and NetWorker-specific configuration settings) are available at the primary site only, while the remote configuration files have minimal configuration information as described in the "NMD configuration file parameters" section under "Backup of a distributed storage area."

Before initiating a remote SA backup, ensure that a NetWorker Client resource is configured for each remote host that is part of the DSA setup.

Performing a scheduled backup requires the proper configuration of the primary NMD configuration file and all of the remote NMD configuration files. The backup command **nsrnmdsv –f /space10/home/dmuser/nmd.cfg** is specified in the Backup Command attribute of the Client resource, and the client's backup group is scheduled to run at a particular time, as specified in the "Backup of a storage area" section.

The following examples assume a DSA setup with two Solaris hosts (one primary and one remote). The repository is repos01**,** and the remote configuration file is repos01_rcs.cfg. The repos01_rcs.cfg file has been placed in the /nsr/res directory at the remote host. The name of the remote host is remote**,** and the service name for the remote host is remoteSA (hence the name of the remote SA is fs_rcs_remote_remoteSA).

Example 1

Assuming that all the mandatory parameters are appropriately set in the NMD configuration file and the remote configuration files are properly configured, the following **nsrnmdsv** command (specified at the command line) performs the backup of a single SA component:

nsrnmdsv –f /space10/home/dmuser/nmd.cfg

This example places a lot of trust in the NMD configuration file settings, with every piece of configuration information being read from the configuration files (at the primary and remote hosts). Some of the parameter settings in the primary configuration file are as follows:

- NMDDE_DM_DOCBASE=repos01
- NMD_SCOPE=SA
- NMD_OBJECT_NAME=fs_rcs_remote_remoteSA
- NMD_RCS_CFG_FILE=reps01_rcs.cfg

Example 2
Assuming that all the pathnames and other NetWorker parameters are appropriately set in the NMD configuration file and the remote configuration files are properly configured, the following **nsrnmdsv** command (specified at the command line) performs an incremental backup (let us assume that a prior full backup exists) of a remote SA (the command line options override the corresponding parameter settings in the NMD configuration file at the primary site):

nsrnmdsv –f /space10/home/dmuser/nmd.cfg –B repos01 –M SA –a fs_rcs_remote_remoteSA –l incr

Example 3
Assuming that all the pathnames and other NetWorker parameters are appropriately set in the NMD configuration file and the remote configuration files are properly configured, the following **nsrnmdsv** command (specified at the command line) performs a backup of all the SA components (including the remote SAs configured for the repository) of the repository (the command line options override the corresponding parameter settings in the NMD configuration file at the primary site):

nsrnmdsv –f /space10/home/dmuser/nmd.cfg –B repos01 –M SA_ALL

Backing up remote client indexes

This section describes important information that enables NetWorker disaster recovery of remote SA hosts.

Remote SA backups work in a manner similar to local SA backups. However, since the backup process is initiated at the primary site, the backup index records in the online NetWorker indexes reflect the indexes of the primary host *only*. This is a known limitation. The workaround is to run a small backup of the remote client (for example, backing up only the remote configuration file, specified in the Save Set attribute of the Client resource), so that the remote client indexes are also recorded in the online NetWorker indexes.

FTI backups

A Documentum repository can contain millions of objects. Locating objects containing particular values is a major challenge. Carefully structuring the folders in the repository can make the task somewhat easier by storing related objects together. Full-text indexing (FTI) is a solution to the problem of rapidly locating the information you need in a repository. Full-text indexing is a process that indexes all content files and properties in a repository, creating an index that can be searched rapidly to retrieve objects whose properties or associated content files contain the values for which you are searching.

NMD backs up FTIs by using the **nsrnmdsv** command. A FTI can be backed up with the –M FTI_ALL or the –M FTI –a *fti_name* option of the **nsrnmdsv** command.

On the primary or remote host, either a single FTI can be backed up or all the FTIs in the given repository can be backed up. NMD supports only the full backup level for FTIs.

For a remote FTI backup, NMD_FTI_HOST in the configuration file may be specified with the remote host where the FTI is installed.

Database backups

As mentioned in the "EMC Documentum Content Server overview" section, the database is the heart of the EMC Documentum Content Server architecture as content attributes are stored in tables of a relational database. Backing up of the database is an important task. Documentum/NMD supports the following databases:

- SQL Server
- Oracle
- DB2

- Sybase

NetWorker has plug-ins called NetWorker Modules that facilitate the backup of these databases, such as:

- NetWorker Module for Microsoft SQL Server
- NetWorker Module for Oracle
- NetWorker Module for IBM DB2
- NetWorker Module for Sybase

NMD strongly recommends the use of NetWorker Modules to back up the underlying databases as each of these modules are functionality rich and can be leveraged for backing up the databases.

NMD on its own does *not* back up any databases. Instead, NMD relies on user-defined scripts (with or without NetWorker Modules) to back up the databases. NMD just provides an option to invoke the scripts from the NMD configuration file.

NMD provides the following features:

- Consistency of backups when all the repository components are backed up together.
- A single command line option to back up all the components of a repository.
- Independence from changes in the databases when the NetWorker Modules are used for the database backups. For example, if there is change in Oracle, a newer version of NetWorker Module for Oracle supports those changes. NMD uses the new version of the module to support the new features, which provides modularity to the overall solution.

The following NMD configuration file parameters are pertinent to database backups:

- NMD_DB_FULL_BACKUP_CMD − Specifies the script to use for full backup of the database.
- NMD_DB_LOG_BACKUP_CMD − Specifies the script to use for database log backups.
- NMD_DB_INCR_BACKUP_CMD − Specifies the script to use for incremental backup of the database.

The following parameter settings specify the scope of the database backups:

- NMD_SCOPE=DB − Invokes the script specified by NMD_DB_FULL_BACKUP_CMD or NMD_DB_INCR_BACKUP_CMD. Apart from this, if the NMD_DB_LOG_BACKUP_CMD parameter is set, NMD also invokes the script specified by this parameter.
- NMD_SCOPE=DB_LOG − Invokes the script specified by NMD_DB_LOG_BACKUP_CMD only.

If you specify the absolute path for the parameters NMD_DB_FULL_BACKUP_CMD, NMD_DB_INCR_BACKUP_CMD, and NMD_DB_LOG_BACKUP_CMD, the scripts are searched locally. If you have a remote database, specify just the base name of the script (not the absolute path) and place these scripts in the /nsr/bin or \nsr\bin directory of the remote host.

NMD interprets the success or failure of the database backups by using the return status returned from the backup scripts. You must set the return status in the scripts. Following are example scripts that set the return status on Windows and Unix platforms:

Windows platform

```
call C:\legato\nsr\bin\nsrnmo.bat -s sunflower.blrsql.com -g NMO -m hana.blrsql.com -N
DCB:\sheleshrepos\database RMAN:C:\scripts\rman2.txt
set rc=%errorlevel%
```

```
if "%rc%" == "0" (
echo ##completed
)exit %rc%
```

UNIX platform

```
#!/bin/sh
/usr/sbin/nsrnmo -s marsh.blrsql.com -m marsh.blrsql.com -g NMO /nmoscripts/rman2.txt
rc=$?
if [ $rc = 0]; then
   echo "##completed"
fi
exit $rc
```

Restore of the database should be done by using the underlying module that backed up the database. NMD does not provide an option to recover the database. A user must follow the restore steps to maintain the consistency of the Documentum system.

ICF backups

If the EMC Documentum Content Server ICF is destroyed or corrupted, having a backup of the ICF could prove to be of great help in restoring the configuration. Not only will it prevent sleepless nights, it will also save a lot of administrative work involved in restoring or rebuilding the setup.

Preparing for ICF backups

ICF backups may be initiated in two ways:

- Manually executing the **nsrnmdsv** command
- Scheduling a backup using NetWorker, as mentioned in "Backup of a storage area"

Setting up the environment for ICF backups requires the configuration of certain parameters in the NMD configuration file. These parameters are listed in the "NMD configuration file parameters*"* section. However, the NMD_OBJECT_NAME and NSR_BACKUP_LEVEL parameters do not apply to ICF backups. ICF backups need a different parameter setting (instead of NMD_OBJECT_NAME**)** to identify which files and directories need to be backed up. This will be explained shortly.

ICF backups are performed always at the full backup level and hence the NSR_BACKUP_LEVEL parameter has no significance. Irrespective of what backup level is specified with NSR_BACKUP_LEVEL in the NMD configuration file or with the –l command line option, the ICF is backed up at the full backup level.

ICF backups are not enabled by default, and hence they need to be enabled before starting an ICF backup. The reason for this is because the ICF does not change as often as other components in a Documentum setup, and therefore backing up ICF on a daily basis or at short intervals is unnecessary.

Enable ICF backups by one of the following methods:

- Set the NMD_ICF_INCLUDED parameter to **yes** if the ICF backup is to be part of the entire repository backup.
- Set NMD_SCOPE to **ICF** or specify –M ICF at the **nsrnmdsv** command line if the ICF backup is to be performed as a separate backup.

How does **nsrnmdsv** know what to back up as part of an ICF backup? The user decides what you consider to be important, either installation and configuration files or directories. Once identified, specify the files and directories that must be part of the ICF backup with the NMD_ICF_SUBDIRS parameter, with each path separated by a comma (do not include any spaces around the comma separator). The sample NMD

configuration file that is installed in the NetWorker bin directory after an NMD installation has the NMD_ICF_SUBDIRS parameter set as follows:

NMD_ICF_SUBDIRS=/db/dctm/dba,/db/dctm/product/5.3SP3,/db/dctm/dmcl.ini

This line is extracted from the sample NMD configuration file installed on an HP-UX machine that has EMC Documentum Content Server installed. The directories mentioned previously may be considered as mandatory while you perform ICF backups. Even though we consider only the EMC Documentum Content Server installation and configuration files for ICF backups, you may include the NMD configuration file as part of the backup as it is also a very important piece of configuration information.

Backing up the ICF

The following examples consider that EMC Documentum Content Server has been installed on a machine running on an HP-UX OS platform. A repository has been configured and is named repos01.

Example 1

Assuming that all the mandatory parameters are set in the NMD configuration file, the following **nsrnmdsv** command (specified at the command line) performs the backup of ICF:

nsrnmdsv –f /space10/home/dmuser/nmd.cfg

This command works properly if the following configuration setting is made in the NMD configuration file (specified by the **–f** option in the above command):

- NMDDE_DM_DOCBASE=repos01
- NMD_SCOPE=ICF
- NMD_OBJECT_NAME is commented out
- NMD_ICF_SUBDIRS=/db/dctm/dba,/db/dctm/product/5.3SP3,/db/dctm/dmcl.ini

You may add more directories and files as you feel necessary to the list of directories and files specified with the NMD_ICF_SUBDIRS parameter.

In this example, if NMD_SCOPE is set to **ALL**, then NMD_ICF_INCLUDED must be set to **yes** while the rest of the configuration settings remain the same. This NMD_ICF_INCLUDED setting applies whether you specify all the configuration settings in the NMD configuration file or through the command line (for example, you specify the **–M ALL** option).

Example 2

Assuming that all the pathnames and other NetWorker parameters are appropriately set in the NMD configuration file, the following **nsrnmdsv** command (specified at the command line) performs the backup of the ICF (the command line options override the corresponding configuration settings in the NMD configuration file):

nsrnmdsv –f /space10/home/dmuser/nmd.cfg –B repos01 –M ICF

In this example, NMD_ICF_SUBDIRS must be set in the NMD configuration file.

Remote storage node for multihost configurations

EMC Documentum Content Server supports remote configurations or multihost configurations. For instance, the database can be remote or local to the primary host or DSAs, where SAs have replicated contents at primary and remote sites. NMD can leverage the remote storage node functionality of NetWorker whereby a user can set the affinity of the client to a particular SA.

A remote storage node gives flexibility to direct the backup data to go to a particular storage node, which indirectly means to a set of devices belonging to that storage node. Figure 5 illustrates the settings for specifying the affinity for a storage node.

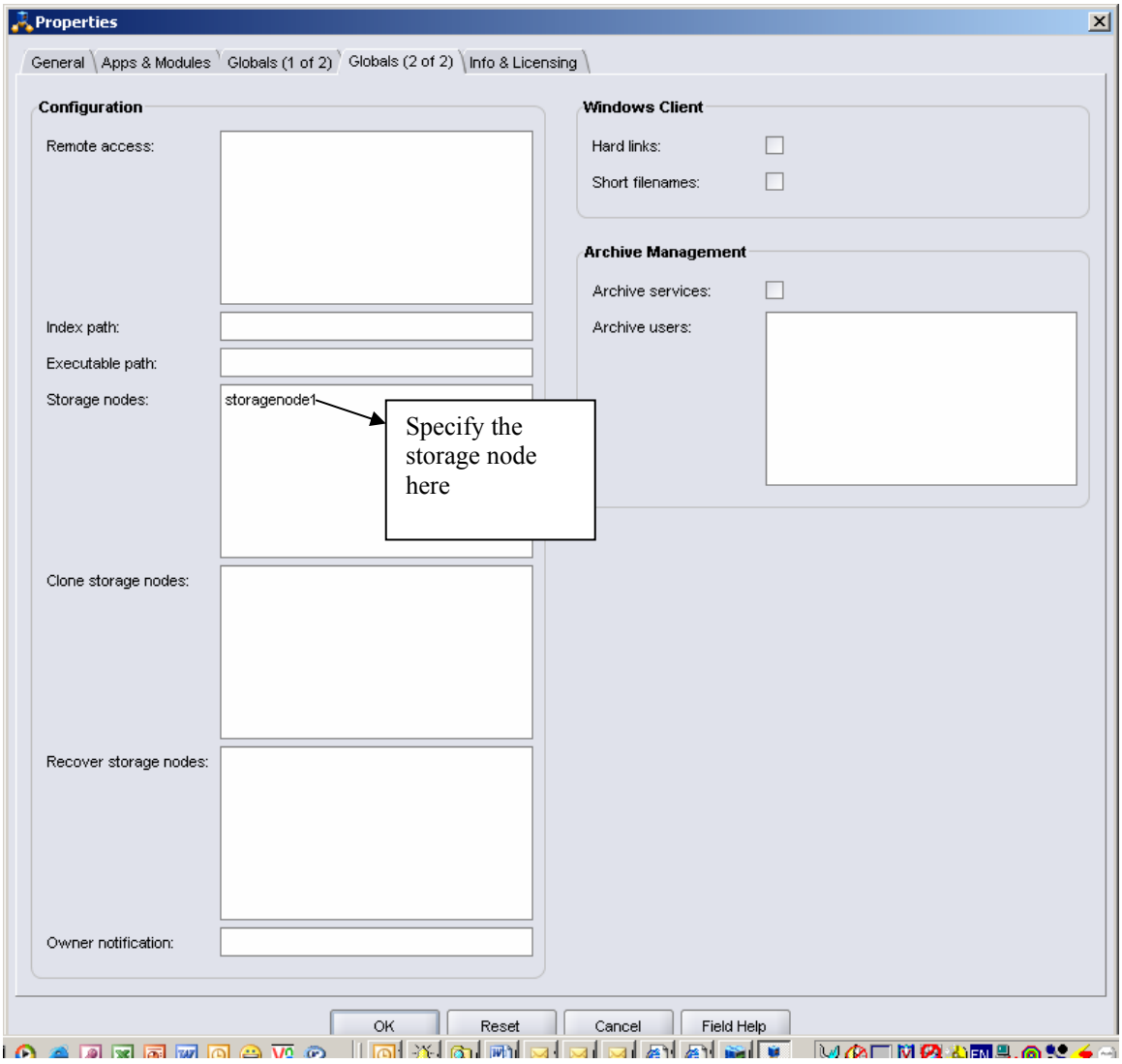

Figure 5. Specifying the affinity of a storage node

Using this functionality, a user can set the affinity for different storage nodes. For instance, the remote host that has the remote SA can set the storage node affinity to itself (if it has the storage node installed) or to a different storage node, nearer to itself, in order to avert network traffic. By default, the storage node is set to **nsrserverhost**, which means that the data is backed up to devices that are connected directly to the NetWorker server.

Configure a remote device by using the following convention:
rd=storagenode1:\\.\Tape0 or rd=storgaenode1:/networkerdevice

where storagenode1 is the hostname of the remote storage node

Figure 6 shows a remote device configured on a NetWorker server.

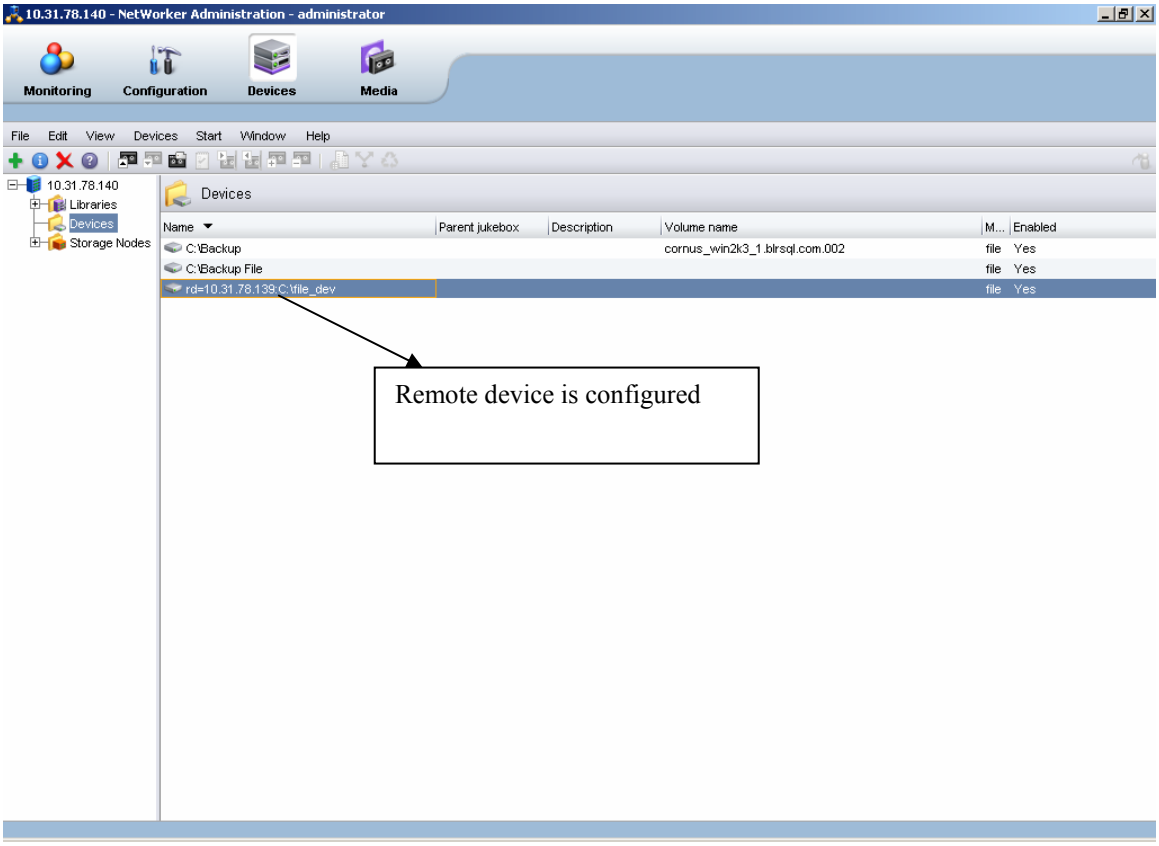

Figure 6. Configuring a remote device

The same concept can be extended to remote FTI or remote DB backups where a user can set the client affinity to a particular storage node.

Browsing NMD backups

The backup information is stored in the online NetWorker indexes, which reside on the NetWorker server host. Browsing for NMD backups is a feature that may be used either to validate a backup or to gather useful information about a backup before planning for an NMD recovery. Browse the backup indexes to gather useful information about backups before performing a recovery of any Documentum system components.

NMD offers a versatile backup information browsing command, **nsrnmdnx**, which may be used to browse backup information on the previous NMD backup sessions. The highest level at which backup information can be browsed is the repository level. From the repository level, it is possible to drill down further to find backup information about a single component of a repository, like a single SA component, a single FTI component, and so on. The **nsrnmdnx** command can also provide information about a backup of a component that was backed up at a particular level.

Browsing NMD backup information

Unlike the commands **nsrnmdsv** and **nsrnmdrs**, the **nsrnmdnx** command does not rely on any NMD configuration file. Obtain information about backups by specifying the appropriate command line options with the **nsrnmdnx** command. The **nsrnmdnx** command provides you with a stronghold on the granularity in the browsing of the indexes.

The **nsrnmdnx** command has a versatile set of options as shown:

nsrnmdnx -B *repos_name* [**-s** *server*] [**-c** *client*] [**-n** *count*] **-T** *tag* [**-D** *debug_level*] **-v**
nsrnmdnx -B *repos_name* [**-s** *server*] [**-c** *client*] [**-n** *count*] **-M** *backup_scope_opts* [**-l full**|**incr**]
[**-L** *lower_bound_time*] [**-U** *upper_bound_time*] [**-D** *debug_level*] **-v**

backup_scope_opts:= **-M SA** [**-a** *SA_name*] | **-M FTI** [**-a** *FTI_name*] | **-M ALL** | **-M SA_ALL** | **-M FTI_ALL** | **-M ICF** | **-M DB** | **-M DB_LOG**

Note that similar to the **–f** option that is mandatory for the commands **nsrnmdsv** and **nsrnmdrs**, the **–B repos_name** option is mandatory for **nsrnmdnx**.

The following examples show how to browse NMD backup information by using the **nsrnmdnx** command with certain command line options. All the examples assume a machine with the HP-UX OS and EMC Documentum Content Server 5.3 SP3 installed. A repository has been configured and is named repos01. Full backups with scope SA, SA_ALL, and ICF, and an incremental backup of the SA component backed up with the SA scope have been performed, in that order. The SA that was backed up is filestore_01.

Example 1

The following command browses for all the backups that were performed for the repos01 repository and displays the results on the console:

nsrnmdnx –B repos01

This command displays onscreen details of the backups performed on the repository. The output is organized into sections with each section pertaining to a *single backup session*. The output of this command yields four records, one for the full backup of the SA component filestore_01, one for the full backup of all the SA components, one for the backup of ICF, and the last one for the incremental backup of the SA component filestore_01. The record for the most recent backup appears first on the screen.

Example 2

The following command browses for and displays information on all the backups of the SA component filestore_01:

nsrnmdnx –B repos01 –M SA –a filestore_01

This command displays two records, one for the full backup and the other for the incremental backup. The information displayed for these records includes the timestamp, the level of backup, the starting and ending time of the SA backup, and the location of the directory to look into for the corresponding file report.

Example 3

The following command browses for the backup information for the SA component filestore_01 that was backed up at the incremental backup level:

nsrnmdnx –B repos01 –M SA –a filestore_01 –l incr

Example 4

The following command browses for NMD backup information on the backup of installation and configuration files:

nsrnmdsv –B repos01 –M ICF

Example 5

Consider a different scenario this time. A number of backups of a SA component filestore_01 were performed over a week’s time. The NetWorker server is located on a remote machine named remote (not

on the machine where you wish to run the **nsrnmdnx** command). The following command displays the backup information of all the backups of filestore_01 that were performed since yesterday:

nsrnmdnx –B repos01 –M SA –a filestore_01 –s remote –L yesterday

In this example, the **–L** command line option specifies the lower-bound on the time, from when to start searching for NMD backup information.

How backup information is browsed

Browsing of the NMD indexes involves the following process interactions:

1. The browse command invokes the nsrnmdnx process.

2. The nsrnmdnx process invokes the nsrindexd process to search the online NetWorker indexes for the backup information records that match the query request.

3. The nsrindexd process returns the information from the backup index search to the nsrnmdnx process.

4. The nsrnmdnx process prints the backup index information to the screen.

How NMD restores data

During a backup, the NetWorker server adds an entry for each backup save set in the online client file index (or simply, CFI) and records the location of the data in the media database. These entries provide information required to restore every Documentum data object backed up.

The CFI entry is maintained until the *browse* policy expires, and the media database entry is maintained until the *retention* policy expires.

NMD uses the client file index entries to restore Documentum backup save sets. An NMD backup includes two types of save sets:

- Save sets that contain the backed-up file data
- Save sets that contain metadata

Both the information about the file data save set and metadata are required to restore the backed-up data and browse the backup index.

Use the **mminfo** command to check for the metadata and data that were backed up:

```
cornussles92:/ # mminfo
 volume        client       date      size   level  name
cornussles92.001 cornussles92 09/20/07 5 KB    full  DCB_repos01:metadata
cornussles92.001 cornussles92 09/20/07 1920 KB full DCB_repos01:SA_filestore_01
```

Figure 7 shows the restore workflow for NMD.

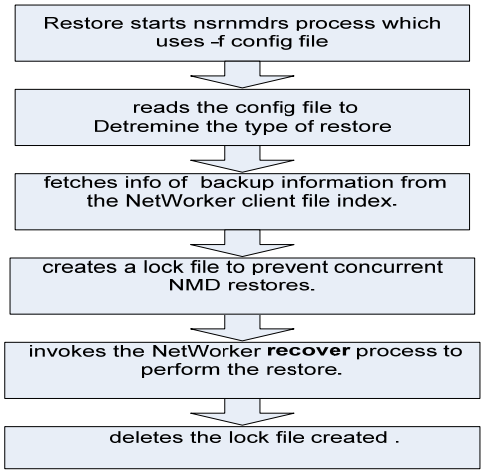

Figure 7. Restore workflow for NMD

NMD can restore the following components:

- FTI
- SA
- ICF
- Content files, and their associated renditions

The **nsrnmdrs** command may be specified at the command line to initiate the NMD restore.

SA recovery

SAs may be recovered through the **nsrnmdrs** or **nsrnmdcr** command.

SA recovery through nsrnmdrs

The nsrnmdrs command browses the NMD backup index and restores the selected component from a Documentum SA, FTI, or ICF backup.

A specific SA or all SAs backed up can be restored at once by using the appropriate options with the **nsrnmdrs** command.

Figure 8 shows the interaction between the Documentum server, NetWorker server, and the storage medium.

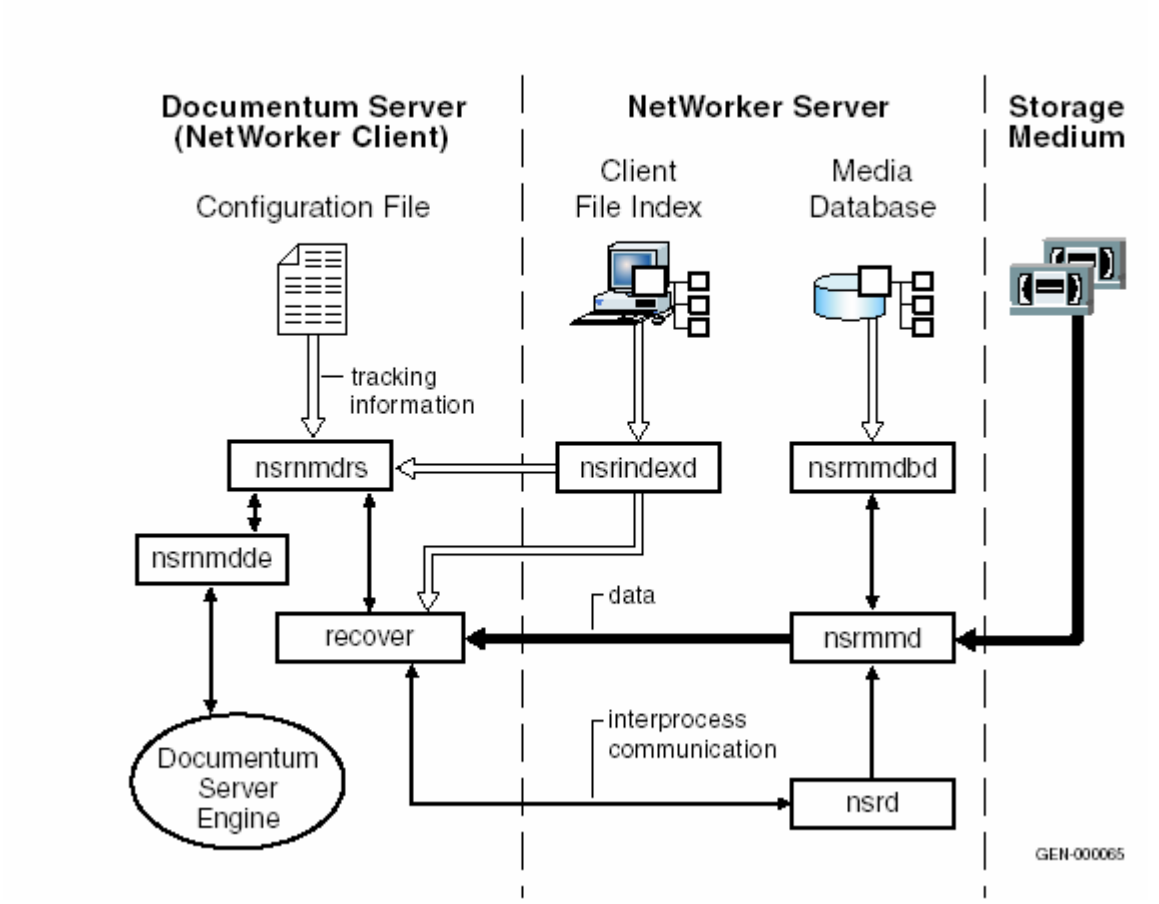

Figure 8. Process interaction in an NMD restore

The **-C SA -a object_name** option specifies that the restore process searches for a SA backup.

When **-a** option is set	The restore searches for a backup of scope SA
When **-a** option is not set	Restore searches for a backup of scope ALL or SA_ALL

A backed-up SA may be restored to a user-specified directory (relocation directory) with the **–d** *relocation_dir* option, or the relocation directory may be specified in the NMD configuration file with the NSR_RELOCATION parameter. During a restore or disaster recovery of a FTI, SA, or ICF, if an upper-level parent directory of the component is missing, the directory is automatically re-created and the ownership and permissions of the directory are changed to the user who performs the restore, such as the root user on UNIX. Hence, **nsrnmdrs** gives the user flexibility to relocate the data to be restored to a different location from the original location of the data.

If the **–d** option is not specified or the NSR_RELOCATION parameter is commented out, the data is restored to its original location. Typically, during a restore to an original location, a user must temporarily put the respective repository component offline.

The **nsrnmdrs** command syntax and options are as follows:

nsrnmdrs -f config_filename [**-B** repository_name]
-M ALL|**-C FTI** [**-a** object_name]|**-C ICF**|**-C SA** [**-a** object_name]
[**-T** backup_tag|**-L** lower_bound_time|**-U** upper_bound_time]
[**-c** client_name] [**-D** debug_level] [**-d** relocation_path]

[**-i** {**n**|**N**|**y**|**Y**|**r**|**R**}] [**-P**] [**-p** parallelism] [**-s** server_name] [**-?**]

Sample nsrnmdrs commands for different types of restores

The following **nsrnmdrs** command restores the latest backup of the SA filestore_01 belonging to the repository repos01:
nsrnmdrs -B repos01 -f /space10/home/dmuser/nmd.cfg -C SA –a filestore_01

The following **nsrnmdrs** command performs a relocated restore of all SAs from the latest backup that occurred before today:
nsrnmdrs -B repos01 -f /space10/home/dmuser/nmd.cfg -C SA -U "today"
-d /reloc

SA recovery through nsrnmdcr

NMD software supports browsing of Documentum file reports that are optionally generated at the end of SA backups. The **nsrnmdcr** command is used for this purpose. The command displays a list of content files (listed in the file reports) that match the specified search criteria. Optionally, the command can also restore the content files found during the browse operation.

The **nsrnmdcr** command syntax and options are as follows:

nsrnmdcr -B repository_name [**-A** SA_name]
[**-E** file_report_path | **-P** file_report_directory] [**-T** file_report_time]
[**-N** max_entries_returned] [**-F** results_filepath] [**-V**] [**-O**]
[**-R** [**-C** nsr_client_name] [**-D** relocation_directory] [**-I** {**n**|**N**|**y**|**Y**|**r**|**R**}]
[**-S** nsr_server_name]] [**-c** client_name] [**-d** document_name]
[**-f** folder_name] [**-m** document_format] [**-o** original_document_name]
[**-r** {**org**|**mod**}] [**-t** document_save_time] [**-u** username]
[**-v** document_version]

The **nsrnmdcr** command helps the user narrow down the search to a particular SA with the –**A** *SA_name* option. With the –**E** option, the **nsrnmdcr** command searches for information in a particular file report, rather than through all the file reports in a particular directory that match the specified search criteria. With the –**P** option, the **nsrnmdcr** command searches for all relevant file reports in the specified file report directory.

Granular recovery

Granular recovery can be done to search for a particular file by either specifying the format of document, name of file or document, and so on, and then recover the file if found during the browsing process. Just two of the ways to specify the commands follow:

- Recovery of files with the **–d document_name** option specifies the document name to search for in the file reports. All document names that contain the string document_name are displayed. Wildcards or regular expressions should not be specified with the **-d** option. The search looks for document names containing the (case-insensitive) string "mobile"

  nsrnmdcr -B repos01 -A filestore_01 -P "F:\Documentum\stores\Filereports\current"
  -d "mobile

- The filename of the physical file on the host may be specified for the document to be searched for in the file reports using **-o original_document_name,** which will help in faster reporting and restoring.

Here the search looks for all the documents of text format. It adds the wildcard character *, before and after the format specified by the user.

nsrnmdcr -B repos01 -A filestore_01 -P "F:\Documentum\stores\Filereports\current"

-o txt

The –R option may be specified with the **nsrnmdcr** command to indicate that a restore option must be provided upon browsing for content files. User may also specify a relocation directory with the –D option to relocate the content.

Relative pathnames are interpreted relative to the current working directory. If the **-D** option is not specified, a content file is restored to its original location.

Importance of file reports

While performing SA backups, NMD generates file reports in a text format that has the list of individual files that are backed up as part of the SA backup.

File reports help in quickly searching the files from the backup while querying the backup taken. NMD through the **nsrnmdcr**, **nsrnmdrs** command searches the file report to query SA backups and optionally does the redirected restore if the user opts for restoring the data.

Sample nsrnmdcr commands for different types of restores

The following **nsrnmdcr** command searches file reports for the SA filestore_01 of the repository repos01. The file reports searched are those created on October 04, 2007. The search looks for document names containing the (case-insensitive) string "mobile", with the folder name mobilitydocs and version number 1.0. The documents were checked in from the client waterloo.

nsrnmdcr -B repos01 -A filestore_01 –P "F:\Documentum\stores\Filereports\current"
-T "20071004" -V -d "mobile" -f "mobilitydocs" -v "1\.0"

FTI recovery

We have covered the backups of the FTI component in previous sections. In this section, we will cover the restore of the FTI component.

Recovering the FTIs

The scope has to be mentioned explicitly at the command line either with the –C option or with the –M option to the **nsrnmdrs** command.

The following example assumes that EMC Documentum Content Server is installed on a machine running an HP-UX OS with one repository named repos01 configured. The FTI name is repos01_ftindex_01.

Example 1

Assuming that all the pathnames and other NetWorker parameters are appropriately set in the NMD configuration file, the following **nsrnmdrs** command (specified at the command line) performs FTI recovery to a relocated directory /reloc as specified with the NSR_RELOCATION parameter:

nsrnmdrs –f /space10/home/dmuser/nmd.cfg –C FTI

or

nsrnmdrs –f /space10/home/dmuser/nmd.cfg –C FTI –a repos01_ftindex_01

The following parameters are set in the NMD configuration file:

- NMDDE_DM_DOCBASE=repos01
- NSR_RELOCATION=/reloc

The relocation directory may also be specified with the –d command line option to the **nsrnmdrs** command as follows:

nsrnmdrs –f /space10/home/dmuser/nmd.cfg –C FTI –d /reloc

or

nsrnmdrs –f /space10/home/dmuser/nmd.cfg –C FTI –a repos01_ftindex_01 –d /reloc

ICF recovery

An unfortunate scenario and the one that we must all live with is that of data loss. In a scenario wherein EMC Documentum Content Server installation and configuration files were destroyed or if certain data has been corrupted, a backup of ICF will provide relief to those responsible for rebuilding or restoring the setup. Backing up the installation and configuration files at regular intervals may be considered to be a good backup policy. This section will discuss ICF recovery procedures.

Preparing for ICF recovery

Recovery of ICF can be achieved from the command line using the **nsrnmdrs** command. Unlike recovery of SA components, ICF recovery does not have the file-by-file recovery option. The entire ICF must be recovered. A good recovery practice could be to recover the files into a relocated directory and then move the contents to their actual location as the need may be. This could prove helpful in situations wherein only a few files are corrupt or are missing, and you need to restore these files (provided you were able to identify which files are actually missing or are corrupted).

While performing a full repository backup, that is, NMD_SCOPE=ALL, you may specify that ICF must also be backed up by setting NMD_ICF_INCLUDED=yes and specifying the files and directories to be backed with the NMD_ICF_SUBDIRS parameter. But a full repository recovery performed by specifying the **nsrnmdrs** command with the –M ALL option at the command line will recover only the FTI and SA components, that is., ICF will not be recovered. ICF must be recovered exclusively with the –C ICF option to the **nsrnmdrs** command.

Recovering the installation and configuration files

Setting the scope in the NMD configuration file has no significance in NMD recoveries. The scope has to be mentioned explicitly at the command line either with the –C option to the **nsrnmdrs** command.

The following example assumes that EMC Documentum Content Server is installed on a machine running on the HP-UX OS platform with one repository named repos01 configured.

Example 1

Assuming that all the pathnames and other NetWorker parameters are appropriately set in the NMD configuration file, the following **nsrnmdrs** command (specified at the CLI) will perform an ICF recovery to a relocated directory:

nsrnmdrs –f /space10/home/dmuser/nmd.cfg –C ICF

The following parameters are set in the NMD configuration file:

- NMDDE_DM_DOCBASE=repos01
- NSR_RELOCATION=/reloc

The relocation directory may also be specified with the –d command line option to the **nsrnmdrs** command as follows:

nsrnmdrs –f /space10/home/dmuser/nmd.cfg –C ICF –d /reloc

This command after execution will recover all the files and directories that were specified with the **NMD_ICF_SUBDIRS** command during the point in time of the ICF backup. Since relocation has been used the ICF will be relocated to a directory named ICF under the relocation directory /reloc. Take care to validate that the recovery user has write permissions for the directory to which the recovery is performed. Moreover, once you shift the files to their original locations make sure to change the permissions of the files and directories accordingly to the installation owner permissions.

SA backup and recovery using SnapImage

The NetWorker SnapImage module is a snapshot-based backup module that facilitates fast backups and restores of high-density file systems. SnapImage (SI) works at its best and produces high performance on backups and restores when you back up high-density file systems that contain millions of small files, with sizes ranging from 1 KB to 100 KB. SnapImage uses the industry-standard snapshot-based backup protocol called NDMP (Network Data Management Protocol) to perform backup and recovery.

The following points summarize the features of the NetWorker SI module:

- Full-image, block-level snapshot backups and restores, allowing large amounts of high-density, file system data to be backed up and restored quickly
- Snapshot backups that are faster than traditional file-based backups under appropriate conditions
- Restores of specific files and directories at the volume level

SI will need a raw device to be configured to cache backup data temporarily. SI uses snapshot technology that creates a frozen or snapshot view of the file systems to be backed up. Any changes to blocks on the file system that occur during the backup are intercepted by the *write intercept driver*. Once intercepted, the changed blocks are saved to cache. When SI reaches a block that changed during the current backup session, it knows the block must be read from cache instead of the original block location. Blocks that change after being backed up during the current session are updated as the changes occur.

Using the SnapImage module with NMD

The NetWorker SI module works in conjunction with NMD to provide faster backups and restores of SAs that are hosted on high-density file systems. The SI module is installed on the host where the EMC Documentum Content Server is installed or where remote SA is configured.

Since SI is used to perform block-level file system backups the entire file system that contains the SA(s) will be backed, irrespective of the fact that we wish to back up a single SA or multiple SAs.

Certain conditions need to be met before initiating an SI backup. The following points summarize some of these conditions:

- The total number of content files for all SAs on each file system or drive is greater than approximately 2 million, and the average content file size is less than approximately 150 KB.
- The SAs on each file system make up more than 75 percent of the file system's total used space, and more than 75 percent of files on the file system belong to these SAs. In short, there must be little non-SA data on the SA file systems.
- The Oracle control files and online redo logs are not located on a disk that is backed up by SI, as it may cause the termination of an Oracle instance on the same file system.
- The SA file systems have little disk write activity that cannot be redirected to a nonstorage area file system.
- Last but not the least, you have the expertise to configure and maintain the SI environment.

Backup and restore using SnapImage

SI with NMD may be used to perform quick backups of storage components. Hence, if the SI module is installed on a EMC Documentum Content Server host, set the NMD_SCOPE parameter in the NMD configuration file to **SA**, **SA_ALL** or **ALL**, or use the same options with the –M option of the **nsrnmdsv** command at the command line. Moreover, there must be an NDMP client configured on the host on which the backup is initiated. To configure an NDMP client, select the NDMP checkbox in the client resource. To perform a scheduled backup, add the following name-value pairs in the Application Information textbox located in the client resource:

- DDIMAGE=n
- DIRECT=y
- DSA=Y (if the setup has a DSA configured)
- HIST=y
- REMOTE=Y

Also specify the Remote user and Password fields in the client resource if performing an SI backup of a SA on a remote client.

The NMD_USE_SNAPIMAGE parameter in the NMD configuration file must be set to **yes** to configure SI-based SA backups. Also set the NSR_SNAPIMAGE_DATA_VOLUME_POOL parameter in the NMD configuration file to specify the volume pool to be used to direct the data backed up by SI.

To run a scheduled backup, enter the following command in the Backup Command attribute of the client resource:

nsrnmdsv –f /space10/home/dmuser/nmd.cfg –I primary

where primary is the name of the client on which to initiate the SI backup. (On a DSA setup to perform an SI backup of a remote SA, specify the name of the remote client on which the remote SA resides). The –I option that has been used above is used only when specifying the command for scheduled backups. While performing a manual backup, this option is not to be specified. The only difference between performing a SA backup using SI and without SI is the way content is backed up. With SI the entire file system hosting the SA will be backed up, while a backup without the SI module will perform the backup of the SA component only. Irrespective of the method chosen, file reports will still be generated (provided you have opted for it).

The restore of an SA backed up using SI is done the same way as it is would be done normally, that is, either using the **nsrnmdrs** command or by performing a file-by-file restore using the **nsrnmdcr** command. However, an important point to keep in mind is that a SA component (or components) that was backed up using SI will always be restored using SI.

Conclusion

NMD provides a complete solution for backup of the EMC Documentum Content Server. The solution ensures the consistency of the backup images by following the rules to back up different components in a specific manner. Users can also use the various NetWorker Modules (like SnapImage, NetWorker Modules for Oracle, Sybase, DB2, Microsoft SQL Server) because NMD provides an integrated solution with these modules.

References

- *Documentum Backup using NetWorker Module for Documentum* customer presentation
- *EMC Documentum Content Server Version 6 Administration Guide*
- *EMC Documentum Content Server Version 6 Installation Guide*

- *Documentum 5 Architecture – A Technical Overview*
- *EMC Documentum Content Server Version 6 Distributed Configuration Guide*
- *EMC NetWorker Module for Documentum Release 1.2 Multiplatform Version Administration Guide*
- *EMC NetWorker Module for Documentum Release 1.2 Multiplatform Version Installation Guide*
- *EMC NetWorker Module for Documentum Release 1.2 Multiplatform Version Best Practices Guide*
- *EMC NetWorker Module for Documentum Release 1.2 Multiplatform Version Command Reference Guide*
- *EMC NetWorker Module for Documentum Release 1.2 Multiplatform Version Release Notes*