Module 4: Performing Backups

Upon completion of this module, you should be able to:

- Perform server-initiated backups.
- Describe the data flow in a server-initiated backup.
- Explain the relationship among the client attributes and certain resources.
- Configure schedule, policy, group, directive and client resources.
- Use backup commands other than `save`.
- Perform client-initiated backups.
- Describe how virtual clients and database applications can be backed up in NetWorker.

EMC²

This module focuses on ways of performing backups, including starting backups manually from the NetWorker server and from a NetWorker client. We look at the resources relevant to configuring a NetWorker client and then schedule a backup to occur automatically. Then, we look at how to customize the backup environment.

Module 4: Performing Backups

Lesson 1: Before Performing a Backup

During this lesson the following topic is covered:
- Minimum requirements for performing a backup

EMC²

This lesson covers what must be in place, at a minimum, before performing a backup.

Performing NetWorker Backups

Client-initiated Backup	Server-initiated Backup
Manual	Scheduled (Group resource)
A user on the NetWorker client initiates the backup of selected files.	The NetWorker server software requests a backup of the client.
Performed using a GUI or the command-line.	Can also be started manually using a GUI or the command-line.
Manual level backups	Full, incremental, synthetic full, level 1-9 backups

EMC²

A **backup** is the process of copying data and storing the copy in another location. The destination can be another directory on the same system, another server, or some other type of storage media.

EMC NetWorker allows you to perform two types of backups:

- A *client-initiated backup* is started from a NetWorker client.
- A *server-initiated backup* is started from the NetWorker server and sends a backup request to one or more NetWorker clients.

A client-initiated backup is a manual process that the user must perform. A server-initiated backup is usually configured to start automatically but may also be started manually, either from NetWorker Administration or the command line.

Server-initiated backups are the preferred option for performing day to day backups as well as ad-hoc backups. It is recommended to reserve client-initiated backups for specific use cases only as needed.

Before Performing a Backup

- During NetWorker server installation, a client resource and a storage node resource are automatically created for the NetWorker server.

- Before running a backup, at a minimum, the administrator must:

  - Install the client software on and create a client resource for the client to be backed up.

    - Create a NetWorker device resource:
      - Drive within a tape library (also known as an autochanger or jukebox).
      - Stand-alone device like Advanced File Type Device (AFTD) or DD Boost

   - Label and mount a volume for the device.

EMC²

Before performing a backup, at a minimum, a client resource must be created for the client being backed up, a device resource must be created, and a volume must be labeled for the device.

During the NetWorker server installation, the NetWorker server is automatically configured as a client of itself. Also, a storage node resource is automatically configured for the NetWorker server. When backing up any other client, the NetWorker client software must be installed on the client, and a client resource must be configured.

The backup device can be a drive within a library or a standalone drive, such as a disk storage device. After a volume is labeled, it must be mounted before NetWorker can use it for backups.

These tasks are accomplished using the NetWorker Administration Interface in the NetWorker Management Console.

Create a Client Resource for a Backup Client
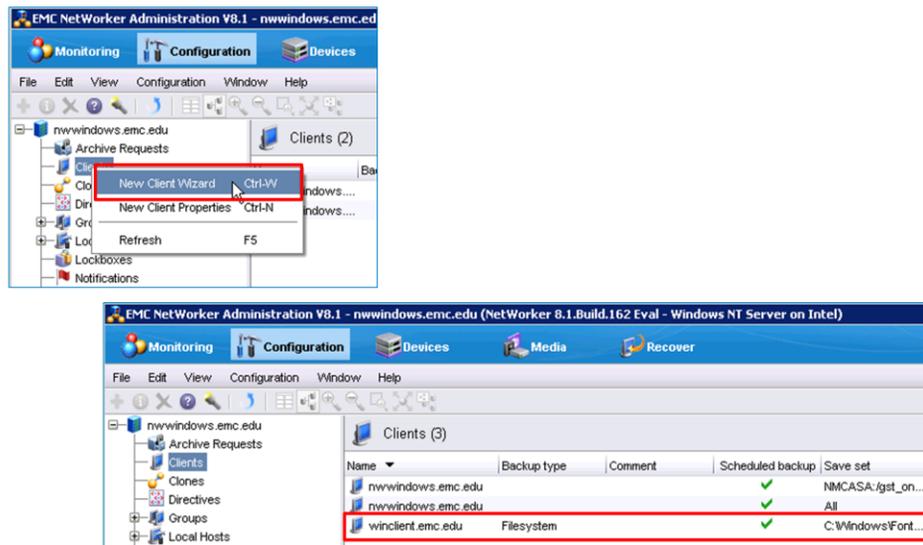
As an example, let's create the resources necessary to perform a backup of a NetWorker client. In this way, we will verify our installation configuration.

First, we create a client resource for the backup client. To quickly create a client resource, use the **New Client** wizard accessed from the **NetWorker Administration Configuration** window. The wizard will ask for the client name and will supply default values for the several attributes in the client resource. The slide lists the client resource created for a client named winclient.emc.edu. Also notice the client resources that were created for the NetWorker server, nwwindows.emc.edu, during installation.

It is important to note that prior to configuring the client in using the New Client Wizard, we first installed the NetWorker client software on the client. Client installation was not illustrated in this slide.

Creating an Advanced File Type Device (AFTD)
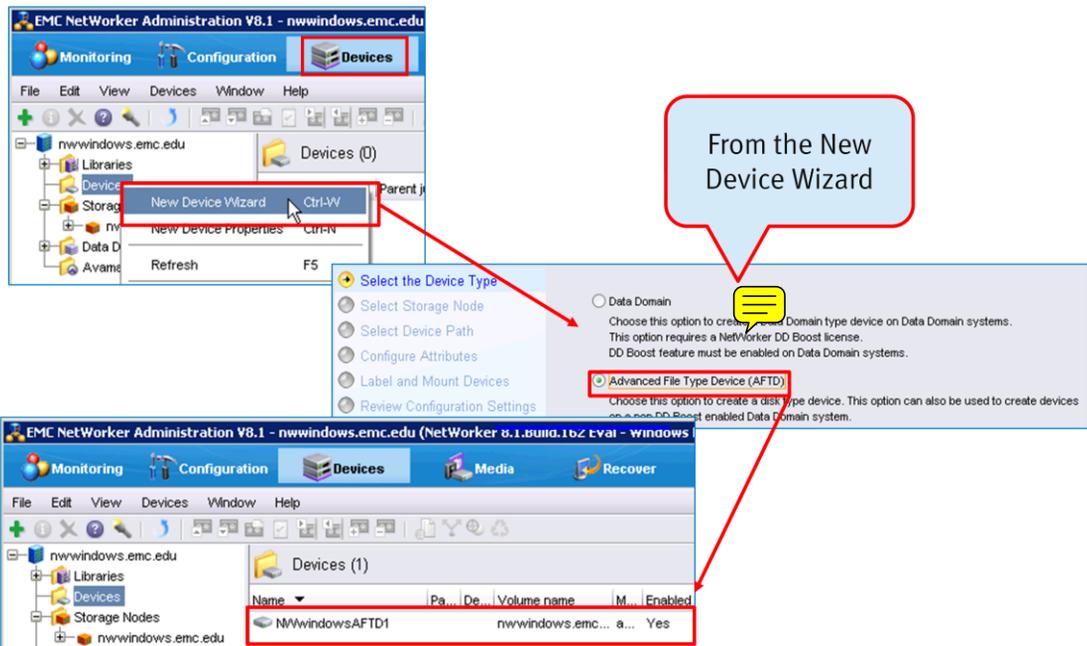
Module 4: Performing Backups   6

NetWorker supports several types of disk storage devices. One of the most commonly used NetWorker disk storage devices is the Advanced File Type Device or AFTD. The AFTD supports concurrent backup and restore operations. An AFTD uses a directory within a file system as its volume. Each save set directed to the device is written to a separate file within the directory.

To quickly create an AFTD resource, use the **New Device** wizard accessed from the NetWorker Administration **Devices** window. The wizard asks for the directory and device name and supplies default values for the several attributes in the AFTD resource. By default, the wizard labels and mount a volume in the AFTD device.

Before Performing a Backup - Summary

- Before running a backup, at a minimum, the administrator must:

Task	Completed
Create a client resource for the backup client	Created a client resource for winclient.emc.edu
Create a NetWorker device resource	Created an AFTD, Adv_filedev1
Label and mount a volume for the device	Done automatically for the AFTD by the Create Device wizard

- We are now ready to run our first backup.

EMC²

After creating the required minimum resources (client and backup storage device), we're now ready to run backups.

In the next lesson, we learn how to run backups using the NetWorker Administration user interface and the savegrp command.

Lab Exercise 4-1: Prepare for Backups

In this lab, we will create the resources necessary to perform a first backup of the NetWorker client.

In this lab, you will:

- Create a client resource for the NetWorker client.
- Create a disk storage device (advanced file type device - AFTD).
- Label and mount the device as part of the New Device Wizard

Module 4: Performing Backups

Lesson 1 Summary

During this lesson the following topic was covered:
- Minimum requirements for performing a backup

This lesson covered what must be in place, at a minimum, before performing a backup.

Module 4: Performing Backups
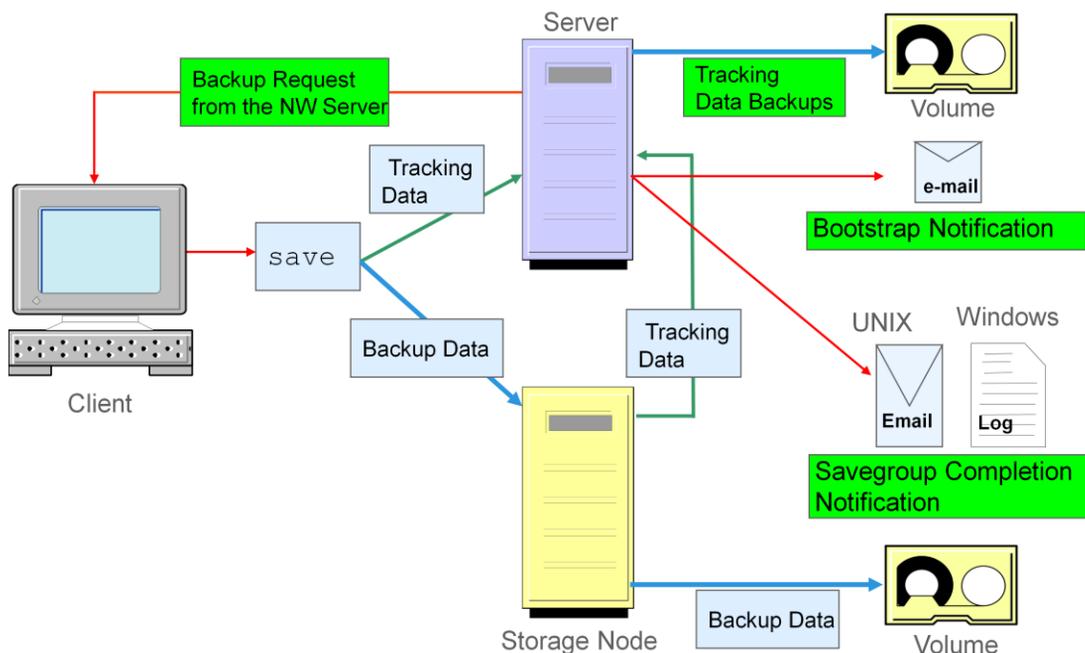
Lesson 2: Performing Server-initiated Backups

During this lesson the following topics are covered:
- Description of a server-initiated backup
- Process and data flow in server-initiated backups
- How to start server-initiated backups
- Tools for monitoring group backups

This lesson covers backups initiated from the NetWorker server, including starting backups both automatically and manually. The lesson also covers various monitoring tools and the process and data flow of server-initiated backups.

Server-initiated Backup Overview

Server

Backup Request from the NW Server

Tracking Data Backups

Volume

Tracking Data

save

Client

Backup Data

Tracking Data

e-mail

Bootstrap Notification

UNIX   Windows

Email   Log

Savegroup Completion Notification

Storage Node

Backup Data

Volume

A **server-initiated backup** is a backup initiated from the NetWorker server using the `savegrp` command. Although `savegrp` can be run manually, it is normally scheduled to start automatically. `savegrp` issues remote execution requests to a configured group of clients, causing the clients to run a backup command (usually `save`) for their configured save sets. The client resource and other configured resources determine what is backed up, when it is backed up, and how it is backed up.

Server-initiated backups also cause the NetWorker server to back up the client file indexes and the bootstrap save set to protect the NetWorker configuration and tracking data. The **bootstrap** is a save set containing information from the media database and the resource directory. The bootstrap save set is required for recovering the server's media database and resource files. The NetWorker server generates and sends a **bootstrap report** as an email to the default email recipient, either the administrator or root. The report contains information about the bootstrap save set, including the name of the volume it was written to and the bootstrap's save set id (ssid). This information is needed to recover the bootstrap save set in the event of a NetWorker server disaster.

The NetWorker server also performs a consistency check on the client file indexes and media database, and performs the aging of save sets.
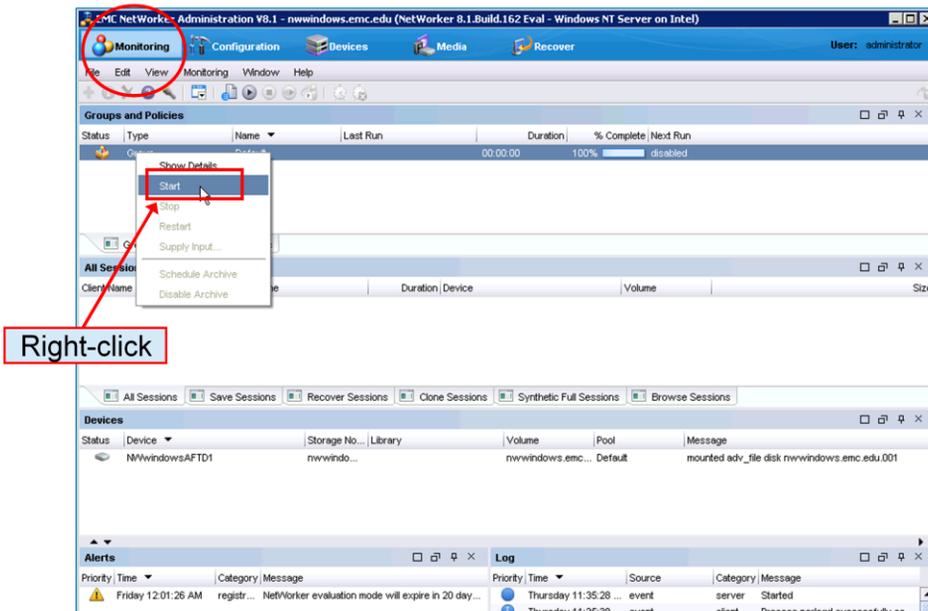
Server-initiated Backup Defaults

NetWorker is installed with a default set of resources. Included in these default resources is a client resource for the NetWorker server that is configured to back up all locally mounted file systems/drives (each file system is a separate save set) on the NetWorker server. By default, full backups are performed on Sundays while incremental backups are performed on all other days of the week. During a *full backup* all specified files and directories, regardless of modification time, are backed up. In an *incremental backup*, only files and directories that have changed since the last backup are saved. Other backup levels available in NetWorker are discussed in later slides.

A group resource with a name of **Default** is also configured and is automatically applied to the NetWorker server client resource.

Assuming that a device resource has been configured and a volume labeled and mounted, the only thing required to enable an automatic daily backup of the NetWorker server is to change the value of the **Default** group's **Autostart** attribute to **Enabled**. After doing this, the server automatically backs up at 3:33 a.m.

Running a Server-initiated Backup: Administration GUI

Copyright © 2013 EMC Corporation. All Rights Reserved.

Module 4: Performing Backups   13

A server-initiated backup can be started manually from either the NetWorker Administration window or the command-line. When using the **_Administration_** window, select the _Groups_ tab in the **_Monitoring_** window. Right-click the group you want to back up and select **Start** from the pop-up context menu.

Starting the group causes the `savegrp` command to be executed on the NetWorker server. `savegrp` asks each NetWorker client in the group, through the server's `nsrjobd`, to execute a `save` process.

Selection	Action
Show Details	Display information concerning the most recent running of the group.
Start	Start the group. This executes the `savegrp` command.
Stop	If the group is currently running, abort it.
Restart	Rerun the group, but back up only those save sets that did not complete successfully during the last running of the group.

Table 4-1: Group Menu Selections

Running a Server-initiated Backup Using `savegrp`

```
savegrp [ -pv ] [ -l level ] [ -c client ] ... [ group ]
```

Option	Description
-p	Preview
-v	Verbose mode
-l level	The level of the save to perform
-c client	Backup only the specified client. Multiple -c options may be used
group	The NetWorker group to start

You can run a server-initiated backup from the NetWorker server's command line using the `savegrp` command. When you execute this command without arguments, it causes any clients belonging to the **Default** group to back up their save sets. Using options, you can customize the backup by specifying such things as which group to back up, which client(s) to back up, and which level of backup to perform.

The syntax of the `savegrp` command is:

```
savegrp [ -option ...] [ -option argument ] [ groupname ]
```

-option represents a single-character option that alters `savegrp` behavior, "*-option argument*" represents an option that requires an argument, and *groupname* represents the NetWorker group.

Example

Back up the client named **winclient.emc.edu** that belongs to the group called **Default** and generate verbose output. No other clients in the group are backed up.

```
savegrp -v -c winclient.emc.edu Default
```

Note: You must log in as root (UNIX) or Administrator (Windows) to run `savegrp`. For more information, see the `savegrp` topic in the *EMC NetWorker Command Reference Guide* or the UNIX/Linux man pages.

Monitoring the Group Backup (1 of 3)
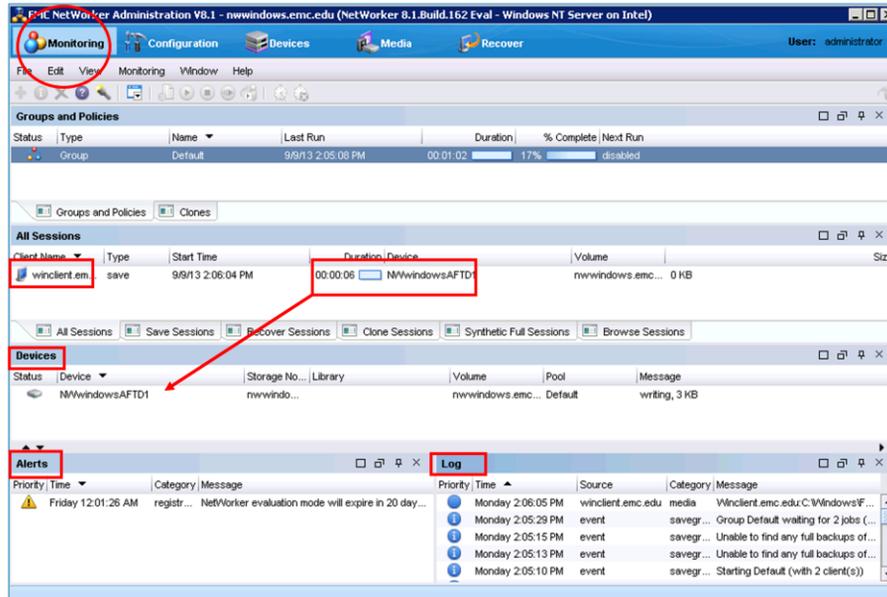
You can monitor a group backup in several ways.

As shown here, from the *Monitoring* window, click the *Save Sessions* tab to view the progress of backups currently running. Additional monitoring information can be seen from this view as well:

- **Groups** – Lists all groups related to the server, the backup status, the time the last backup was run, the duration of the backup, the completion percentage, and the next time the backup will run.

- **All Sessions** – Displays all sessions currently running on the NetWorker server. You can select other sessions tabs to display only certain session types, such as recoveries and clones. You can cancel a session by highlighting the session, right-clicking and selecting **Stop**.

- **Devices** – Contains device performance, amount of data written, and number of sessions currently being written/read.

- **Log** – Contains information about many actions performed by NetWorker during the backup.

- **Alerts** – May contain information such as NetWorker waiting for an appendable tape to be made available.

Monitoring the Group Backup (2 of 3)

Some other ways to monitor backups are shown on the slide:

- When running the **savegrp** command from a command line, use the **-v** (verbose) option to display detail information about the backup. You can use multiple verbose options (up to five) to display additional information.

- For UNIX NetWorker hosts, the **nsrwatch** command-line utility can be used to show information about currently running backups.

Monitoring the Group Backup (3 of 3)

The *Group Details* dialog box contains information about the most recent running of a group. To display group details, double-click the group in the *Groups* tab of the *Monitoring* window. The dialog box can also be displayed by right-clicking the group and selecting **Show Details** from the context menu. Details shown include the start time and duration of the backup, the clients that are waiting to backup, those that are currently running and those that completed successfully or failed. By double-clicking a row containing a message, more information about that particular item is shown.

Savegroup Notifications

Server

Email

Bootstrap Report

```
From root@nwlinux.emc.edu  Wed Sep  5 16:23:01 2012
Date: Wed, 5 Sep 2012 16:23:01 -0400
From: root <root@nwlinux.emc.edu>
To: root@nwlinux.emc.edu
Subject: nwlinux's Bootstrap Information

    date       time        level ssid        file  record   volume
09/05/2012 04:11:14 PM    full 4064784863     0       0    nwlinux.002
09/05/2012 04:19:23 PM    full 3997676488     0       0    nwlinux.002
09/05/2012 04:23:00 PM    full 3913790625     0       0    nwlinux.002
```

UNIX   **Windows**

Email   Log

Savegroup Completion Report

```
Sep 05 08:00:19 nwwindows.emc.edu: Networker savegroup: (notice) Default completed, Total 2 client(s), 2 succeeded.
Sep 05 08:00:19 nwwindows.emc.edu: succeeded: nwwindows.emc.edu, winclient.emc.edu
Sep 05 08:00:19 nwwindows.emc.edu: Start time:   Wed Sep 05 07:59:29 2012
Sep 05 08:00:19 nwwindows.emc.edu: End time:     Wed Sep 05 08:00:19 2012
Sep 05 08:00:19 nwwindows.emc.edu: --- Successful Save Sets ---
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:savefs savefs winclient.emc.edu: succeeded.
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\inf 66135:save: NSR directive file (C:\NETWORKR.CF
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\inf 66135:save: NSR directive file (C:\NETWORKR.CF
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\inf 66135:save: NSR directive file (C:\NETWORKR.CF
Sep 05 08:00:19 nwwindows.emc.edu:   winclient.emc.edu: C:\WUTemp\inf  level=full,    58 MB 00:00:20   1094 files
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\inf completed savetime=1346857177
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\Common Files 66135:save: NSR directive file (C:\NET
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\Common Files 66135:save: NSR directive file (C:\NET
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\Common Files 66135:save: NSR directive file (C:\NET
Sep 05 08:00:19 nwwindows.emc.edu:   winclient.emc.edu: C:\WUTemp\Common Files level=full, 25 MB 00:00:10    225 fi
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\Common Files completed savetime=1346857186
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\Java 66135:save: NSR directive file (C:\NETWORKR.CF
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\Java 66135:save: NSR directive file (C:\NETWORKR.CF
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\Java 66135:save: NSR directive file (C:\NETWORKR.CF
Sep 05 08:00:19 nwwindows.emc.edu:   winclient.emc.edu: C:\WUTemp\Java level=full,   100 MB 00:00:13    651 files
Sep 05 08:00:19 nwwindows.emc.edu: * winclient.emc.edu:C:\WUTemp\Java completed savetime=1346857190
```

EMC²

NetWorker automatically executes two notifications to report information about the completed backup and the location of the bootstrap save set.

Save group completion report: The save group completion report is, by default, either appended to a text file on the NetWorker server (Windows) or emailed to the root user on the NetWorker server (UNIX). This report contains information about the status of each save set backup and provides save set statistics.

Bootstrap report: The bootstrap report is, by default, sent as an email to the default email recipient (administrator or root). The report lists all bootstrap save sets backed up in the last five weeks, with the volume, save set id, and location information necessary to speed the process of control data restoration. The email recipient and the method of sending the report (e.g. send to printer) can be changed by modifying the Action attribute of the Bootstrap notification.

Important: Because accurate and complete control data (resources and databases) is mandatory for the proper functioning of NetWorker, you should make sure you are receiving the bootstrap information. When the report is generated, immediately file the bootstrap report in a safe location for later reference in case a recovery is necessary.

If the bootstrap notification is configured for email and an email recipient is not configured, the bootstrap reports are lost. When a recipient is later configured, the reports are generated during the next savegrp operation. If configured to the printer (not the default configuration) and fails for any reason, the bootstrap information is appended to the savegroup completion report.
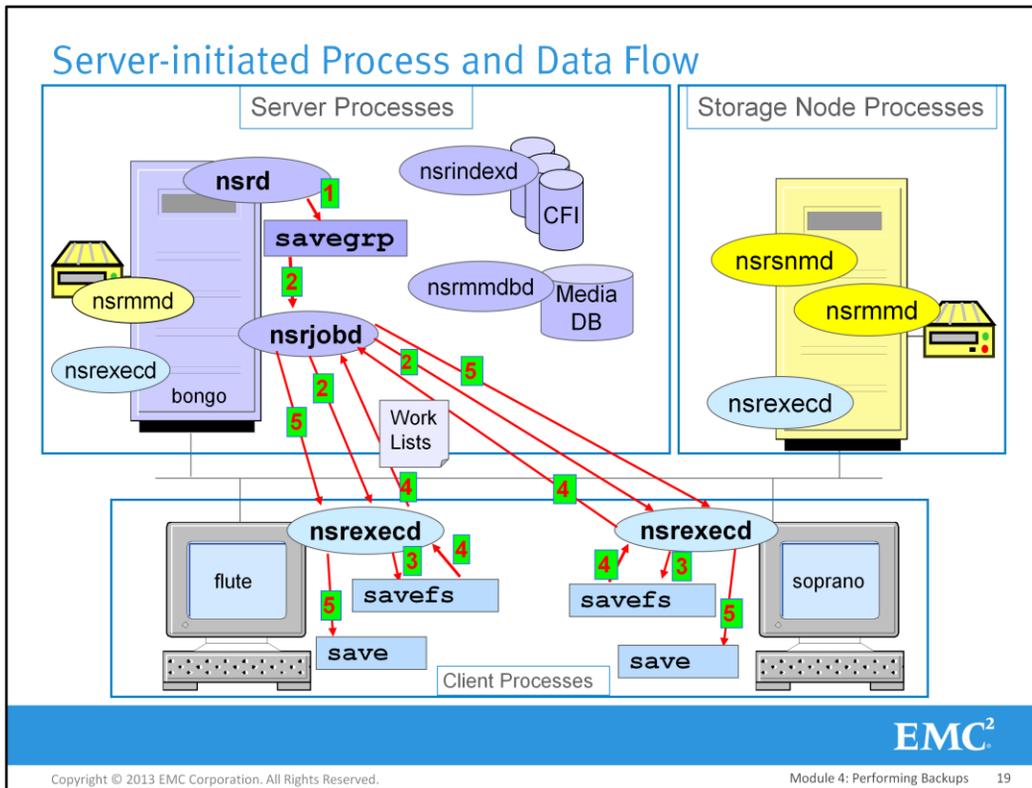
When the server begins a group backup, the `savegrp` command is executed and controls the activity of the backup. The following steps are not necessarily sequential, but represent what occurs when `savegrp` executes. Be aware that the number of processes started depends on the number of save sets specified in the **save set** attribute of the clients in the group.

1. The `savegrp` command starts on the NetWorker server for each autostarted group or as the result of a manual server-initiated backup.

2. The `savegrp` command asks `nsrjobd` to manage remote execution requests by communicating with each client's `nsrexecd`. `nsrexecd` only allows remote execution requests from NetWorker hosts listed in its `/nsr/res/servers` file.

3. `nsrjobd` asks each client to run `savefs` to probe for host and file system information which it returns to `nsrjobd`. If the client's **save set** attribute is **All**, the information returned includes a list of locally mounted file systems/drives.

4. Each client sends a list of save sets to the server as a work list, which the server compiles into a master work list for the group.

5. For each save set on the master work list, `nsrjobd` contacts the client's `nsrexecd` and asks it to back up the save set. `nsrexecd` executes `savefs` again, but with a different set of options which invokes the `save` command to generate the save stream.

CFI and Bootstrap Backup

Server Processes

nsrd

bongo

nsrmmd

savegrp

Media DB

Resource Directory

bootstrap

index:soprano

index:flute

CFI

7

6

Storage Node Processes

nsrsnmd

nsrmmd

Client Processes

flute   nsrexecd

nsrexecd   soprano
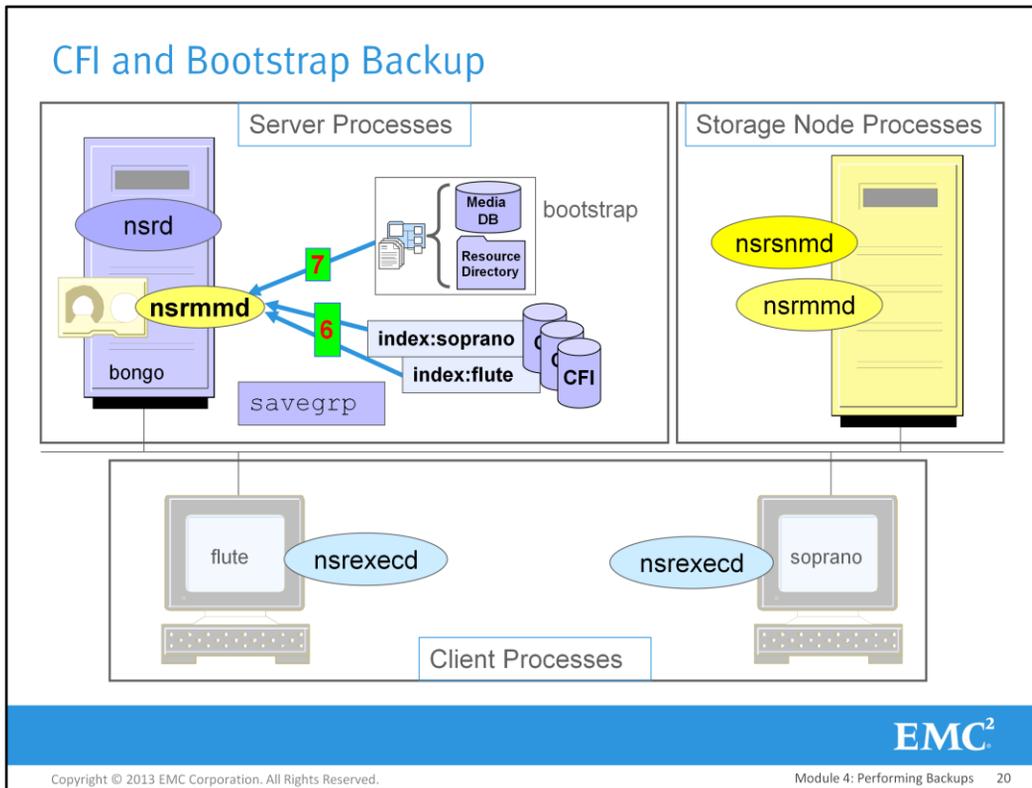
Module 4: Performing Backups   20

EMC²

6. As each client finishes backing up, the NetWorker server uses a special program to back up the client's CFI as a save set named **index:clientname**.

7. The bootstrap save set is backed up at the completion of the group backup for every active (*autostart* = enabled) group containing the NetWorker server.

   If the NetWorker server is not a member of any active group, the bootstrap save set is backed up once every 24 hours. In this case, when any group runs and a bootstrap has not been backed up within 24 hours, the bootstrap is backed up at the completion of the backup.

savegrp Notifications

Server Processes

Storage Node Processes

nsrd

8 Notifications

nsrmmd

bongo

email

Bootstrap Report

nsrindexd

CFI

nsrmmdbd

Media DB

email

UNIX

Log Windows

Savegroup Completion Report

nsrsnmd

nsrmmd

Volume

flute   nsrexecd

nsrexecd   soprano

Client Processes
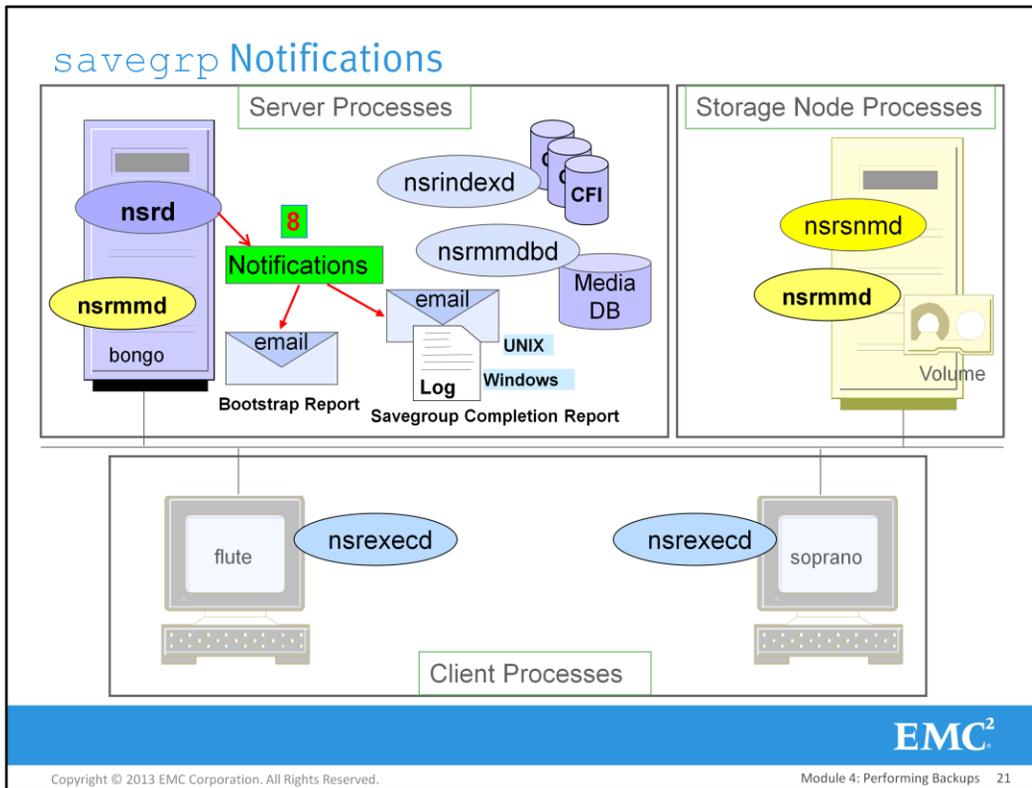
8. After all the client backups are finished and the control data is backed up, the NetWorker server executes configured notifications for these events.

Note: If the NetWorker server performing the backup is not listed in a client's `servers` file, the save group completion report may report an error of *Probe job had unrecoverable failure(s)*.

CFI and Media Database Check

Server Processes

nsrd

nsrmmd

bongo

nsrindexd

CFI

9

nsrmmdbd

Media DB

9

savegrp

✓ **Aging Status**
✓ **Consistency**

Storage Node Processes

nsrsnmd

nsrmmd

Volume

flute

nsrexecd

nsrexecd

soprano

Client Processes
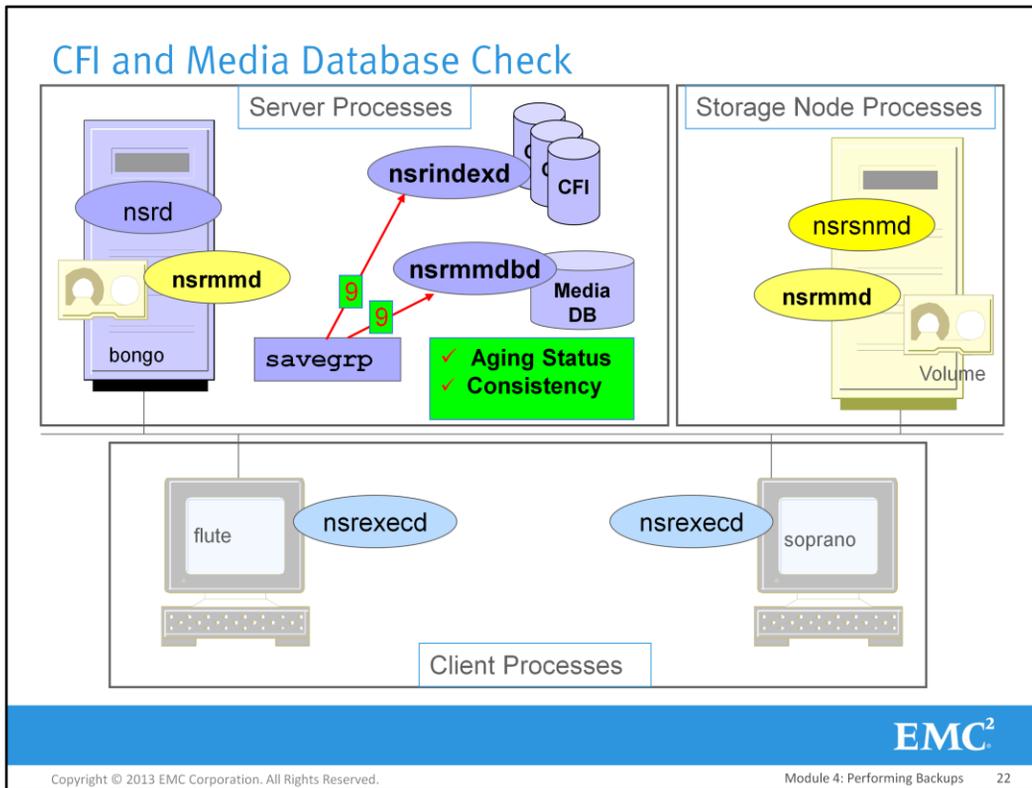
9. At the end of a server-initiated backup, aging and consistency checking is performed. `savegrp` calls `nsrim` to perform the aging of save sets and volumes (based on the browse and retention times of each save set). `nsrim` is run automatically no more than once every 23 hours. The timestamp of `/nsr/mm/nsrim.prv` is used to determine when `nsrim` was last run by `savegrp`.

    After updating the save set browse and retention times in the media database, `nsrim` automatically invokes `nsrck` to remove any entries in the client file indexes that have exceeded the retention policy. If a consistency problem is detected, a more thorough check is automatically performed on the client file index in question.

<u>Note</u>: NetWorker databases and save set life cycles are discussed in more detail later in this course in the module, *NetWorker Database Management*.

Lab Exercises 4-2 and 4-3: Perform Server-initiated Backups

In these labs, you will be running manual backups of the Default group using the NetWorker Administration interface and the savegrp command.

- Lab Exercise 4-2: Perform Backups Using NetWorker Administration
- Lab Exercise 4-3: Perform a Backup Using savegrp

In this lab, you will:

- Perform server-initiated backups using NetWorker Administration.
- Perform a server-initiated backup using `savegrp`.

Module 4: Performing Backups

Lesson 2 Summary

During this lesson the following topics were covered:
- Description of a server-initiated backup
- Process and data flow in server-initiated backups
- How to start server-initiated backups
- Tools for monitoring group backups

EMC²

This lesson covered backups initiated from the NetWorker server, including starting backups both automatically and manually. The lesson also covered various monitoring tools and the process and data flow of server-initiated backups.

Module 4: Performing Backups

Lesson 3: Configuring Resources for Backup

During this lesson the following topics are covered:
- The relationship among the client attributes and resources
- How to configure group, schedule, policy, and client resources
- NetWorker backup levels

EMC²

This lesson covers the resources relevant to configuring a NetWorker client for backups.

Now that you have performed NetWorker backups using default settings, you are ready to customize them by configuring client **resources**, the NetWorker components that determine the specifics of how a client is backed up.

Each resource has attributes that define its relationship to other resources. The slide shows the following relationships among the client attributes and other resources covered in this lesson:

- **Policies** establish aging values for a client's save sets.
- **Groups** define the start time for server-initiated backups (time-based groups) or the time of day when probing occurs (probe-based groups).
- **Schedules** define the backup level used for a client backup on any given day.
- **Directives** provide optional directions for how and what to backup.

Customizing a Backup: Example Scenario

Example Environment

- There are two new NetWorker clients in the Astronomy department.
- Clients need to be backed up at 7:00 PM.
- Full backups must be performed on Monday:
  - ▸ Incremental backups on all other days.
- Astronomy data needs to be recoverable for two months:
  - ▸ Must be able to perform browsable recovery during this time.

The information in the slide describes a fictional environment that NetWorker will be configured to back up. The configuration process is discussed in the upcoming slides.

Configuring a Time-Based Group Resource
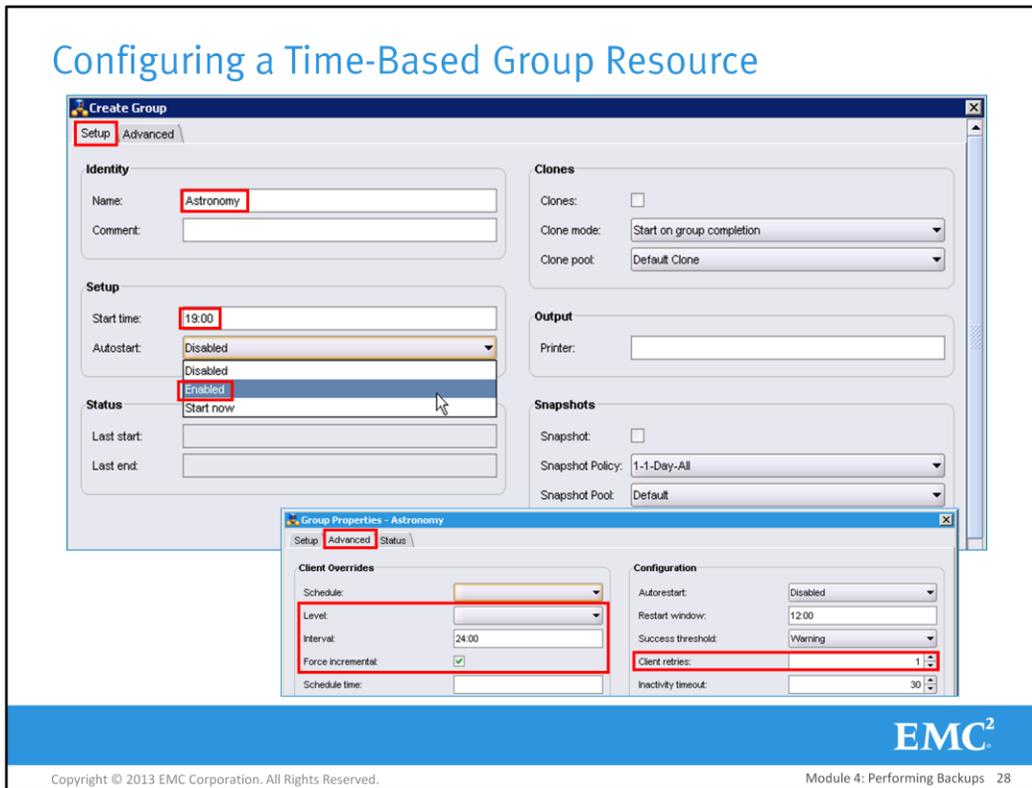
A **time-based group** resource specifies the start time for the backup of a group of clients. Time-based groups can be used to balance the workload throughout the day and to schedule backups for when network traffic is low.

Some common group resource attributes found in the *Setup* tab include:

Name: the name assigned to the group

Start time: the time to start this group if it is autostart enabled. This value is specified using a 24 hour clock. (7 a.m. is 7:00, 7 p.m. is 19:00)

Autostart: determines whether this group will be started automatically (**enabled**) or if it must be started manually (**disabled**).

Some commonly used attributes on the *Advanced* tab for the group resource include:

Level: the backup level used for all client save sets backed up by the group. If this attribute has a value, it overrides the schedule of the clients.

Interval: how often the group runs. For example, a value of 12:00 will cause the group to be backed up every 12 hours. The default value is 24:00, which means once each day.

Force Incremental: when enabled (default), it forces an incremental level of a save group when the interval attribute is less than 24 hours. Uncheck this attribute to allow more than one full backup per day.

Client retries: how many times to retry unresponsive clients. The default is 1.

Note: See the nsr_group topic in the *EMC Command Reference Guide* and the UNIX/Linux man page for more information.

Probe-based Group Backups (1 of 2)

- Event-driven backups.
- Backup of the group is performed based on the result of executing user-created client *probes.*
- A *probe* is a user-created program/script configured to run on clients in a probe-enabled group:
  - The success (exit 0) of the probe triggers the backup.
  - **Probe Interval** determines how often probing occurs (minutes).
  - **Probe Start** and **End** times define the probing window when client probing occurs.
  - **Probe success criteria** determines whether *any* one of the probes must succeed or *all* probes must succeed in order to perform the backup.

Probe-based groups are event-driven backups. The backup of the group is triggered by the successful execution of one or more user-defined client probes (scripts or programs) that are associated with the group.

Unlike a time-based backup group that is run according to the group's **Start time**, when the group **Autostart** attribute is enabled for a probe-based group, probing occurs throughout the probing window, as defined by the **Probe start time** and **Probe end time**. The **Probe interval** determines how often the probes are run during the probing window. If a probe-based group is started manually, probes are executed immediately.

Multiple probes can be run with the group. The execution of the probe(s) determines if the backup of the group is performed. A **Probe success criteria** value of **all** means that all probes associated with the group must succeed for the group backup to be performed; a value of **any** means that the backup is performed if any of the probes associated with the group succeeds. When a probe is executed, success is indicated by a return code of 0.

Probe-based Group Backups (2 of 2)

- The *NSR probe* resource defines the user-defined probe script or program:
  - Name must start with "nsr" or "save"
  - Script/program must reside in the nsr binaries directory on the client
- Probe resources are associated with one or more clients in the probe-based group.
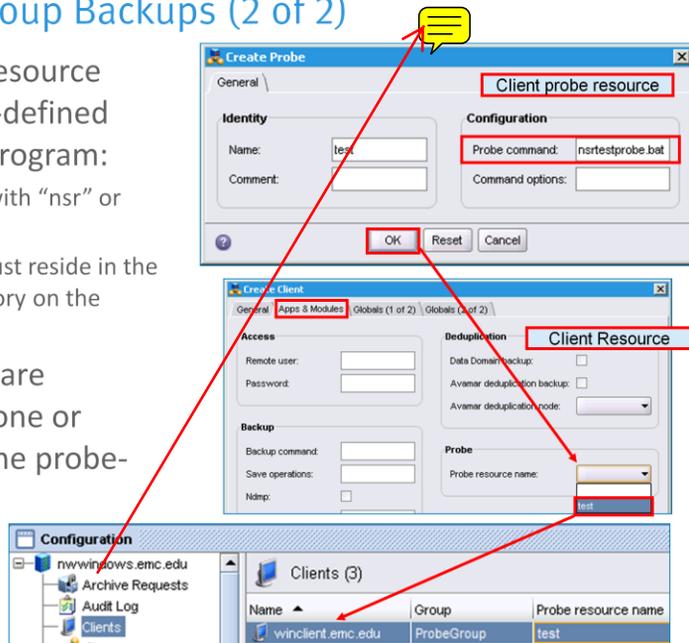
A **probe** is a user-defined script or program that passes a return code. The name of the probe script must begin with `nsr` or `save`.

A NetWorker probe resource is created for each probe script. The probe resource specifies the probe script name and command options, if any.

The probe resource is then associated with one or more client resources. The probe script must reside in the directory that contains the NetWorker client binaries on each client referencing the probe, such as `C:\Program Files\EMC NetWorker\nsr\bin` for Windows clients and `/usr/sbin` on UNIX machines.

Clients are associated with probe-based groups just as they are with time-based groups. Each client in a group can have a probe resource associated with it. In a probe-based group, this determines which probe is run on which client. Clients in a probe-based group are not required to have an associated probe; however, a probe-based group must have at least one probe-enabled client.

Configuring a NetWorker Schedule Resource

Module 4: Performing Backups   31

The **schedule** resource defines the level of backup performed by a client or group of clients in a server-initiated backup.

When a new schedule resource is created, it is initially configured so that a full backup is performed on Sundays and incremental backups are performed on all other days of the week. The slide depicts creating a new schedule to perform full backups each Monday and incremental backups on all other days of the week.

To change the schedule so that an incremental backup is performed on Sundays, right-click on any Sunday on the calendar, select **Set Level**, and choose **Incremental** from the menu.

To configure a full backup on Mondays, right-click on any Monday, select **Set Level**, and choose **Full** from the menu.

You can specify a NetWorker schedule in both a client resource and a group resource. If the schedule is applied to a group, it overrides the schedules applied to clients within that group.

NetWorker Backup Levels

Level	Description
Full	Back up all files regardless of whether they have changed. This is equivalent to a UNIX level 0 backup.
Level 1–9 (Differential)	Back up files that have changed since the most recent lower level backup.
Incremental	Back up files that have changed since the last backup, regardless of what level it was.
Skip	Do not perform a backup today.
Synthetic Full	Combines a full backup and subsequent incremental backups to make a new full backup
Incremental Synthetic Full	Performs an incremental backup, adds that to the full and incrementals, and then performs a synthetic full

NetWorker supports full level backups that back up all data in a save set, or one of several levels that back up only data that has changed since a previous backup. The levels used are similar to the UNIX `ufsdump` or `dump` command.

The supported backup levels are listed in the slide and are explained over the next several pages.

A **full backup** backs up all files and directories in a save set and is the lowest backup level, being equivalent to a UNIX level 0 backup. A full backup requires the most storage space and takes the longest time to perform.

Incremental and Level Backups
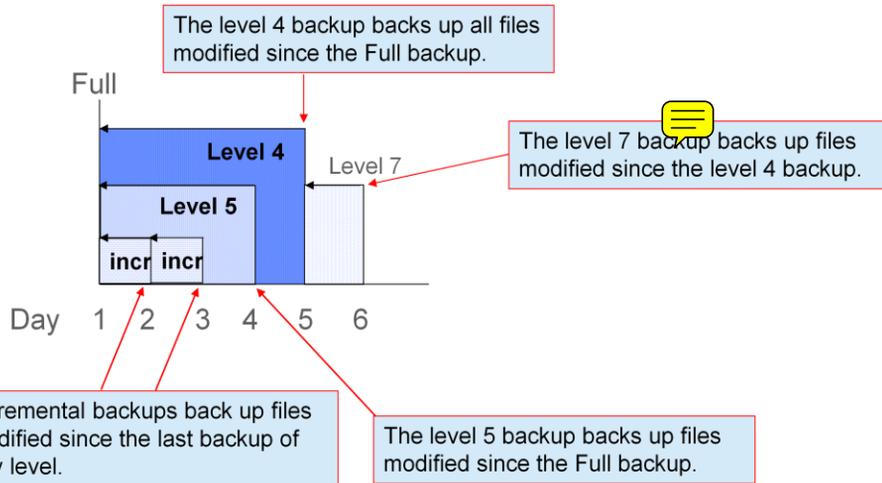
The level 4 backup backs up all files modified since the Full backup.

The level 7 backup backs up files modified since the level 4 backup.

Incremental backups back up files modified since the last backup of any level.

The level 5 backup backs up files modified since the Full backup.

Levels 1-9

A **level backup** backs up only files and directories that have changed since the most recent lower level backup. For example, a level 5 backup backs up all files and directories that have changed since the most recent level 4, 3, 2, 1, or full backup. A full backup is treated as a level 0. Valid levels include level 1 through level 9.

Incremental

An **incremental backup** backs up files and directories that have changed since the last backup, regardless of the level of the previous backup.

Using level [1–9] and incremental backups takes less time than full backups and uses less volume space. However, using them can also slow file recovery if multiple save sets are required to recover to a particular point in time.

Synthetic Full Backups

- Creates a new full backup using a previously created full backup and subsequent incremental backups.
  - Takes into account deleted files.
- Results:
  - New save set that is independent of previous backups.
  - Movement of backup data is transferred to storage node.
  - Recovery time and steps are reduced.

Monday	Tuesday	Wednesday	Wednesday	Thursday
Full Backup	Incremental Backup	Incremental Backup	Synthetic Full Backup	Incremental Backup

A **synthetic full** backup is formed by combining a full backup and subsequent incremental backups. The resulting backup is a full backup equivalent to a traditional full backup as of the time of the last incremental backup used in the creation of the synthetic full backup. The synthetic full backup is not just the sum of the incrementals , but takes into account deleted files as well.

Only the NetWorker server and storage nodes are involved in synthetic full backup processing. By lessening the number of traditional full backups, the backup workload of backup clients is reduced, as well as the network overhead involved in transferring the backup data from the clients to the storage node. Synthetic backups also reduce recovery time and steps as data can be restored from the synthetic full backup instead of a traditional full backup and all its dependent incrementals.

In the example shown on the slide, the synthetic full backup taken on Wednesday combines the full backup run on Monday with the incremental backups run on Tuesday and Wednesday. The resulting synthetic full backup is equivalent to a traditional full backup run at the same time as the Wednesday incremental backup and reflects the state of the data as of Wednesday's incremental. The incremental backup run on Thursday will include all changes since the incremental on Wednesday. The next synthetic full backup (not shown on the slide) will combine the previous synthetic full backup and subsequent incremental backups.

Virtual Synthetic Full Backups (Data Domain Only)
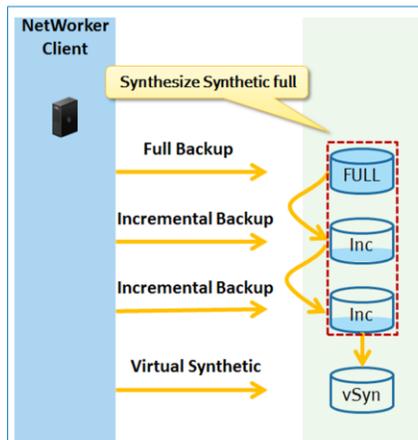
- When full and incremental backups are stored on Data Domain, NetWorker can synthesize the metadata to create a virtual synthetic full
  - Takes into account deleted files.
- Results:
  - Reduces processing overhead
  - No data movement required
  - Provides near instantaneous incremental backups
  - Up to 90% reduction in number of full backups run annually

NetWorker 8.1 introduces support for the ability to perform Virtual Synthetic Full backups with Data Domain. The processes of creating a Virtual Synthetic full is a much more efficient way to create a Synthetic full backup and it is now the default method in NetWorker 8.1.

In typical backup cycle the administrator will schedule a full backup followed by several incremental backups. To create a Virtual Full backup, NetWorker sends commands to the Data Domain System that consist of the regions that are required to create a full backup. During the transfer no data is transferred over the network. Instead, the regions of the full backup are synthesized from the previous full and incremental backups that are on the system using pointers. This process eliminates the data that needs to be gathered from the file server, reducing system overhead, time to complete the process, and network bandwidth requirements. NetWorker uses the DDBoost API to create the Virtual Synthetic full backups.

Virtual Synthetic full backups are an out-of-the-box integration with NetWorker, making it 'self-aware.' Therefore, if you are using a Data Domain System as you backup target, NetWorker will use Virtual Synthetic full backups as the backup workflow by default when a Synthetic full backup is scheduled, thus optimizing incremental backups for file systems. Virtual Synthetic Fulls reduce the processing overhead associated with traditional Synthetic full backups by using metadata on the Data Domain system to synthesize a full backup without moving data across the network. A traditional full backup is recommended only after every 8-10 Virtual Fulls have been completed. Therefore, the use of Virtual Synthetic Full backups also reduces the number of traditional full backups from 52 to 6 per year – a 90% reduction. If a Virtual Synthetic full operation fails, NetWorker will default to creating a Synthetic full.

Virtual Synthetic Full Backups: Requirements

- Save sets
  - Save sets must not be Avamar deduplication save sets.
  - Directives cannot be specified with virtual synthetic fulls.
- Data Domain
  - DDOS must be 5.3 or greater.
  - Virtual synthetic full feature must be enabled.
  - Target volumes used to create virtual synthetic full must belong to the same Data Domain device.
- NetWorker
  - NetWorker 8.1 is required on the NetWorker server and SN.

This slide lists a few requirements to create virtual synthetic fulls. It is important to note virtual synthetic fulls cannot use directives such as encryption or compression like synthetic fulls can because data is not read from the Data Domain device during the creation of the virtual synthetic full. Additionally, Virtual Sythentic Full backups were not introduced until NetWorker 8.1, so this version of the NetWorker software is required on both the server and the storage node.

The Backup Period

The **Period** attribute of the schedule resource determines how a change to the level on one day of the schedule is propagated to other days on the schedule.

Week – A change to a day of the week is applied to that day of the week, for all weeks. For example, changing any Sunday to **skip** results in all Sundays being set to **skip**.

Month – A change to a day of the month is applied to that day of the month, for all months. For example, changing April 15 to **full** results in March 15, April 15, May 15, etc. being set to **full**.

Setting a Level Override

- Changes level for a specific date.
- Useful for scheduling a single backup that departs from the normal schedule.

Right-click date on which to set override

An '*' signifies an override
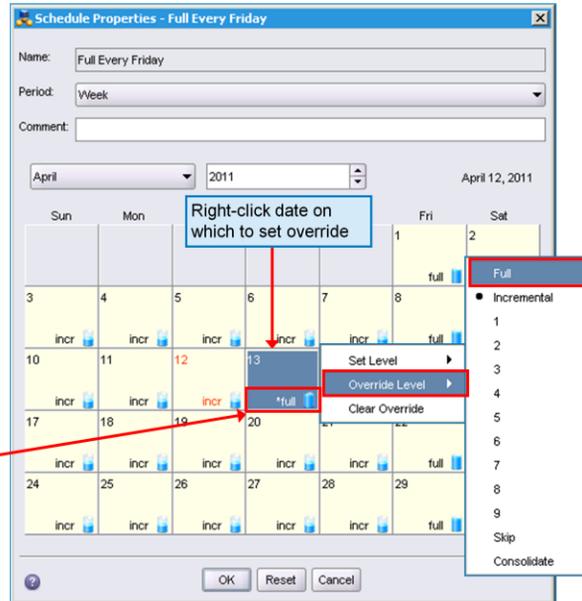
The **Level Override** feature allows you to schedule a level of backup to be performed on a single, specific date. Setting a level override does not affect any day other than the date being configured.

As an example, maybe you did an operating system upgrade on Tuesday, the 10th of September, and you want to perform a full backup on Wednesday the 11th. With a period of **Week**, changing the level for the 11th results in all Wednesdays getting set to a full backup. With a period of **Month**, changing the 11th results in February 11, March 11, etc. being changed to full. Neither of these situations is desirable. However, you can accomplish the goal by setting an override level of full for February 11th, as shown in the slide.

Schedule properties are displayed by default as a calendar (**Configuration > Show Schedule as Calendar**). By disabling this setting, you can specify overrides using a list of dates and actions. For example, to perform a full backup the second Sunday of every month, with the period set to month, you enter "full second Sunday every month".

NetWorker Time Policies

A NetWorker Time Policy:

- Defines a length of time.
- Determines how long data is tracked in the media database or client file indexes.
- Is used for either of two client attributes:
  - *Browse policy* – determines the minimum length of time that entries remains in the client's CFI.
  - *Retention policy* – determines the minimum length of time that a client's save sets are protected from automatic recycling and therefore remain recoverable.

A NetWorker **policy** is simply a length of time, for example two months or three years. A policy is applied to a client resource through two different client resource attributes: **Browse policy** and **Retention policy**.

A NetWorker client's **browse policy** determines how long entries are maintained in the client's CFI. CFI entries older than the browse policy are automatically removed during a server-initiated backup. When the CFI entries for a save set are removed that save set can no longer be recovered by browsing the CFI and selecting files to recover. However, files can still be recovered using a save set recovery. This is discussed later in this course in the module, *Performing NetWorker Recoveries*.

A NetWorker client's **retention policy** determines the minimum length of time that a save set backed up by the client is guaranteed to not be automatically overwritten through the recycling (relabeling) of the volume on which it resides When a volume is recycled, which can also be done manually, its tracking information is removed from the media database and the save sets on the volume can no longer be recovered.

Because several other factors determine when a volume is automatically recycled, a client's save sets may be recoverable well beyond the length of time specified by the client's retention policy. The module, *NetWorker Database Management,* discusses these factors in detail.

Configuring a NetWorker Time Policy Resource
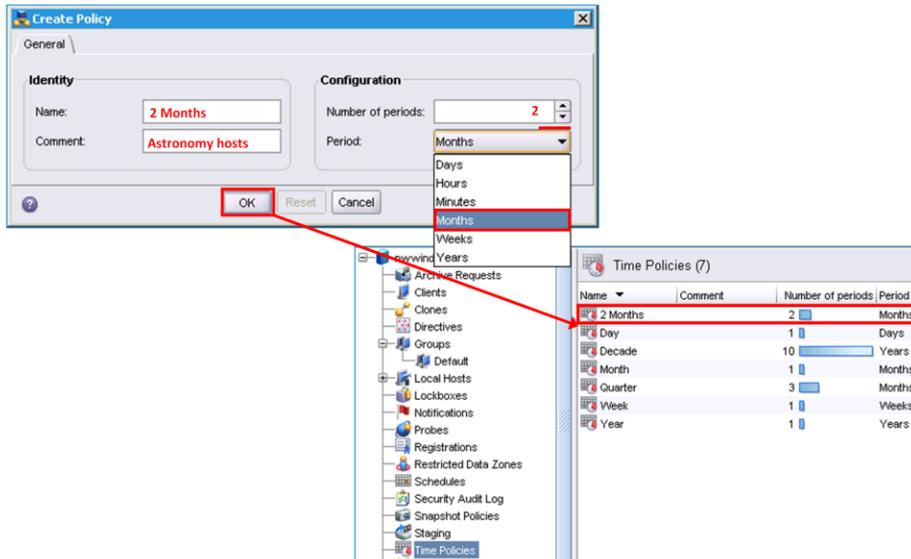
The NetWorker Time Policy resource defines a length of time. It is used within the client resource to define the browse policy, retention policy, or both. A policy can be used for one client's browse policy and another client's retention policy.

Refer to the `nsr_policy` topic in the *EMC NetWorker Command Reference Guide* or the UNIX/Linux man pages for more information.

Attribute	Function
Name	Displays the name assigned to this policy.
Period	Shows the basic length of time used by the resource. Valid values are **Minutes**, **Hours**, **Days**, **Weeks**, **Months**, and **Years**. This length of time is multiplied by the **Number of periods** to determine the actual length of time of the policy.
Number of periods	Lists the number of periods this policy represents.

Table 4-2: `nsr_policy` Resource Attributes

Configuring a NetWorker Client Resource

After customized group, schedule, and policy resources are created, they can be applied to a client resource.

In the slide, a new client resource is being created for one of the astronomy hosts mentioned at the beginning of the lesson. The newly created **2 months** policy is used as the client's retention policy, the client is added to the new **astronomy** group, and the client is configured to use the new **Full on Monday** schedule.

Note that a client can belong to multiple groups. However, for this example, **winclient.emc.edu** is only backed up once a day. Thus, it is important to remember to remove the client from the **Default** group, which is automatically selected when creating a new client resource.

The Save Set Attribute - the All Save Set

Operating System	Components of All
Windows XP Professional, Windows Server 2003 with VSS disabled	All local, physical drives SYSTEM STATE SYSTEM DB SYSTEM FILES SHAREPOINT
Windows Server 2003 with VSS enabled Windows Vista Windows Server 2008	VSS SYSTEM BOOT VSS SYSTEM FILESET VSS SYSTEM SERVICES VSS USER DATA (Windows Server 2003 only) VSS OTHER (Windows Server 2003 only) All local physical drives
Windows 2008 Windows 2008 R2 Windows 7	DISASTER_RECOVERY:\ (included in full backup only, not supported for EFI systems) VSS SYSTEM BOOT VSS SYSTEM FILESET VSS SYSTEM SERVICES All local physical drives
UNIX	All currently mounted file systems listed in /etc/vfstab (Solaris), /etc/fstab (HP-UX, Linux), /etc/filesystems (AIX)
MAC OS X	All local and mounted volumes

Use an "All" keyword to define file systems to include in the backup

Module 4: Performing Backups

The **Save set** attribute defines which files on the client are backed up. The default value for the **Save set** attribute is **All**, which causes all local file systems/drives to be backed up. For Microsoft Windows clients, a number of operating system files are also backed up. The version of Microsoft Windows determines precisely which save sets are backed up.

When processing the **All** save set, the `savefs` command probes the client's file systems and creates a separate save set for each local file system/drive and SYSTEM save set.

Notes:

For more information about VSS SYSTEM save sets and backing up with VSS, please see the module, *VSS Backups and Cluster Environments,* later in this course.

Use special keywords with All to define the file systems to include in a client backup. Examples of keywords include:

- all-*filesystem*: only back up locally mounted file systems of a particular type; examples include all-zfs, all-ntfs, all-ext3.
- all-mounts: back up all of the currently mounted file systems (UNIX); for Windows, equivalent to the save set All.
- all-local: backs up all file systems in the sparse or whole root zone on the physical host for global zone clients; for sparse or whole root zone clients, equivalent to the save set All.
- all-global: backs up all file systems in the global zone for global zone clients, all sparse and whole root zone file systems on the physcial host are skipped; for sparse or whole root zone clients, equivalent to the save set All.
- Beginning with NetWorker v8 clients, ASR backup is no longer supported. However, ASR recovery using pre-NetWorker v8 client ASR save sets is still supported.

The Save Set Attribute - Selecting Files to Backup

You can also use the **Save set** attribute to manually define the files, directories, and file systems the client is asked to back up as save sets. Each entry must be a single file, a single directory, or a single file system. There is no limit to the number of save sets you can specify.

On the client resource **Properties** window, you can enter the information in the **Save set** box or click **Browse** (magnifying glass) to browse the client's file systems and select the items to back up from the browse list.

Important: If **save set** is set to anything other than **All** and you want any of the SYSTEM save sets to be backed up, you must explicitly specify them in the save set list.

Checkpoint Restart Backups
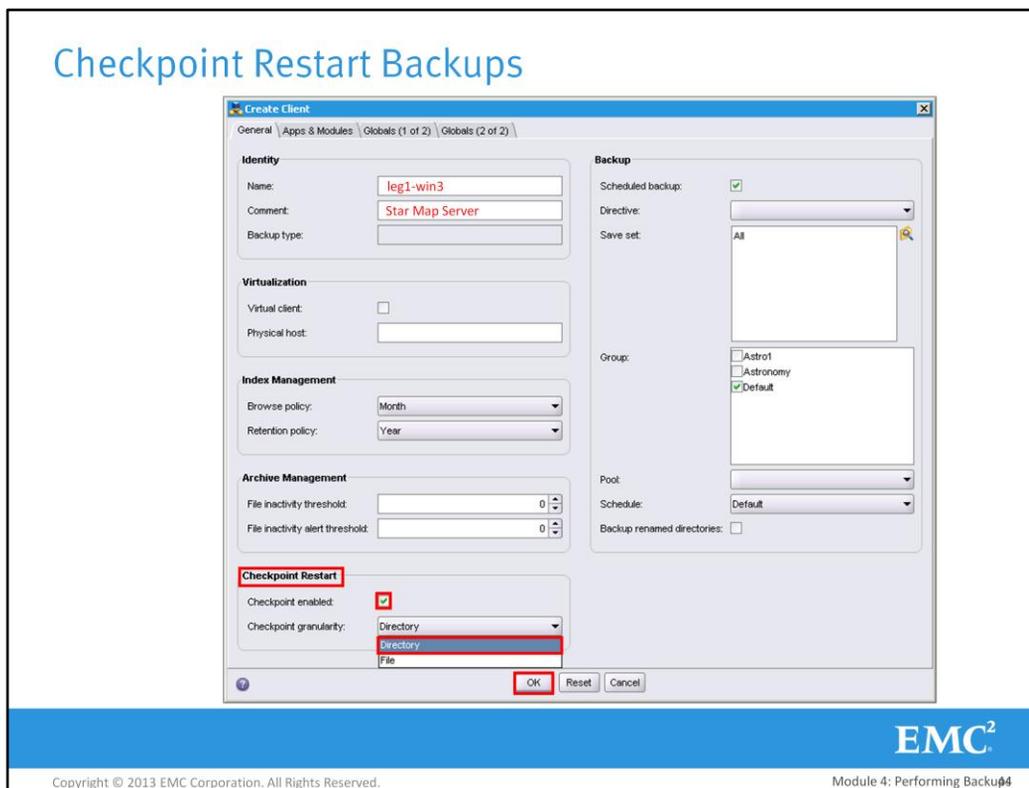
When **Checkpoint Restart** is enabled, failed backup operations can be restarted at a known good point prior to the point-of-failure during the backup. A known good point is defined as a point in the backup data stream where the data was successfully written to the backup media and that data can be located and accessed by subsequent recovery operations. Client backups can be restarted should they fail while running, and files and directories that have already been backed up are not backed up again.

Checkpoint Restart is configured on a per client basis.

The checkpoint restart feature is not enabled by default. To enable the feature, check the **Checkpoint enabled** box on the client resource general tab. Checkpoint granularity is the level at which the backup can be restarted, either at the directory or file level. When restart by directory is selected, after each directory is saved, the data is committed to the index and media database. If restart by file is selected, every file is committed to the index and media database. This is time consuming and has the potential to degrade performance during a backup containing many small files. Because of this, restarting by file is recommended only for save sets with few, large files.

Note: The NMC database cannot be backed up as part of a Checkpoint Restart backup.

Configuring Synthetic Full Backups (1 of 2)

- Create a client resource for each participating storage node
- For each participating backup client resource:
  - Use a group resource created for synthetic group backups.
  - Assign a pre-configured Incr+Synthetic Full schedule or a new schedule that includes the synthetic full level.
  - Enable the **Backup renamed directories** attribute.
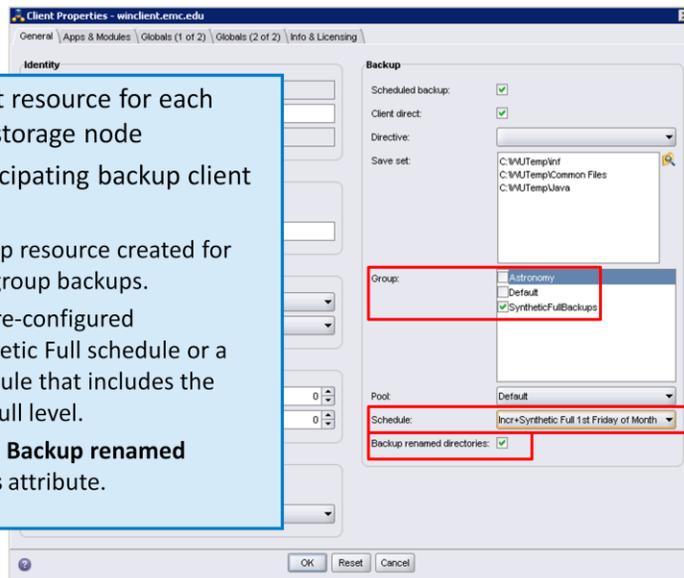
Tasks for configuring a scheduled synthetic full backup include:

- Create a client resource for the participating storage node.
- Create group resource(s) especially for scheduled synthetic full backups. Configure Windows clients within a dedicated group. Do not mix with UNIX clients.
- Review the pre-configured schedules for synthetic full backups. Plan to use one of the pre-configured schedules or create a new schedule for specific site requirements.
- Create a client resource for each storage node that will be performing scheduled synthetic full backups.
- Create a client resource for each backup client that participates in the synthetic full.
  - Ensure that the save sets meet synthetic full requirements.
  - Assign group and schedule.
  - Enable the **Backup renamed directories** attribute. This attribute is enabled by default with client resources created with NetWorker 8.0 or later. It must be specifically enabled for other participating clients.

.

Configuring Synthetic Full Backups (2 of 2)

Group resource attributes for synthetic full backups:
- Verify synthetic full - verifies the integrity of the new index entries
- Revert to full when synthetic full fails

There are two attributes on the group resource that can be enabled to check the integrity of synthetic backups:

- **Verify synthetic full:** Verifies the integrity of the new index entries that are created for the synthetic full backup in the client file index.

- **Revert to full when synthetic full fails:** Forces the group to perform a full backup on the save set if the synthetic full backup fails.

.

Requirements for running synthetic full backups include:

- A full backup or a synthetic full backup, created with NetWorker 8.0 or later, must exist.
- All incremental backups participating in the synthetic full backup are included in the media database.
- All save sets participating in the synthetic full must:
  - Have the same client and save set names.
  - Be browsable, that is entries for the save set must be in the client file index.
- Limit the number of concurrent synthetic full operations to 20.
- Participating storage nodes must have attached devices for read and write.
  - Devices supporting concurrent read/write access can be used if pool restrictions and session limits allow.
  - Consider using AFTD or Data Domain devices to store all backups on a single device.

.

Synthetic Full Considerations and Recommendations

- Supported for file system backups only
- Schedule to run outside of normal backup windows
  - ▶ Create separate groups to run synthetic full backups
  - ▶ Run synthetic full backups in place of full backups
  - ▶ Schedule regular full backups once a month
- Can be directed to any device that can be used in a traditional full backup
  - ▶ Recommendation to use devices supporting concurrent read/write operations, such as AFTD and Data Domain, to avoid volume contention issues and for better performance.

Only file system type backups are supported. Application modules or NDMP backups are not supported.

Using synthetic full backups can reduce the number of full backups that need to be run but does not eliminate the requirement to run full backups. Scheduling recommendations include:

- Use a separate group for running synthetic full backups.
- Run synthetic backups as a replacement for full backups, not in addition to.
- Perform full backups on a regular basis, typically once a month.
- Schedule outside of backup windows. Because synthetic full processing is resource intensive on the storage node, run synthetic full backups at times other than when backups are running so as not to impact regular backup processing.

Synthetic backup processing involves concurrent recover and save operations. Because of this, it is recommended to use backups devices that support concurrent operations, such as advanced file type and Data Domain devices. This allows NetWorker to automatically manage volume contention.

These slides present an overview of the synthetic backup feature. For more information, please refer to the *EMC NetWorker Administration Guide*.
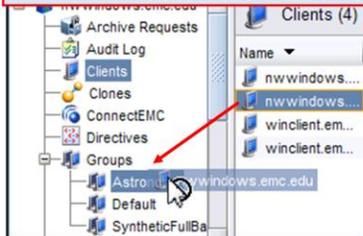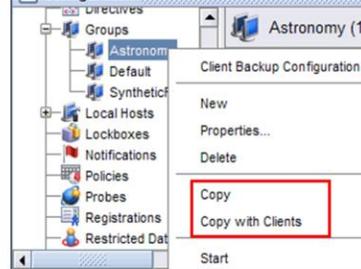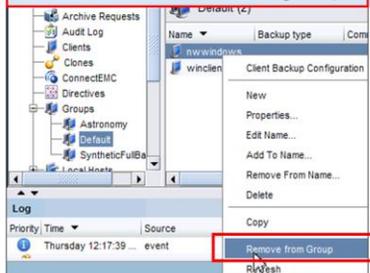
Several ease of use capabilities, including drag and drop, are available in the Administration interface for managing group resources. These capabilities include the ability to:

- Add a client or assign a schedule to a group by dragging a resource from the right pane onto a group name in the left pane.

- Remove a client from a group by selecting the group in the left pane, selecting the client in the right pane and selecting **Remove from Group** from the drop-down menu.

- Copy a group resource with or without adding all clients belonging to the original group to the new group. Note also that a group can be started from the **Configuration** window.

- Start multiple groups simultaneously.

Client Backup Configuration Wizard

- Configure new or existing client resources

File System Wizard Windows

- Specify the Client Name and Type
- Specify the Backup Configuration Type
- Specify the Client Backup Options
- Select Files to Backup
- Select the Client Properties
- Choose the Backup Group
- Specify the Storage Node Options
- Backup Configuration Summary
- Check Results

1st Wizard Window

Traditional Client

Copyright © 2013 EMC Corporation. All Rights Reserved.   Module 4: Performing Backups   50

Up until now, we have been using menu commands to manually configure our backup client. NetWorker provides the *Client Backup Configuration* wizard to walk users through the steps to quickly create a client. Currently support is provided for configuring new and existing file system clients, virtual clients and VMware proxy hosts.

The *Client Backup Configuration* wizard is accessed through the *Configuration* view, by right-clicking **Clients** in the left pane, and for an existing client, by right-clicking the client row in the right pane of the *Configuration* window.

With the *Client Backup Configuration* wizard, you can:

- Create a client resource.
- Modify an existing client resource.
- Create a backup group.
- Add a client resource to a backup group.

Note that the Console server, NetWorker client and NetWorker server must all be using NetWorker 7.5 or above in order to use the wizard. Multiple wizard hosts cannot access the same client machine at the same time.

Lab Exercise 4-4: Create New Resources

In this lab, you are presented with a scenario for a data zone configuration. You will configure all the resources needed to perform the backups requested in the scenario.

In this lab, you are presented with a scenario for a data zone configuration. You configure the NetWorker resources needed to attain that configuration, creating group, schedule, policy and client resources. Then, you perform a backup of all the groups and verify they complete without error.

Module 4: Performing Backups

Lesson 3 Summary

During this lesson the following topics were covered:
- The relationship among the client attributes and resources
- How to configure group, schedule, policy, and client resources
- NetWorker backup levels

EMC²

This lesson covered the resources relevant to configuring a NetWorker client for backups.

Module 4: Performing Backups

Lesson 4: Client Backup Command Attribute

During this lesson the following topics are covered:
- Backup command attribute on the client resource
- Using the savepnpc command

This lesson covers the **Backup command** attribute on the client resource.

Client Backup Command Attribute
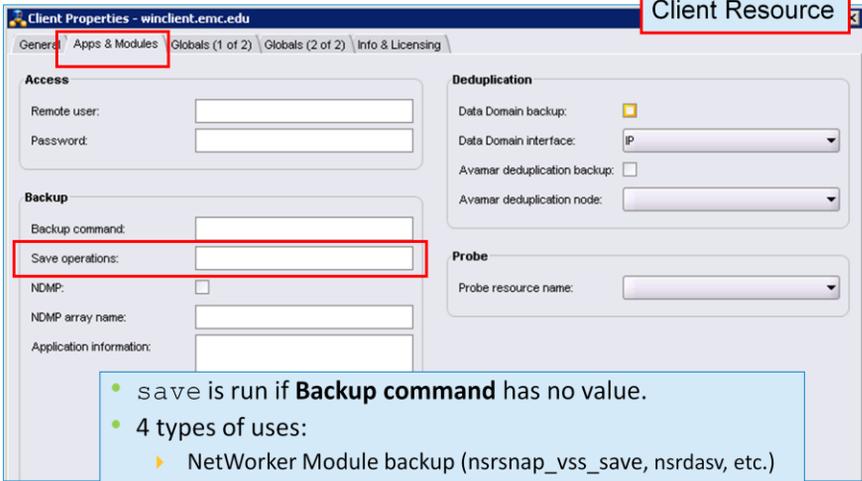
Client Resource

Client Properties - winclient.emc.edu

General | Apps & Modules | Globals (1 of 2) | Globals (2 of 2) | Info & Licensing

Access

Remote user:

Password:

Backup

Backup command:

Save operations:

NDMP: ☐

NDMP array name:

Application information:

Deduplication

Data Domain backup: ☐

Data Domain interface: IP

Avamar deduplication backup: ☐

Avamar deduplication node:

Probe

Probe resource name:

- `save` is run if **Backup command** has no value.
- 4 types of uses:
  - ‣ NetWorker Module backup (nsrsnap_vss_save, nsrdasv, etc.)
  - ‣ Custom script that you write
  - ‣ `savepnpc`
  - ‣ `save` with non-default options

By default, a NetWorker client's **Backup command** attribute is blank, causing **`save`** to be executed for each save set listed in its **save set** attribute. By modifying the **Backup command** attribute, you can change the command used to perform the backup.

The **Backup command** attribute is often used when implementing one of NetWorker's add-on modules, such as NMM and NMDA, to do application-specific backups.

You can create a custom script to perform tasks before, after, or instead of the `save` process. These tasks might include moving, deleting or renaming files, stopping and starting processes, or generating logging information. When writing a custom script, you must include the `save` command if you want a save stream to be generated. The `save` command should have an argument of `$*` to retain all of the arguments sent by the NetWorker server.

The custom script must have a name that begins with `nsr` or `save` (for example, `nsr_my_custom_command` or `save_my_custom_command`). The custom script file must also reside in the same directory as the NetWorker `save` command. On Windows hosts, the default location of `save` is `C:\Program Files\EMC NetWorker\nsr\bin`; on UNIX hosts, execute `which save` to determine the location.

You can also specify the `savepnpc` command in the client's **Backup command** attribute. Use `savepnpc` if you want to run either pre-processing commands before any client save sets are backed up and/or post-processing commands after all save sets have been backed up.

Backup Command: Blank, NW Module, or Custom Script
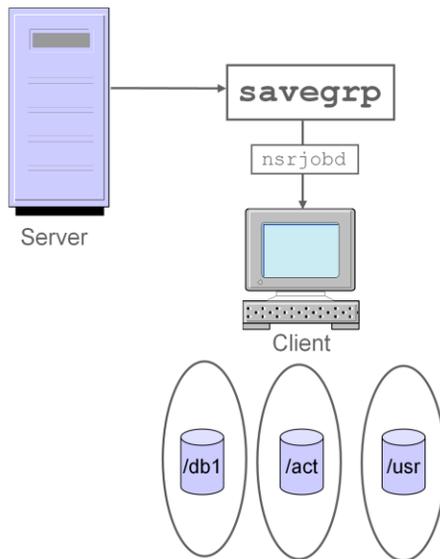
`savegrp`

`nsrjobd`

Server

Client

/db1   /act   /usr

1. The server starts a server-initiated backup.

2. The **savegrp** command initiates a client backup request.

3. The client runs the command specified in the client's **Backup Command** attribute once for each save set.

Module 4: Performing Backups   55

As shown in the slide, when a client's **Backup command** attribute contains anything other than `savepnpc`, the specified command (or `save` if the attribute is blank) is executed once for each save set. Thus, if a client has three save sets, the backup command is executed three times.

Using the `savepnpc` Command
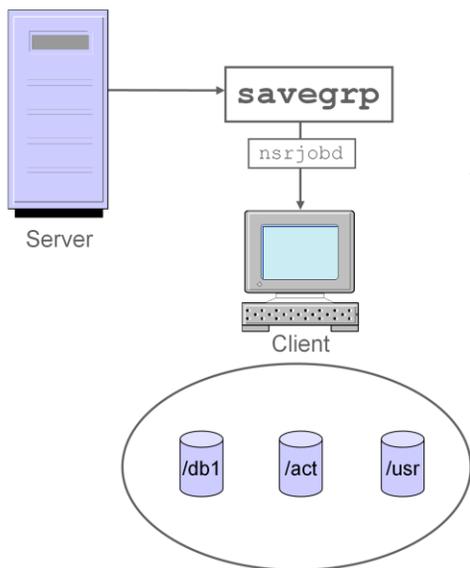
Server

savegrp

nsrjobd

Client

/db1 /act /usr

1. The server starts a server-initiated backup.

2. The **savegrp** command initiates a client backup request.

3. The client runs the **savepnpc** command.

   a) The **savepnpc** command performs optional pre-processing before *any* save sets are backed up, regardless of the number of save sets.

   b) The **savepnpc** command runs the NetWorker **save** command for each save set.

   c) The **savepnpc** command performs optional post-processing after *all* save sets are backed up.

Unlike other backup commands which execute once for each save set, `savepnpc` runs only once, regardless of the number of save sets specified in the client resource. This behavior is useful if the client is running an application that you need to shut down before backing up the client, `savepnpc` can then restart the application when the backup is complete.

<u>Note</u>: For more information on `savepnpc`, see the `savepnpc`, `preclntsave`, and `pstclntsave` topics in the *EMC NetWorker Command Reference Guide* or the UNIX/Linux man pages.

savepnpc and the /nsr/res/groupname.res File

```
type: savepnpc;
precmd: "echo hello";
pstcmd: "echo bye, /bin/sleep 5";
timeout: 12:00:00;
abort precmd with group: No;
```

Default *groupname*.res file (UNIX)

Attribute	Syntax	Description
type	savepnpc;	Resource type is savepnpc
precmd	cmd1, "cmd2 arg", …;	Command(s) to run before any saves
pstcmd	cmdX, "cmdY arg", …;	Command(s) to run after all saves are finished
timeout	time	Run the **pstcmd** at *time*, whether or not the backup is finished. *time* may be specified in any nsr_getdate format.
abort precmd with group	Yes | No	Specifies whether or not to kill **precmd** if the group is aborted

To configure a client to run savepnpc, enter savepnpc in the client's **Backup command** attribute.

When savepnpc runs on a client, it looks in the client's /nsr/res directory for a file named *groupname*.res. The *groupname*.res file contains NetWorker resource file format syntax that defines the savepnpc resource and its attributes. The attributes are listed in the slide.

The *groupname* in *groupname*.res is the name of the NetWorker group that is currently backing up and to which the client belongs. The *groupname*.res file may be created and modified using any text editor. If the *groupname*.res file does not exist, a default groupname.res file (shown in the slide) is created the first time the client backs up. After the /nsr/res/*groupname*.res file is created, it may also be edited using the following command:

```
nsradmin -f /nsr/res/groupname.res
```

If the client belongs to multiple NetWorker groups, a separate *groupname*.res file is used for each group. For example, if a client belongs to two groups, **Default** and **astronomy**, the /nsr/res directory will contain the files Default.res and astronomy.res.

Important: Commands in the **precmd** and **pstcmd** fields must be specified by their full pathnames. Additionally, there must be a carriage return after the last line in *groupname*.res.
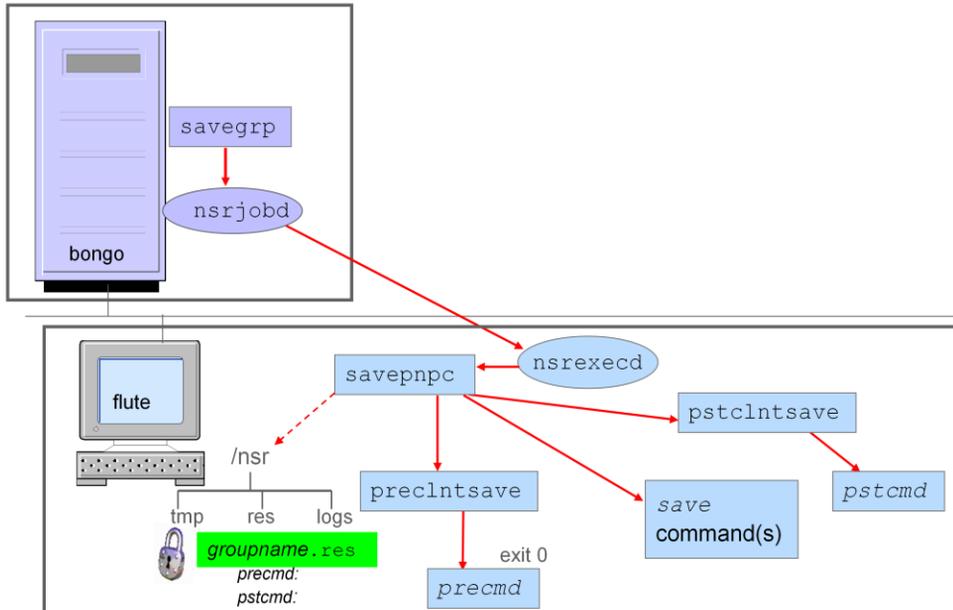
The backup process using `savepnpc` is illustrated in the slide and explained below.

1. `savegrp` contacts `nsrjobd` which tells `nsrexecd` to run the `savepnpc` process.

2. `savepnpc` starts `preclntsave`.

3. `preclntsave` locks `groupname.res` on the client to ensure that only one `savepnpc` can run for this group at any given time. NetWorker locks the file by placing three files in the `/nsr/tmp` directory:

   a. `groupname.lck`
   b. `groupname.res.lck`
   c. `groupname.tmp`

4. `preclntsave` executes the command specified in the **precmd** attribute.

5. If the **precmd** command exits successfully (exit status = 0), `savepnpc` starts the backup of all save sets. If the **precmd** command is not successful, the entire client backup is aborted.

6. If **pstcmd** is configured, `savepnpc` starts `pstclntsave`. If **timeout** is configured, `savepnpc` starts `pstclntsave` at the specified time, even if it causes the backup to fail.

7. `pstclntsave` executes the command specified in the **pstcmd** field. Success or failure of the command is stored in the client's `/nsr/logs/savepnpc.log` file. Non-redirected output from the command is discarded. NetWorker also removes the lock on the `groupname.res` file by removing the `groupname.tmp` file in the `/nsr/tmp` directory.

Note: Any non-redirected output from **precmd** is included in the save group completion report. Non-redirected output from **pstcmd** is discarded. The client's `/nsr/logs/savepnpc.log` file records the success or failure of **precmd** and **pstcmd** commands. Look there first when troubleshooting.

Module 4: Performing Backups

Lesson 4 Summary

During this lesson the following topics were covered:
- Backup command attribute on the client resource
- Using the savepnpc command

This lesson covered the **Backup command** attribute on the client resource.

Module 4: Performing Backups

Lesson 5: Performing Manual Client-initiated Backups

During this lesson the following topics are covered:
- Description of a client-initiated backup
- How to run client-initiated backups using NetWorker User and save

EMC²

This lesson covers backups initiated from a NetWorker client, including how to use the client-initiated backup utilities, NetWorker User and save.

Client-initiated Backups

Server

Tracking Data

Tracking Data

save

Client

Save Set

Save Set

Volume

Storage Node
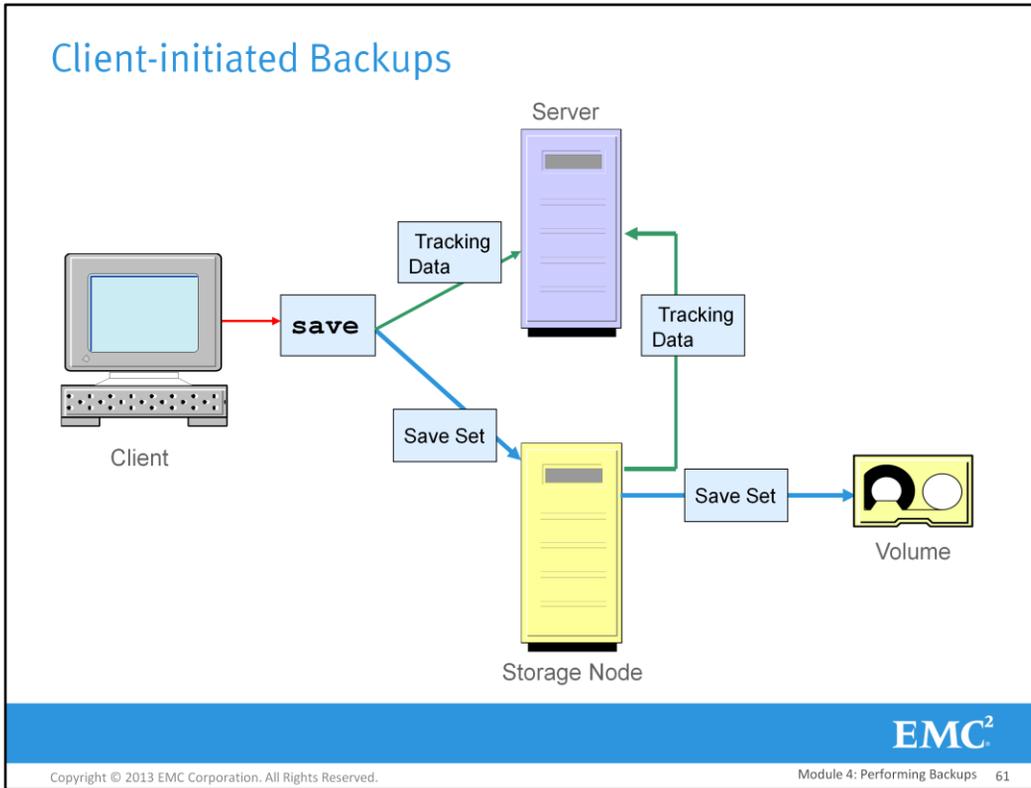
A client-initiated backup is a manual process performed on a NetWorker client from either a GUI or the command line. This type of backup is useful any time you want to perform a backup outside of the regular backup schedule. The user specifies which files, directories, and file systems to save. Client-initiated manual backups have a backup level of "manual" instead of the backup levels of full, incremental, etc..

Although the NetWorker server does not initiate a client-initiated backup, it manages the backup after the client makes a request. This management includes authorizing the backup and determining which storage node and backup device the client should send its save stream to.

For a client to execute any type of backup, it must first be configured as a client resource on the NetWorker server. This can be as simple as specifying the client host name in a client resource and using default values for all attributes.

When the client performs a `save`, it generates a save stream, sends it to the assigned storage node, and sends tracking information to the NetWorker server. The storage node also generates tracking information which it sends to the server.

On Microsoft Windows clients, client-initiated backups can be performed using the NetWorker User (`winworkr.exe`) graphical user interface. The `save` command can be executed directly from the command-line on any NetWorker client.

NetWorker User – Microsoft Windows
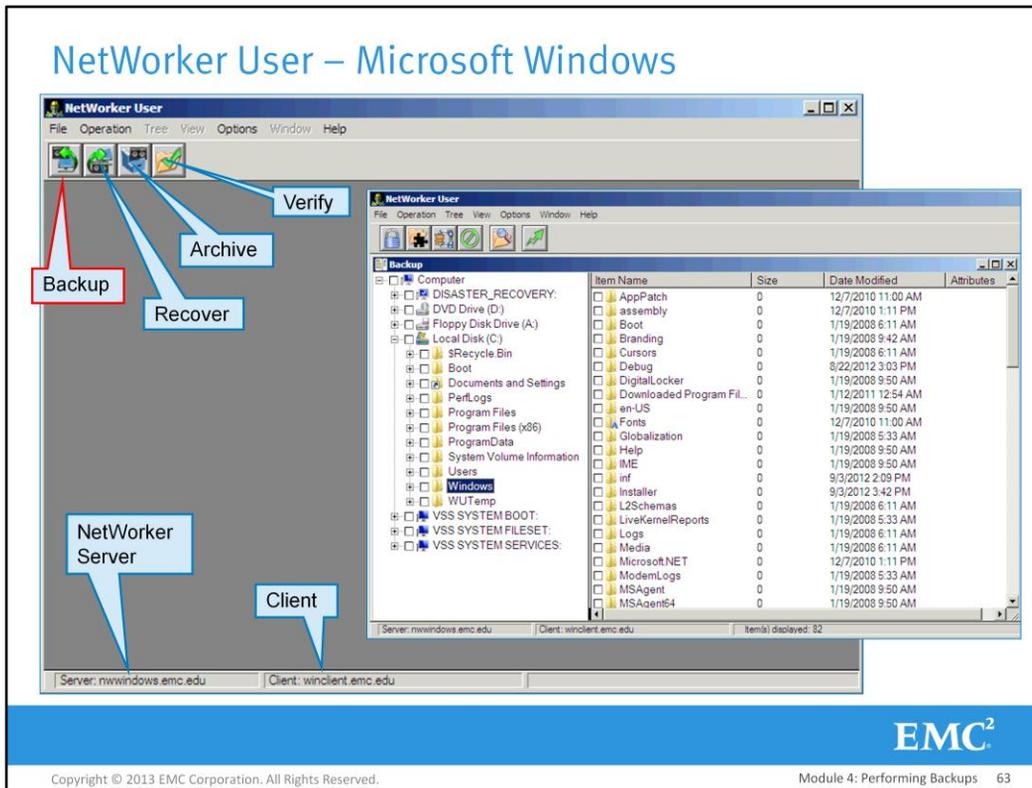
NetWorker User is used to perform both saves and recoveries. It can be started via **Start** > **Programs** > **EMC NetWorker** > **NetWorker User** or by executing `winworkr.exe` on the command line.

The four buttons in the upper-left corner of the window initiate the following tasks:

Perform a backup – This opens the smaller backup window shown in the slide.

Perform a recovery – This opens a recovery window and is discussed in the modules dealing with recoveries.

Perform an archive – This requires a special license and is not covered in this course.

Verify files – This allows you to verify whether a recent backup or archive operation was successful by comparing data on disk to data on a volume. See the *EMC NetWorker Administration Guide* for details.

The client name and NetWorker server managing the backup or recovery are shown at the bottom of the NetWorker User screen.
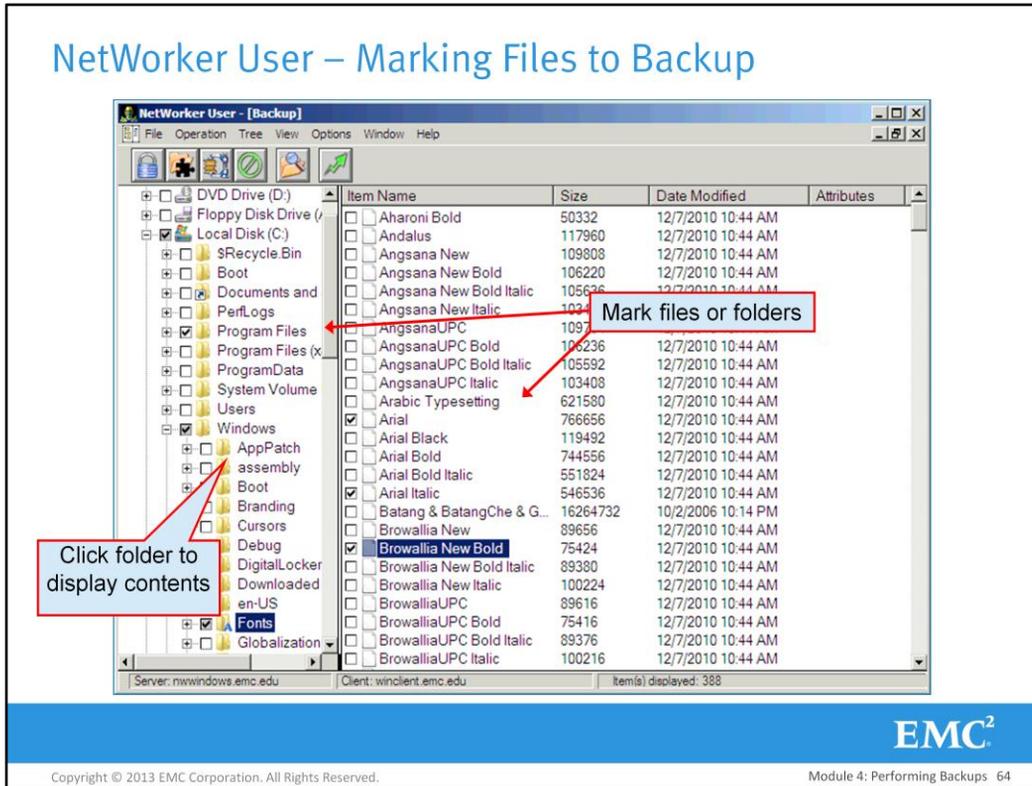
From the **Backup** window, files and folders are marked for backup. Folders are displayed in the left pane. Clicking a folder displays its contents in the right pane. Items can be marked for backup in either pane.

NetWorker User – Starting the Backup

After marking the files and directories to backup, click **Start** (green lightening bolt) to begin the backup. You can monitor the backup in the *Backup Status* window, which opens as soon as the backup begins.

NetWorker User – Special Handling
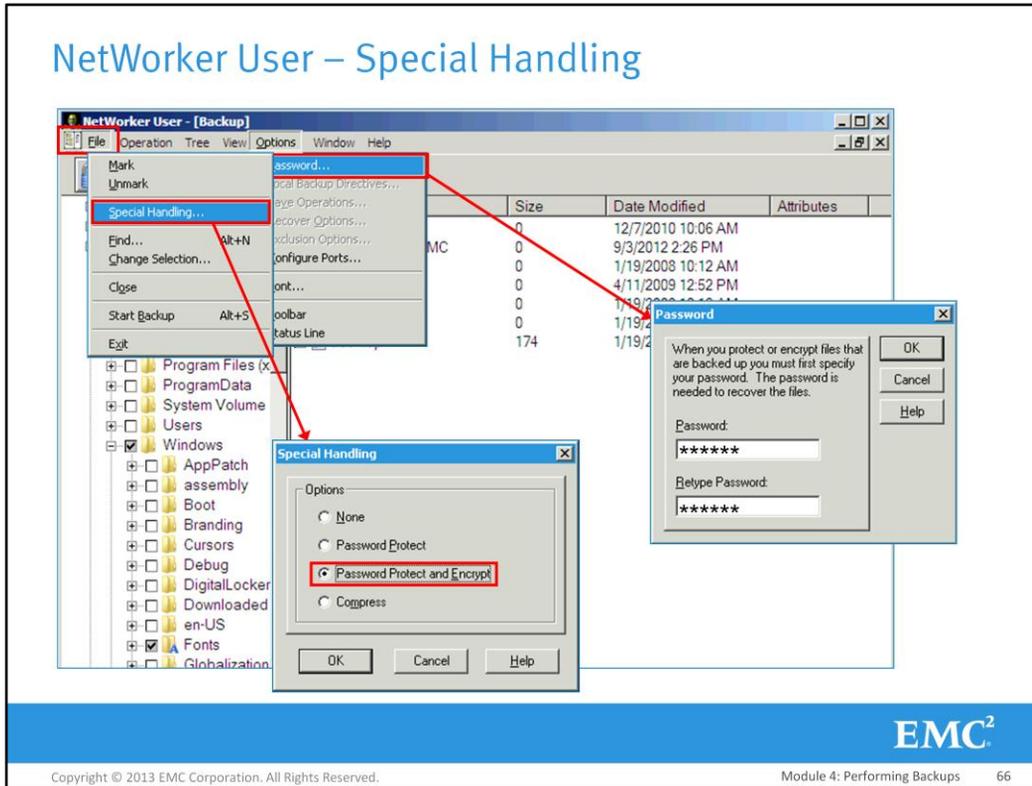
NetWorker User can be configured to perform software compression when generating the save stream and to password protect and encrypt the data using PW2 encryption. These capabilities are set in the *Special Handling* window which is opened via the **File** menu.

A password must be set before password protection or encryption can be performed. This is done by selecting **Options** > **Password** and entering a password in the pop-up window.

Using Special Handling affects all the files backed up during the backup session. To perform compression, password protection, or encryption on selected files, right-click the item you want to handle specially and select the appropriate action from the pop-up menu. The **Attributes** column shows the special handling that is currently set.

Important: When choosing a password option, DO NOT FORGET THE PASSWORD!!! It is not stored anywhere other than the volume on which the data is written. During recovery of PW2 encrypted data items, you are prompted for the password. If you cannot provide it, you cannot recover the files! If backup data is password-protected but not encrypted, an administrative user (root or Administrator) is able to recover the data.

The `save` Command

Option	Description
−s *server*	Specify which NetWorker server to use. The default server is the current host, if it is running `nsrd`. Otherwise, the host with the logical name "nsrserverhost" in the host table is used.
−n	No save. Do not perform a save, but estimate the amount of data that would be generated by the save.
−v	Verbose mode. Cause the `save` program to tell you in great detail what it is doing as it proceeds.
−q	Quiet mode. Display only summary information and error messages.
−b *pool*	Specify the destination pool for the save.
−x	Cross mount points.

Example:

`save /etc`

You can:
- Specify the files to back up
- Preview the backup
- Start the backup
- Monitor the backup

save is the NetWorker command-line utility (CLI) used to back up files and directories. It creates a single save set containing the files and directories specified as arguments. If no files or directories are provided as arguments, the current directory is backed up.

The syntax of the `save` command is:

```
save [ -option ... ] [ -option  argument ... ] [ path ... ]
```

Unless the −x option is used, `save` will not cross mount points. For example, **save /** in a UNIX environment backs up only the root file system.

Please refer to the *EMC NetWorker Command Reference Guide* for additional options and information about `save`.

<u>Note</u>: In a Microsoft Windows environment, **save −x /** causes all drives to be backed up.

<u>Important</u>: Caution should be exercised when using the **−x** option because `save` traverses network-mounted (NFS, CIFS) file systems and drives.

Previewing and Performing a Backup using `save`

```
C:\Users\administrator.EMC>save -n "C:\WUTemp\Common Files\VMware\Drivers\video"

3817:save: Using nwwindows.emc.edu as server
66135:save: NSR directive file (C:\NETWORKR.CFG) parsed
C:\WUTemp\Common Files\VMware\Drivers\video\install.txt
C:\WUTemp\Common Files\VMware\Drivers\video\vmwogl32.dll
C:\WUTemp\Common Files\VMware\Drivers\video\vmwogl64.dll
C:\WUTemp\Common Files\VMware\Drivers\video\vmx_fb.dll
C:\WUTemp\Common Files\VMware\Drivers\video\vmx_mode.dll
C:\WUTemp\Common Files\VMware\Drivers\video\vmx_svga.cat
C:\WUTemp\Common Files\VMware\Drivers\video\vmx_svga.inf
C:\WUTemp\Common Files\VMware\Drivers\video\vmx_svga.sys
C:\WUTemp\Common Files\VMware\Drivers\video\
32477:save:
save: C:\WUTemp\Common Files\VMware\Drivers\video 9 records 2 KB header 6024 KB
data

save: C:\WUTemp\Common Files\VMware\Drivers\video 6026 KB estimated
7167:save: save completion time: 9/4/2012 11:02:47 AM
```

```
C:\Users\administrator.EMC>save "C:\WUTemp\Common Files\VMware\Drivers\video"
3817:save: Using nwwindows.emc.edu as server.
66135:save: NSR directive file (C:\NETWORKR.CFG) parsed
C:\WUTemp\Common Files\VMware\Drivers\video\install.txt
C:\WUTemp\Common Files\VMware\Drivers\video\vmwogl32.dll
C:\WUTemp\Common Files\VMware\Drivers\video\vmwogl64.dll
C:\WUTemp\Common Files\VMware\Drivers\video\vmx_fb.dll
C:\WUTemp\Common Files\VMware\Drivers\video\vmx_mode.dll
C:\WUTemp\Common Files\VMware\Drivers\video\vmx_svga.cat
C:\WUTemp\Common Files\VMware\Drivers\video\vmx_svga.inf
C:\WUTemp\Common Files\VMware\Drivers\video\vmx_svga.sys
C:\WUTemp\Common Files\VMware\Drivers\video\
C:\WUTemp\Common Files\VMware\Drivers\
C:\WUTemp\Common Files\VMware\
C:\WUTemp\Common Files\
C:\WUTemp\
C:\

save: C:\WUTemp\Common Files\VMware\Drivers\video  6030 KB 00:00:03    15 files

7167:save: save completion time: 9/4/2012 11:04:41 AM
```

Previewing the backup does not actually back up any data. Running `save` with the `-n` option performs many of the tasks that take place during a normal backup, such as contacting the NetWorker server to request permission to back up. However, no save stream is generated.

Previewing the backup ensures that the `save` is working properly and displays an estimated size of the save set as well as the number of files to be backed up. A list of files that would be saved is also displayed.

Lab Exercise 4-5: Perform Client-Initiated Backups with NetWorker User

In these labs, you will be running manual backups using the NetWorker User interface.
- Lab Exercise 4-5: Perform Client-initiated Backups Using NetWorker User.

In this lab, you will:
- Perform client-initiated backups using NetWorker User.

Module 4: Performing Backups

Lesson 5 Summary

During this lesson the following topics were covered:
- Description of a client-initiated backup
- How to run client-initiated backups using NetWorker User and the save command

EMC²

This lesson covered backups initiated from a NetWorker client, including how to use the client-initiated backup utilities, NetWorker User and the save command.

Module 4: Performing Backups

Lesson 6: NetWorker Directives

During this lesson the following topics are covered:
- Types of NetWorker directives
- Application-Specific Modules (ASMs)
- The aes directive and the Datazone pass phrase attribute
- Save environment keywords

This lesson covers the various types of NetWorker directives that are used to perform optional tasks when performing a backup, such as skipping or compressing files.

NetWorker Directives

A *directive* is a set of statements used by the `save` command to perform optional tasks such as skipping or compressing of files.

- Server-side directive
  - Also called a "Global" directive
  - Is a NetWorker resource
  - Used only by server-initiated backups

- Client-side directive
  - Also called a "Local" directive
  - Is a text file named `.nsr` (UNIX) or `nsr.dir` (Windows)
  - Used by server-initiated and client-initiated backups
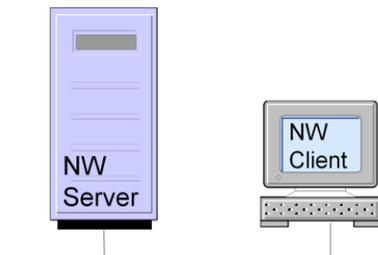
- NetWorker User local directive
  - Is a text file named `C:\Networkr.cfg`
  - Is created using **NetWorker User**
  - Used by server-initiated and client-initiated backups

NW Server

NW Client

A *directive* is a set of statements and arguments that the `save` command uses when generating a save stream. Directives allow you to perform optional tasks such as skipping or compressing files.

There are three types of directives:

- A *server-side directive* (also called a global directive) is a NetWorker resource with directive statements as its attributes. Server-side directives are used only by server-initiated backups.

- A *client-side directive* (also called a local directive) is a text file named `.nsr` (UNIX) or `nsr.dir` (Windows) that contains directive statements. The `save` command always looks for a directive file in a directory before backing up the directory. This type of directive affects both server-initiated and client-initiated backups.

- A *NetWorker User local directive* (Microsoft Windows only) is created from NetWorker User by a user logged in with local Windows Administrator privileges. This type of directive resides in a `networkr.cfg` file located at the root of the system volume (usually `C:\`). The syntax of this type of directive is identical to a server-side directive. A NetWorker User local directive affects both server-initiated and client-initiated backups.

NetWorker Directive Resource
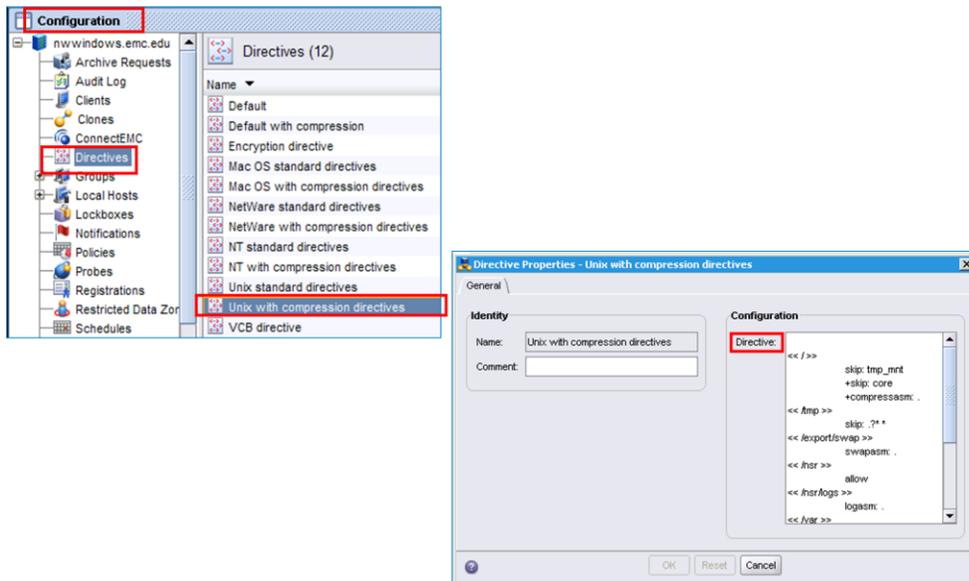
A server-side directive is a NetWorker resource that resides on the NetWorker server. There are a number of preconfigured directives for various operating systems.

The syntax for a directive can include:

- Directory specifications
- Application-specific modules (ASMs)
- Patterns used for filename matching
- Save environment keywords

Directory Specifications

A directory specification indicates which directory the ASMs or save environment keywords are applied to. A single directive resource may contain multiple directory specifications.

In the following example, the `skip` ASM applies only to files or directories in `/etc` whose names end in `.log`.

```
<< /etc >>
    skip: *.log
```

<u>Note</u>: In a client-side directive, a directory specification is optional. If it is omitted, the ASMs are applied to the directory containing the directive file.

If a directory specification is used in a client-side directive, it is resolved relative to the directory containing the `.nsr` or `nsr.dir` file.

For more information on directives, see the `nsr_directive` (for server-side) and the `nsr` (for client-side) topics in the EMC NetWorker Command Reference Guide or the UNIX/Linux man pages.

Application-Specific Modules (ASMs)

ASM	Function
aes	Perform encryption, using the NetWorker server's **Datazone pass phrase**, on file data when generating the save stream.
always	Save the files regardless of the backup level.
compressasm	Perform software compression on the data being backed up.
logasm	Do not generate a warning for files that change in size during backup.
rawasm	Saves a raw partition. It is commonly used to back up an inactive database partition. rawasm follows symbolic links (/dev).
skip	Skip the specified files and directories.
swapasm	Used to avoid saving the contents of a swap file. During recovery, a zero-filled swap file of the proper size is created.

Application-specific Modules (ASM)

An **application-specific module** (ASM) is the part of the directive which directs the save command to back up certain files in a non-standard way, back up files or directories that would not normally be backed up, or omit certain files or directories from the backup.

Examples of ASMs include compressasm, which compresses the files or directories in the save set, and skip, which omits file or directories from the backup. Some of the available ASMs are listed in the slide. A full list of ASMs is available in the uasm topic in the EMC NetWorker Command Reference Guide and the UNIX man pages.

Patterns

ASMs are applied to a whitespace-separated list of *patterns* (files or directories) specified on the same line as the ASM. The patterns can include wildcards (*, ?, [], .) but cannot include pathnames (\ or /). An optional "+" before the ASM causes the ASM to be applied recursively to subdirectories and their contents.

Examples of Directives

1. Skip the file expenses.xls in the C:\docs directory, and compress all files having a .mdb extension and residing in C:\docs and recursively below it.

   ```
   << "C:\docs" >>
       skip: expenses.xls
       +compressasm: *.mdb
   ```

2. Skip all files with .tmp and .jpg extensions anywhere under /opt/data.

   ```
   << /opt/data >>
       +skip: *.tmp *.TMP *.jpg *.JPG
   ```

AES - The "Datazone pass phrase" Attribute
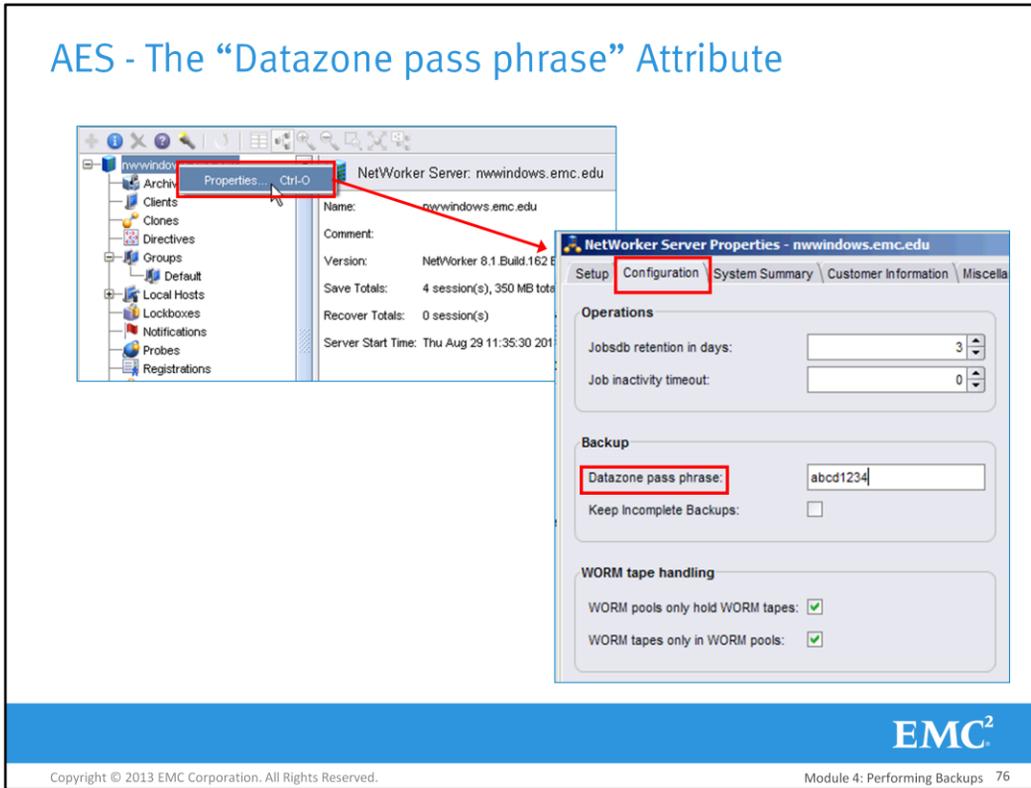
NetWorker Server: nwwindows.emc.edu

Name: nwwindows.emc.edu

Comment:

Version: NetWorker 8.1 Build.162

Save Totals: 4 session(s), 350 MB tota

Recover Totals: 0 session(s)

Server Start Time: Thu Aug 29 11:35:30 201

NetWorker Server Properties - nwwindows.emc.edu

Setup | Configuration | System Summary | Customer Information | Miscella

Operations

Jobsdb retention in days: 3

Job inactivity timeout: 0

Backup

Datazone pass phrase: abcd1234

Keep Incomplete Backups: ☐

WORM tape handling

WORM pools only hold WORM tapes: ☑

WORM tapes only in WORM pools: ☑

The **aes** directive uses the AES (Advanced Encryption Standard) algorithm to encrypt data on the NetWorker client as it is being backed up.

The AES algorithm uses the value of the **Datazone pass phrase** attribute, located in the NetWorker server resource, to encrypt the data. If **Datazone pass phrase** does not have a value when AES encryption is performed during a backup, a default pass phrase, hard-coded in the NetWorker software, is used.

The pass phrase should be difficult to guess and should contain a mixture of uppercase and lowercase letters, digits, and special characters.

Important: The current pass phrase is maintained as a "clear-text" resource attribute value and is therefore, by default, viewable by any user having access to the NetWorker resource database. AES encryption is not intended to protect data from an individual having access to your network. Instead, it is intended to prevent an individual who has gotten unauthorized access to a piece of NetWorker media from being able to read the data on the media.
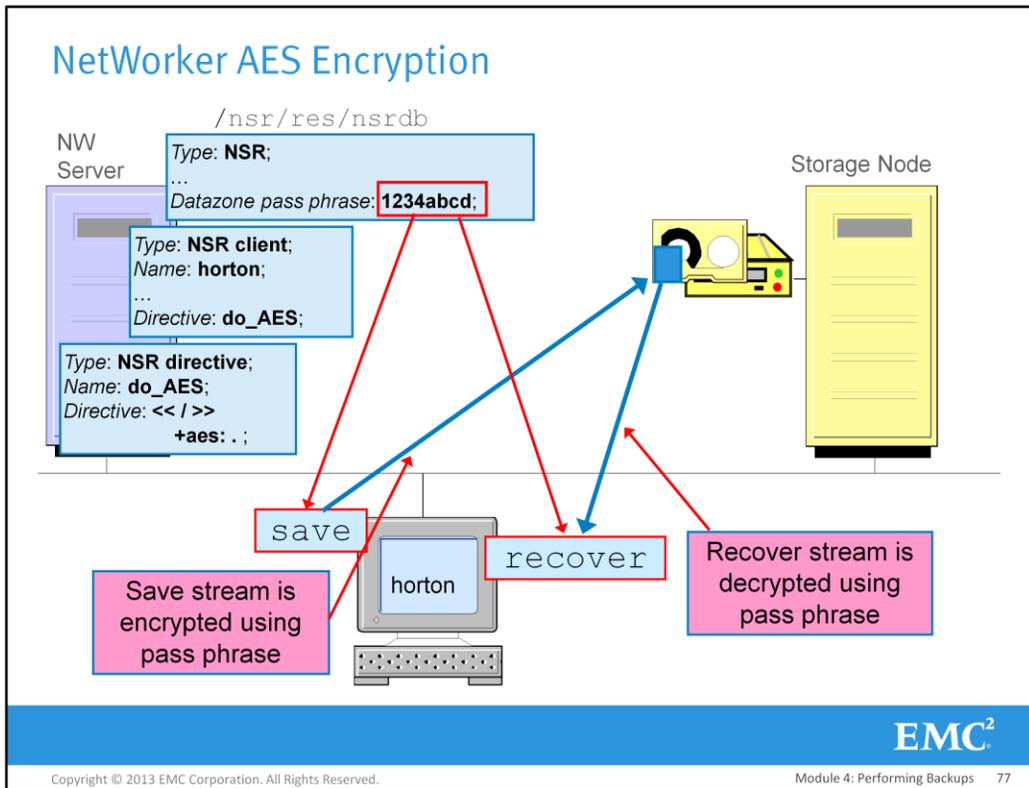
During a backup of a client using the **aes** directive, the NetWorker server provides the value assigned to its **Datazone pass phrase** to the client being backed up. The pass phrase is used to encrypt the data on the client as the save stream is generated. If the **Datazone pass phrase** attribute has no value, the default pass phrase is used.

During recovery of encrypted data, the value of the **Datazone pass phrase** attribute is used for decryption. If decryption fails, another attempt to decrypt the data is made using the default pass phrase. If that decryption fails, the recovery will fail.

To recover data backed up when a different pass phrase was in effect, you must run the recover program with the $-p$ option and specify the pass phrase in effect at the time of the backup. For this reason, it is suggested that the pass phrase not be changed.

Important: If a file recovery fails because the data cannot be decrypted, an error message is displayed and a null-filled file, having the same size as the actual file, is created.

Save Environment Keywords

Keyword	Description
forget	Forget recursive ASMs ("+") currently in effect.
ignore	Ignore .nsr and nsr.dir files in subsequent directories
allow	Use .nsr and nsr.dir files in subsequent directories (default)

SYNTAX

<< "directory_specification" >>
save_environment_keyword

Save environment keywords are used to affect how current ASMs, as well as ASMs further down in the directory structure, are to be applied.

In the following example, software compression is being (recursively) performed on all files under the root directory. However, the **forget** keyword says, "Stop applying any ASMs that are currently in affect" and **ignore** says "Ignore all .nsr files located in or below /export/home."

```
<< / >>
    +compressasm: .
<< /export/home >>
    forget
    ignore
```

The result is that nothing under /export/home is compressed and all .nsr files under /export/home are ignored. Thus, even if a user has a directive file /export/home/bubba/.nsr containing:

```
skip: image_data
```

which was created to avoid backing up a directory of large images that can be easily recreated, the directory is actually backed up because the client-side directive file is ignored.

You could use both **ignore** and **allow** together to correct the situation. To allow only Bubba to have a .nsr file, add a 2nd directive statement, allowing only the .nsr file in Bubba's directory. The directive resource now looks like this:

```
<< / >>
    +compressasm: .
<< /export/home >>
    forget
    ignore
    +compressasm
<< /export/home/bubba >>
    allow
```

Creating a NetWorker User Local Directive File
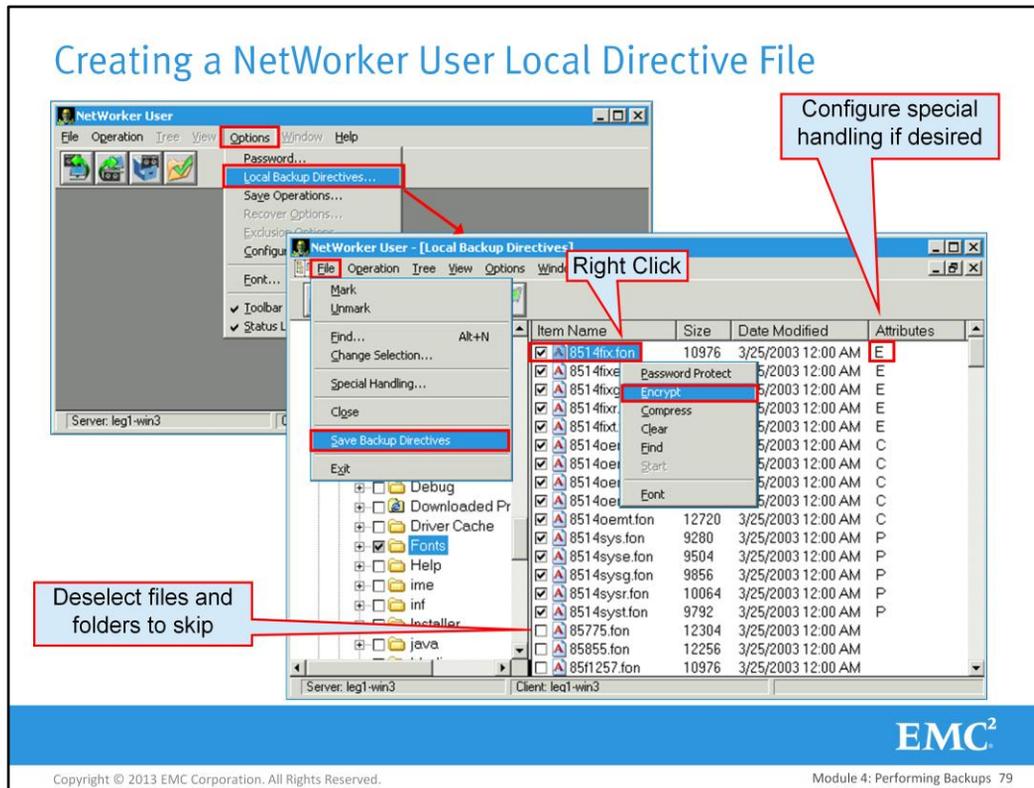
The purpose of configuring local directives using NetWorker User is to avoid having to manually edit a `nsr.dir` file and worry about using correct syntax. Using NetWorker User simplifies the creation of the directives.

This type of directive has limitations. It can only configure ASMs that NetWorker User is familiar with. These include `null` (similar to `skip`), `compressasm`, `pw1` (password-protect), and `pw2` (encrypt).

To configure the directives, start NetWorker User and select **Options** > **Local Backup Directives**. All files and directories are initially marked. Unmark files and directories you want skipped during backups, and apply special handling to those items for which you desire special handling. Save the directives by selecting **File** > **Save Backup Directives**. The `networkr.cfg` file is created and read by `save` during subsequent backups. If the file already exists, it is updated each time you save the directives.

`networkr.cfg` resides at the top level of the system volume (usually `C:\`).

Lab Exercise 4-6: Using NetWorker Directives

In this lab, you will configure a directive resource on the NetWorker server to simulate a real-world use scenario.

In this lab, you configure a directive resource on the NetWorker server to skip certain files in a directory. You will configure your client to simulate the skipping of large open database files that would be backed up using a NetWorker application module and thus would not need to be backed up during filesystem backups

Module 4: Performing Backups

Lesson 6 Summary

During this lesson the following topics were covered:
- Types of NetWorker directives
- Application-Specific Modules (ASMs)
- The aes directive and the Datazone pass phrase attribute
- Save environment keywords

This lesson covered the various types of NetWorker directives that are used to perform optional tasks when performing a backup, such as skipping or compressing files.

Module 4: Performing Backups

Lesson 7: Backing Up Virtual Clients and Applications

During this lesson the following topics are covered:
- Ways to backup VMware virtual machines with NetWorker
- Overview of NetWorker modules for backing up applications and databases
- Block Based Backups

This lesson covers an overview of the ways to backup VMware virtual machines and the NetWorker modules for backing up applications and databases. Additionally an overview of Block Based Backups is provided.

NetWorker Integration with VMware

- NetWorker provides support for backup and recovery of clients in VMware environments using:
  - Guest-based backup and recovery.
  - Image level backup and recovery using VADP.
- NetWorker Administration provides central management of the backup environment including automatic discovery of VMware environments and notification of changes in those environments.
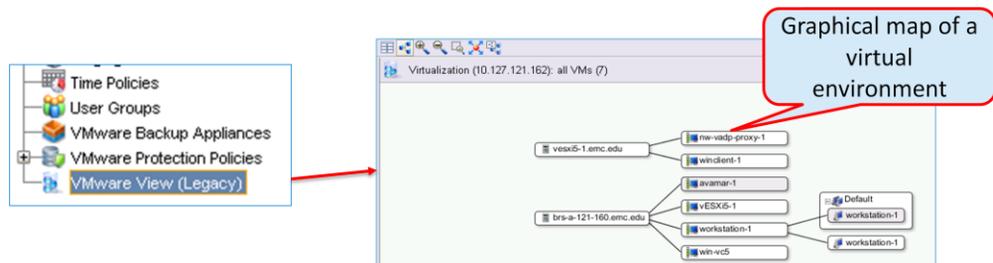
NetWorker provides support for backup and recovery of VMware virtual clients using traditional methods to back up guest machines or by using VMware backup technologies such as the vStorage APIs for Data Protection (VADP) for image level backup and recovery. Additionally, NetWorker provides automatic discovery of VMware environments and notification of changes to those environments. Both a graphical map and tabular view of VMware environments are provided.

With a guest backup, the virtual machine is treated like a normal client with the NetWorker client software installed on the host machine. Using VMware vStorage APIs for Data Protection (VADP), vSphere virtual machines can be backed up without using backup agents inside the virtual machines – the machines being backed up are not impacted.

Guest Backup

- Individual backup for each virtual machine.
- Virtual machines treated like any other physical machine.
- NetWorker client software is installed on each guest (virtual machine)
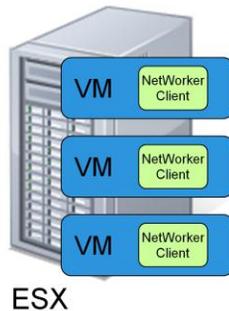
With the guest backup option, NetWorker client software is installed on the individual virtual machines. Backup configuration for this method is identical to that of a physical client host. An advantage of VMware guest backup is that it lets backup administrators leverage identical backup methods for physical and virtual machines. Also, the advantages of using NetWorker modules for backing up databases and other applications, such as application consistency, are realized with guest-based backups and may not be available with image-level backups.

There is no requirement for advanced scripting or VMware software knowledge and it means unchanged day-to-day procedures for backup. Also, the recovery process is the same as for a physical host. However, since each virtual machine has a separate NetWorker client installed, ESX servers with a large number of virtual machines may experience a strain on resources, especially memory, if all machines are backed up at the same time.

Backup policies should be defined to limit the number of simultaneous backups jobs that are running on each physical ESX server in order to minimize the impact on individual ESX servers.

VMware VADP Image Backup (NW 8.0 and earlier)

- NetWorker uses VMware VADP to perform image backups of virtual machines.
- Reduces load on ESX server and virtual machines during backups.
- VMware Changed Block Tracking prevents backup of unused/unchanged vmdk blocks.
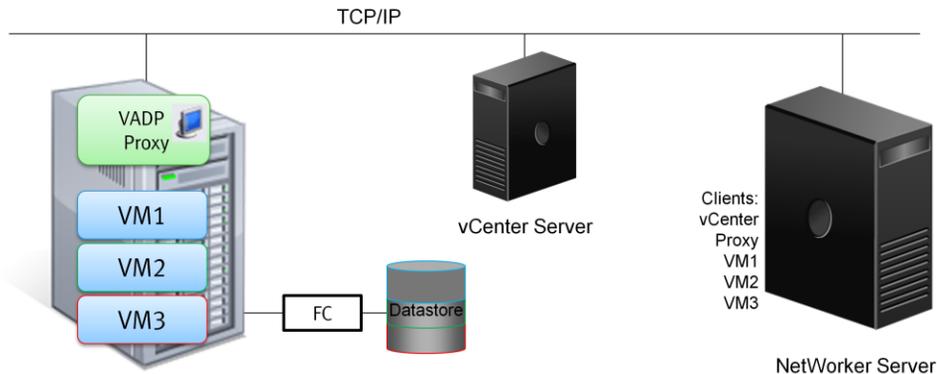
NetWorker supports the VMware image backup method for backups of virtual machines. NetWorker uses VMware vStorage APIs for Data Protection (VADP) to backup virtual machines without requiring backup processing to be done inside individual guest machines. It is not necessary to install NetWorker client software on each virtual machine being backed up.

With NetWorker VADP Image backups, backups are performed using a VADP backup proxy server, which can be a physical or a virtual machine, using VMware snapshot technology. This is accomplished by creating a snapshot of the virtual machine and then backing up that snapshot. No downtime is required for the virtual machine. Backup processing can be offloaded from the ESX server to the VADP proxy server.

Image level backups are supported for virtual machines running any guest operating system supported by VMWare. File level backups are supported for virtual machines running a Microsoft Windows guest operating system. The ability to recover individual files from an image level backup is supported for Windows NTFS only.

VADP backups have the option of using VMware Changed Block Tracking. This is a method of keeping track of which blocks in the virtual disks have been changed since the last backup. This can greatly improve backup time by making it unnecessary to scan through the entire vmdk file.

Please refer to the latest *EMC NetWorker VMware Integration Guide* for supported vCenter and ESX versions, detailed capabilities, functions, and configuration information.

VMware Backup Appliance (NW 8.1+)

- VMware Backup Appliance
  - Optimized backup and recovery solution for VMware enviornments
- Provides centralized backup admin in vCenter web client
  - Centralized backup administration (configuration, monitoring, scheduling, save set retention, and logging)
    - Protection of specific VMs
    - Auto protection of VMs by membership in containers (ESXi, Clusters, …)
    - Incremental forever backups using CBT
    - DR of VMs as a single step from any mounted volume
    - File level recovery for NTFS, EXT2, EXT3
    - Backup to internal storage and/or Data Domain
    - Cloning/Staging from DD to DD or tape/disk
    - Backup/Clone retention
  - Enterprise Scalability: 10,000 VMs per Data Zone

EMC²

The VMware Backup Appliance is an optimized backup and recovery solution for VMware environments. It is able to provide protection to hundreds of VMs ensuring the protection of the largest of virtual environments. The VMware Backup Appliance can be managed and monitored centrally through the vSphere web client. This solution provides the administrator with the ability to backup VMs to the VMware Backup Appliance's internal storage as well as to Data Domain. Most administrators will want to backup to Data Domain for performance and scalability reasons

Application Backups with NetWorker Modules

- NetWorker Module for Databases and Applications
  - Back up DB2, Informix, Lotus Notes, Oracle, Sybase using a single module
- NetWorker Module for Microsoft Applications
  - Uses VSS technology for backing up Exchange, SQL, SharePoint, DPM, Hyper-V, and Active Directory
- Other NetWorker modules include:
  - NetWorker Module for SAP
  - NetWorker Module for MEDITECH

EMC NetWorker application modules act with third-party applications/databases, in conjunction with NetWorker, to provide a comprehensive data storage management system. NetWorker modules allow applications to be backed up in a consistent state.

NetWorker application modules fully integrate with third-party vendor-specific APIs, eliminating the need to develop or maintain custom backup and recovery scripts. They provide fast, online, automated, and reliable granular backup and recovery for popular database, messaging, content, and ERP applications.

NetWorker application modules are listed on the slide and include:

- **NMDA**: EMC NetWorker Module for Databases and Applications (NMDA) delivers a unified backup solution for various databases and applications. NMDA software works with the supported database or application software and NetWorker software to provide backup and restore services for DB2, Informix, Oracle, Lotus Domino/Notes, and Sybase data.

- **NMM**: EMC NetWorker Module for Microsoft Applications (NMM) delivers a unified backup solution for Microsoft applications. NMM works with Microsoft Volume Shadow Copy Service (VSS) technology to provide snapshot backup and recovery services. NetWorker users can take advantage of VSS to protect Microsoft applications and manage the scheduling and retention of shadow copies as well as the rollover of those snapshots to disk, tape or virtual tape for secondary protection. Beginning with NMM release 2.4, NMM provides two methods of backup and recovery of SQL Server 2005, 2008 and 2012: using Microsoft VDI to communicate with the SQL Server or using VSS through Microsoft-supplied application writers.

Block Based Backups - Overview

- Block Based Backup BBB
  - NetWorker 8.1+
  - Integrated into NetWorker client
    - Replaces previous SnapImage module
  - Allows for creation of block based backup images
  - Leverages Changed Block Tracking for incremental backups
    - Requires client reboot on installation to allow incremental BBB backups
  - Allows for Image and File Level recovery

EMC²

Block based backup is an integrated feature in the Windows NetWorker client that replaces the previous support offered with the separately purchased SnapImage Module. It speeds up the backup of workloads such as high density file systems or very large files that incur only very minor and infrequent changes. It does this by taking an image-based backup at the volume level, rather than walking the entire file system in the backup process. Incremental backups are created using a changed block tracking methodology, thus reducing the amount of data that is backed up during an incremental backup. By using Block Based Backup for Windows, file system backups completes up to 5x faster than a traditional file backup, and recovery operations are 2-4x faster.

In addition, no index is created as part of this workflow. This is particularly of benefit for high density file systems where, potentially, millions of files would need to be indexed and indexed again with every backup. The fact that NetWorker does not create an index in this process is a differentiator in the industry. It saves time and space in the backup workflow. Even though an index is not created, recovery at the file level is still supported. This is done by virtually mounting the back up, at which point files can be viewed and recovered. While optimized for backup to disk, the customer using Block Based Backup can also clone to backup media for longer term storage.
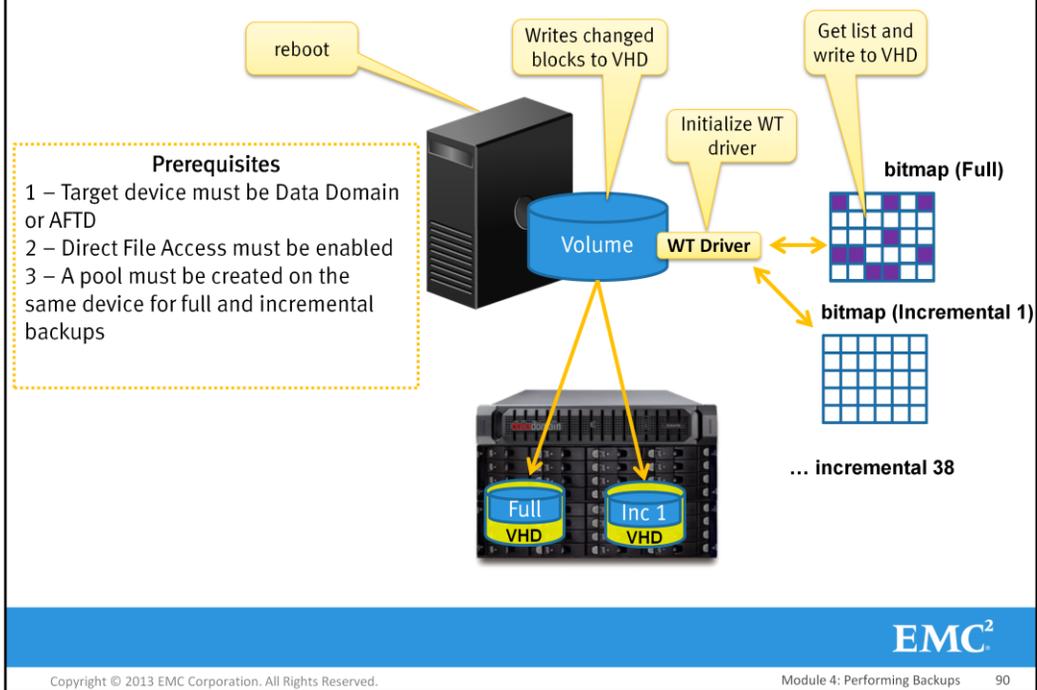
Block Based Backups – Client Configuration

Block Based Backups Require Client Direct

Block-Based Backups are configured by selecting the checkbox on the properties of the client. When using Block Based backups, Client Direct is mandatory and is checked automatically.

BBB Architecture: Backup Overview

reboot

Writes changed blocks to VHD

Get list and write to VHD

Initialize WT driver

Prerequisites
1 – Target device must be Data Domain or AFTD
2 – Direct File Access must be enabled
3 – A pool must be created on the same device for full and incremental backups

Volume   WT Driver

bitmap (Full)

bitmap (Incremental 1)

… incremental 38

Full VHD   Inc 1 VHD

A reboot is required upon installation of the block based backup feature before an incremental BBB can be performed. The reboot is required to initialize the write tracker (WT) Driver. The write tracking driver maintains a bitmap to track changes (write I/O requests) for every block based backup volume (Fulls and incrementals). The initialization of the write tracking driver enables the incremental backup functionality which is part of block based backups.

A full backup must be created initially. NetWorker uses VSS (Microsoft Volume Shadow Service) to create a Snapshot of the volume being backed up. It then checks the block based backup prerequisites to ensure a block based backup can be created. If the prereqs are satisfied, NetWorker creates the VHD container on the target device. It then obtains the list of used blocks in the volume and writes the blocks to the VHD container. After the blocks have been written it updates the MMDB with the VHD information to be used for subsequent incremental backups.

Incremental backups must be created on the same device and must go through a similar process as full backups. Incrementals use Microsoft VSS to create the snapshot. For each incremental a new bitmap is created using the write tracking driver. NetWorker also checks the prerequisites for each incremental. If the prerequisites are met it creates a VHD container on the target device. Finally, the list of changed blocks from the previous level full is obtained and written to the VHD container. For every incremental backup the write tracker driver is queried to get the previous bitmap that contains the list of changed blocks. Up to 38 incrementals can be created. After the 38th incremental a full backup is forced.

Module 4: Performing Backups

Lesson 7 Summary

During this lesson the following topics were covered:
- Ways to backup VMware virtual machines with NetWorker
- Overview of NetWorker modules for backing up applications and databases
- Block Based Backups

This lesson covered an overview of the ways to backup VMware virtual machines and the NetWorker modules for backing up applications and databases. Finally, an overview of Block Based Backups was provided.

Module 4: Summary

Key points covered in this module include:

- Performing client-initiated and server-initiated backups .
- The data flow in a server-initiated backup.
- The relationship among the client attributes and certain resources.
- Configuring schedule, policy, group, directive and client resources.
- How to use backup scripts, including **savepnpc.**
- Backing up virtual clients and applications.

This module covered performing backups, including starting backups manually from the NetWorker server and from a NetWorker client. We also examined the resources involved in customizing and scheduling backups.