STORAGETEK®

APPLICATION NOTE

Using Shared Virtual Array® (SVA™) disk systems and SnapShot software with NetWorker for Windows 2000

APRIL 2004

HARDWARE AND SOFTWARE APPLICATIONS USED

	Name	Vendor	Comments
ISV	NetWorker	Legato	
Operation System	Windows 2000 – SP 2 – SP 3	Microsoft	
SVA Software	SVAA 3.1.0 patch 12 (W2K) SVAC 3.1.0 patch 11 (W2K) SVA Path™ software patch ?? (W2K)	StorageTek	Optional* **
SVA Hardware	SVA 9200, 9500 and V960	StorageTek	Tested on SVA 9500
Backup Software	NetWorker 6.1	Legato	
Backup Hardware	Quantum DLT drive DLT tape	Quantum	

* SVAC is the graphical version of SVAA. The use of the SVAC is optional; the devices can be created using the CLI.

** SVA Path software is the failover product for SVA disk system. The scripts included in this document contain SVA Path software command lines.

1 ABSTRACT

The V-Series Shared Virtual Array® (SVA™) disk system has a feature, called SnapShot software, which allows making instantaneous copies of any single virtual disk defined within the disk subsystem. In order to take advantage of all the benefits of this feature, it is necessary to integrate the SnapShot software operational steps with the common Relational Database Management System (RDBMS) used on the host systems. An RDBMS is a program that allows you to create, update and administer a relational database.

Today the Open Systems market is leading the disk storage market and all the integration efforts remain to be done in this arena. This application note will present a simple integration of SnapShot software with the product Legato NetWorker software. The tests performed here are based on a Windows environment. But, you can refer to the document "*Using SVA SnapShot with NetWorker*" (June 2001) for more information about the same scenario based on a Solaris platform.

We assume that the reader has a good knowledge of Windows disk administration, Legato NetWorker software, SVA principles and SVA administration.

The last part of this document describes the different tests of backup with SVA disk system and NetWorker.

All the scripts, prerequisites and constraints are also detailed in each test.

2 UNDERSTANDING SNAPSHOT SOFTWARE

SnapShot software is a powerful data duplication product that harnesses the unique virtual disk architecture of StorageTek's SVA disk system. SnapShot duplicates entire Open Systems Logical Units (LUNs) in seconds, using no additional physical capacity.

SnapShot software's virtually instantaneous copying speed is accomplished by manipulating pointers to data within the virtual disk subsystem, instead of physically moving it. With no physical data movement required, SnapShot software creates copies of LUN in approximately two to five seconds, no matter how much data is copied. SnapShot software also has the unique ability to create any number of copies that are needed or desired by the user. By eliminating the use of resources such as central processing unit (CPU) cycles, channel utilization, I/O operations, and most importantly time, SnapShot software creates a new paradigm in data duplication. Traditional data duplication products require physical data movement, which is expensive in terms of resources used and precious time.

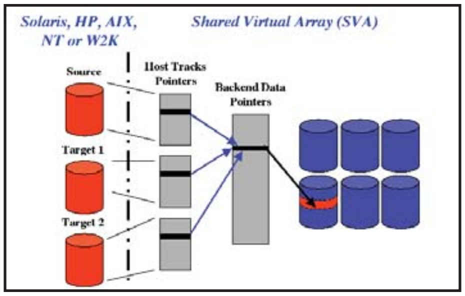


Figure 1. Snapshot mechanism.

2.1 HOW DOES IT WORK?

SnapShot software simply updates pointers within the virtual disk subsystem mapping tables for the data views being duplicated at electronic memory speeds. This updated pointer references to the same disk array locations as the original source data — one physical copy of the data with multiple host "views" (**Figure 1**).

SnapShot software copies use no disk storage initially, since only one physical copy of the data exists at the point in time when the snap is created. All copies and the original copy are totally independent, and can be updated changed or read without affecting each other. Only changes to either the source data or copies use additional physical storage, and any parts of the data that remain common are shared. The result of a SnapShot copy is the same as that of a traditional copy, the main difference is that the time required to create the copy with SnapShot software is measured in mere seconds versus traditional physical duplication products measured in minutes or hours.

3 ARCHITECTURE EXAMPLE

The aim of this section is to present the test environment for the SVA disk system in order to use the snapshot feature with the Legato NetWorker software. Two main architectures are considered:

> Using a simple SAN architecture with only one server that is the production and the backup server
> Using an additional backup server to perform a so-called "production server-free backup".

3.1 SINGLE-SERVER BACKUP

The figure below depicts the simple backup configuration. In this first configuration, the production server and the backup server are configured on the same machine. The aim of this section is to present an online/offline backup in a split mirror scenario with no additional backup server.

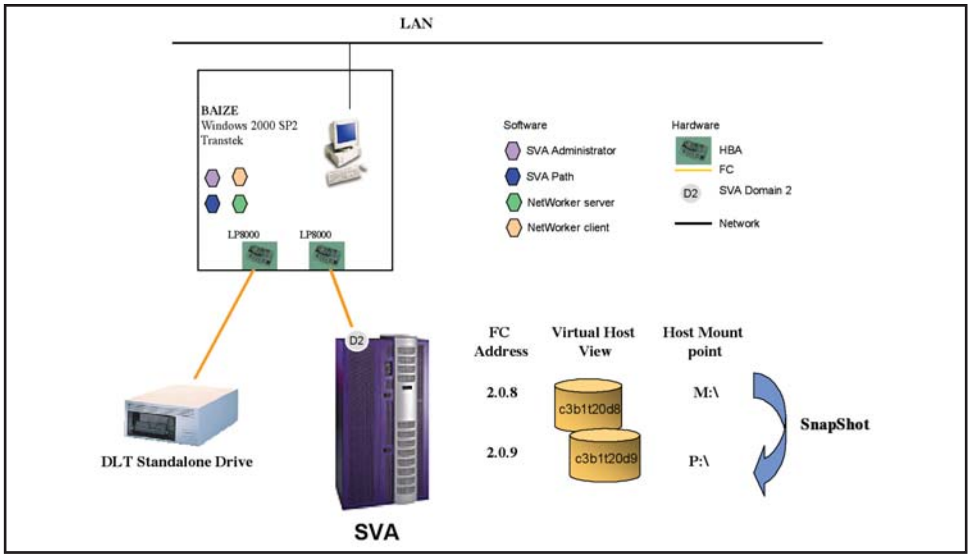


Figure 2. Single production/backup server configuration.

The production server (W2K BAIZE) uses one LUN of the SVA disk system to store data files. Another LUN is used to get the snapshot copy of the original data files. The server uses a Fibre Channel connection (QLogic QLA2200F) to communicate with the SVA disk system on the domain ten. It is configured as a NetWorker client.

The same server acts as the backup server as well. It is connected to the tape drive via a Fibre Channel connection (QLogic QLA2200F). So, it is configured as the NetWorker server.

The tape drive used is a stand-alone one using a DLT tape.

3.2 PRODUCTION SERVERLESS BACKUP

The aim of this section is to present the second test environment for the SVA disk system and the NetWorker software.

The figure below illustrates the basic architecture that intends to reproduce a typical schema with data production server and backup server:

> The production server (DELL NESTE) uses one LUN of the SVA disk system to store the data and another one to receive the SnapShot. A fiber channel connection is used to connect the SUN to the SVA disk system on domain ten (Emulex LP8000).

> The backup server (W2K BAIZE) has a fiber connection to the SVA disk system. Another HBA is used to make a connection to the stand-alone tape drive (via a SCSI bridge 3100). The backup server is configured as a NetWorker Master Server.

> Both servers are on the same LAN.

Note **All the software applications required to manage the SVA disk system are installed on both servers.**

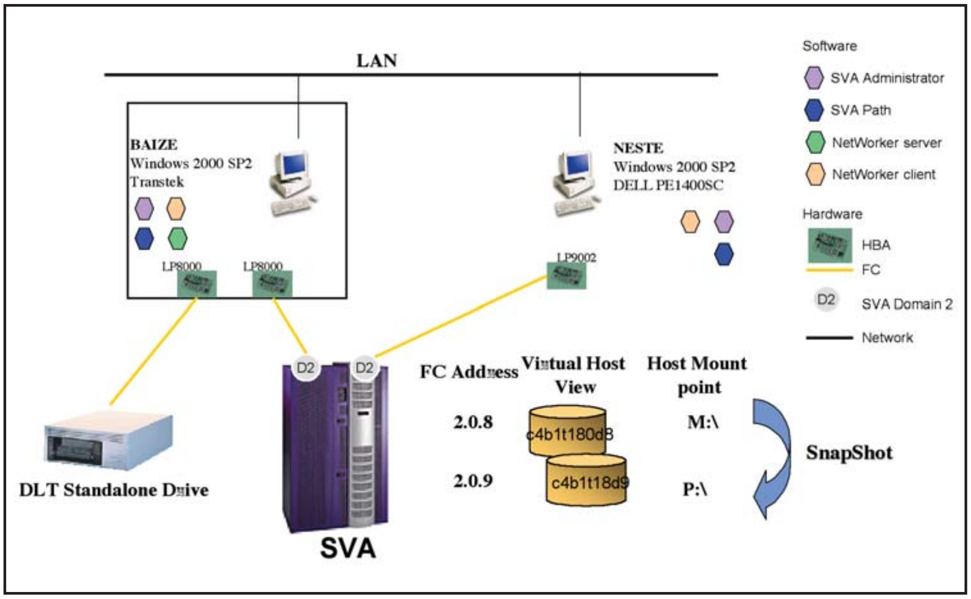


Figure 3. Dedicated backup server configuration.

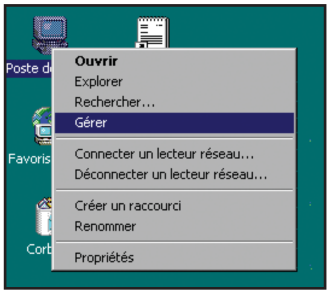
4 CONFIGURATION STEPS

4.1 SVA DISK SYSTEM CONFIGURATION

On the client server we must configure two devices on the SVA disk system, one for the customer data and one for snapshot.

Here we suppose that the target is always defined on the SVA disk system, but this device doesn't use physical capacity of the SVA disk system. Before the backup, a release command is executed in order to clean the device before the snapshot. After the backup, it is also possible to perform a release command on this to remove the capacity of potential data update after the snapshot. Otherwise, the "image" of the data is stored on the target LUNs just in case.

The SCSI partition release facility informs an SVA disk system that a given SCSI partition no longer contains any useful data so the subsystem can release the space the partition is using. Thus, partition release enables you to take better advantage of an SVA's extended capacity, to achieve lower overall storage costs, and at the same time to securely erase unwanted data.



4.1.1 Make sure the SVA disk system is correctly configured

Use the Manager console to display the current general configuration.

Note **The different LUNs that will be defined must be configured from the production server. The file systems for the original data are always mounted on the production server.**

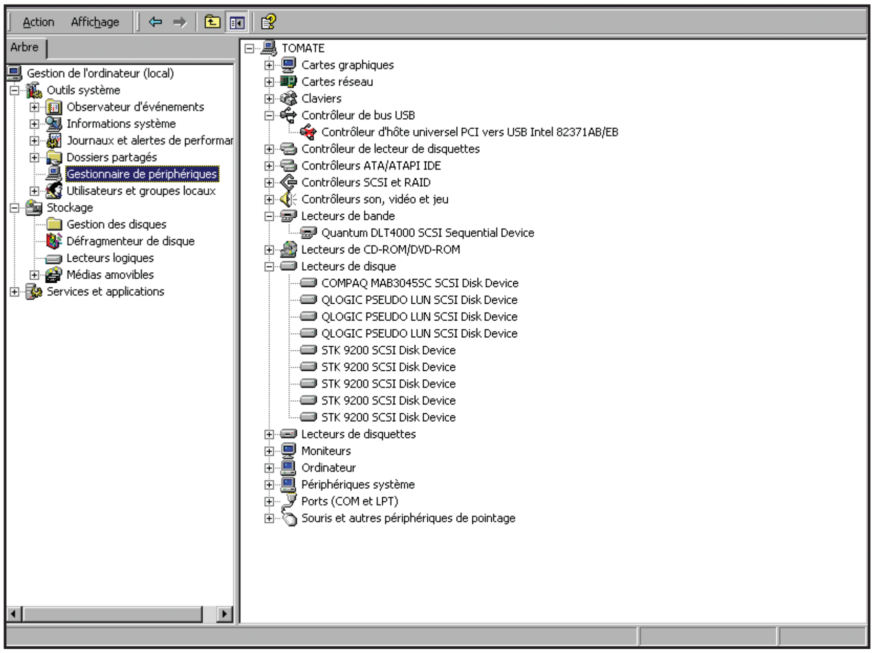

Action
Affichage
Arbre
Gestion de l'ordinateur (local)
Outils système
Observateur d'événements
Informations système
Journaux et alertes de performar
Dossiers partagés
Gestionnaire de périphériques
Utilisateurs et groupes locaux
Stockage
Gestion des disques
Défragmenteur de disque
Lecteurs logiques
Médias amovibles
Services et applications
TOMATE
Cartes graphiques
Cartes réseau
Claviers
Contrôleur de bus USB
Contrôleur d'hôte universel PCI vers USB Intel 82371AB/EB
Contrôleur de lecteur de disquettes
Contrôleurs ATA/ATAPI IDE
Contrôleurs SCSI et RAID
Contrôleurs son, vidéo et jeu
Lecteurs de bande
Quantum DLT4000 SCSI Sequential Device
Lecteurs de CD-ROM/DVD-ROM
Lecteurs de disque
COMPAQ MAB3045SC SCSI Disk Device
QLOGIC PSEUDO LUN SCSI Disk Device
QLOGIC PSEUDO LUN SCSI Disk Device
QLOGIC PSEUDO LUN SCSI Disk Device
STK 9200 SCSI Disk Device
STK 9200 SCSI Disk Device
STK 9200 SCSI Disk Device
STK 9200 SCSI Disk Device
STK 9200 SCSI Disk Device
Lecteurs de disquettes
Moniteurs
Ordinateur
Périphériques système
Ports (COM et LPT)
Souris et autres périphériques de pointage

Figure 4. Device manager interface.

All the SVA LUNs available on the domain the server is connected to are displayed if everything is well configured.

4.1.2 Define one functional SCSI LUN device with SVAA

> Declaration of SCSI LUN

```
SIB> defdev -subsys NOVA -fdid 006 -scsiaddr 2.0.5 -name ECAM -fcapa 1g -devtyp SCSI
SIB9830I: Device (FDID 006) successfully defined with an exact size of 2.5 GB.
```

You must specify the first LUN as an ECAM device to be able to connect to the subsystem from the server used.

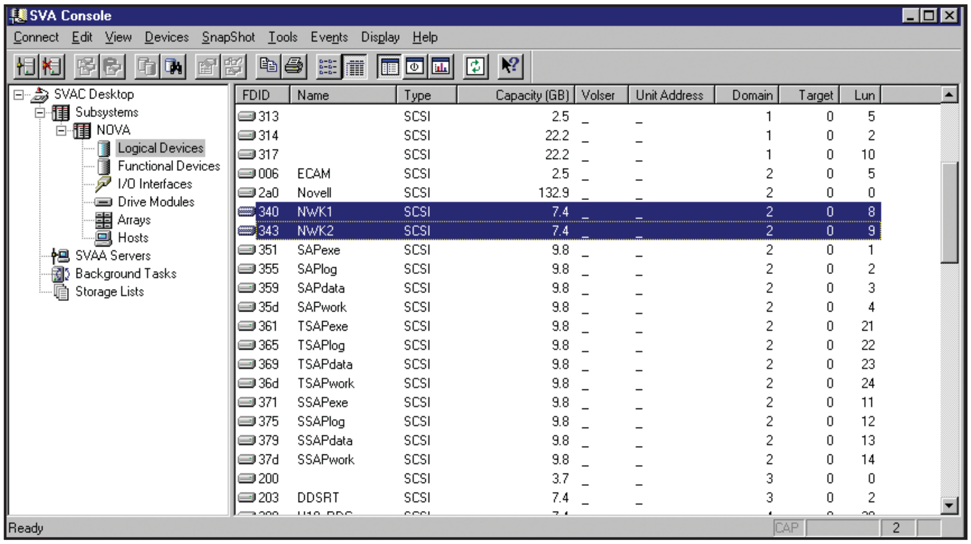
> Server and subsystem information.

```
# sibadmin
Starting CLI...
SIB> dserver
SVAA Server: Baize          Date/Time: 10-10-2002 17:21:27 CEST

Version: 3.1
Description:
Trace Status: Disabled
Installed Features:
   SnapShot
Maintenance Level:
   0 PPFinfo PTF=L2P008T Patch=12 FIX=0 Issue=768241
Host:
   Name: BAIZE
   TCP/IP Port: 41248
Host OS Level:
     5
     0
SIB> qsubsys
SVAA Server: Baize           Date/Time: 10-10-2002 17:21:31 CEST

NOVA
```

If there is no subsystem attached to the server, use the addsubsyspath command. For more information on the sibadmin commands, type help or help command.

4.1.3 Define two functional SCSI LUN devices with SVAC

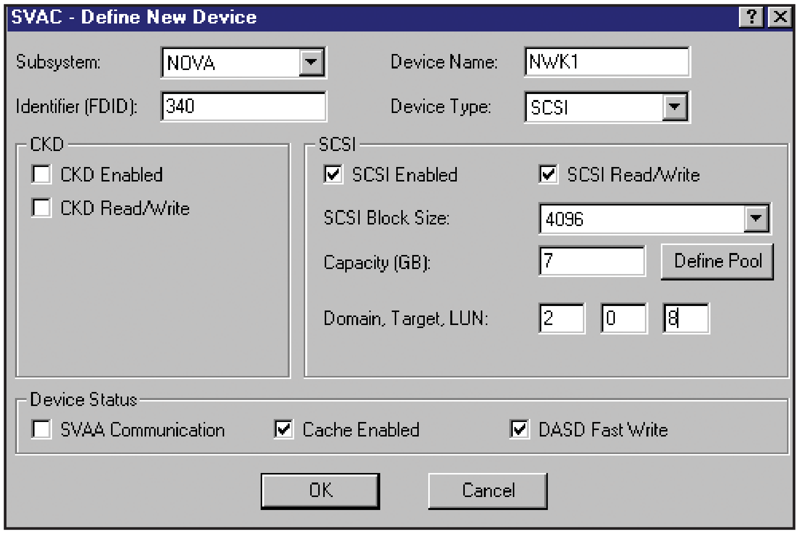
Use the SVA Console to view and create LUNs. Connect to the server that has a connection with the SVA and that has access to an ECAM device. This device enables the connection to a configured subsystem on the SVA.



FDID	Name	Type	Capacity (GB)	Volser	Unit Address	Domain	Target	Lun
313		SCSI	2.5	_	_	1	0	5
314		SCSI	22.2	_	_	1	0	2
317		SCSI	22.2	_	_	1	0	10
006	ECAM	SCSI	2.5	_	_	2	0	5
2a0	Novell	SCSI	132.9	_	_	2	0	0
340	NWK1	SCSI	7.4	_	_	2	0	8
343	NWK2	SCSI	7.4	_	_	2	0	9
351	SAPexe	SCSI	9.8	_	_	2	0	1
355	SAPlog	SCSI	9.8	_	_	2	0	2
359	SAPdata	SCSI	9.8	_	_	2	0	3
35d	SAPwork	SCSI	9.8	_	_	2	0	4
361	TSAPexe	SCSI	9.8	_	_	2	0	21
365	TSAPlog	SCSI	9.8	_	_	2	0	22
369	TSAPdata	SCSI	9.8	_	_	2	0	23
36d	TSAPwork	SCSI	9.8	_	_	2	0	24
371	SSAPexe	SCSI	9.8	_	_	2	0	11
375	SSAPlog	SCSI	9.8	_	_	2	0	12
379	SSAPdata	SCSI	9.8	_	_	2	0	13
37d	SSAPwork	SCSI	9.8	_	_	2	0	14
200		SCSI	3.7	_	_	3	0	0
203	DDSRT	SCSI	7.4	_	_	3	0	2

Figure 5. SVA console interface (connection with ECLIPSE).

To create a device, go to the menu Devices > Define New…and complete with the right information.



Subsystem: NOVA

Device Name: NWK1

Identifier (FDID): 340

Device Type: SCSI

CKD

☐ CKD Enabled

☐ CKD Read/Write

SCSI

☑ SCSI Enabled

☑ SCSI Read/Write

SCSI Block Size: 4096

Capacity (GB): 7

Domain, Target, LUN: 2 0 8

Device Status

☐ SVAA Communication

☑ Cache Enabled

☑ DASD Fast Write

Figure 6. Definition of a new device interface.

Create two new LUNs (one for the source and one for the target):

> Go to menu Devices > Define New …

> Complete with the right information

> Specify the correct domain, target and LUN numbers

> To choose the Identifier of the LUN, use the Define Pool button to display all the available FDID

> Choose OK to instantly create the LUN.

4.1.4 LUNs configuration

> Configure the SVA path.

```
C:\STK_scripts\BackupServer\target>sppath -v
...
SPD=5  c3b1t 20d5 dev=0,  6 type=2 SANID=”STK 9500 0000000000180006”
SPD=25 c3b1t20d8 dev=0,  7 type=2 SANID=”STK 9500 0000000000180340”
SPD=26 c3b1t20d9 dev=0,  8 type=2 SANID=”STK 9500 0000000000180343”
```

> Generate kernel configuration files.

```
D:\>setsp -g
D:\>setsp -S
D:\>setsp
==============================================================================
   5    c3b1t20d5/6          Good      X        32     1       20      3000       1
 HardDisk 6   I:                                  ID = “STK 9500 0000000000180006”
==============================================================================
  25    c3b1t20d8/7         Excluded  X    X    32     1       20      3000       1
 HardDisk 7   P:                                  ID = “STK 9500 0000000000180340”
==============================================================================
  26    c3b1t20d9/8         Excluded  X    X    32     1       20      3000       1
 HardDisk 8   G:                                  ID = “STK 9500 0000000000180343”
==============================================================================
```

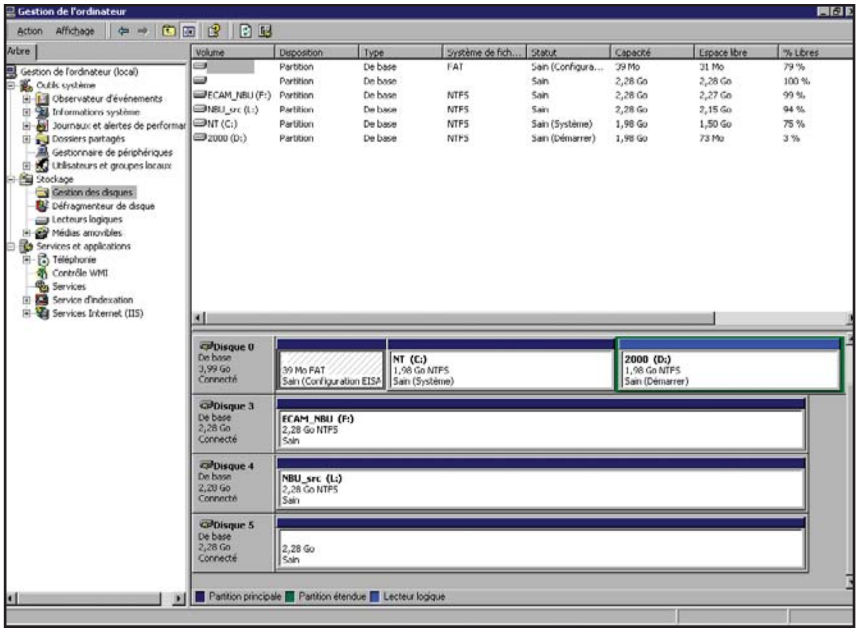
> Make LUNs available for the production server (depending on the configuration used — single server backup or with a dedicated backup server).

```
D:\>setsp –e0 –l5
D:\>setsp –e0 –l25
```

> Label and format the new devices

- Use the Manager Console to view the disks attached to the current server.
- Then, choose a basic disk rather than a dynamic one (dynamic disk does not support the snapshot operation except using specific scripts detailed in the white paper named *“Snapshot”* with Windows 2000 Dynamic Disk).
- Create a partition using the wizard tool of the console (choose a name to label it and preferably use the quick format option).
- Choose the drive letter on which the device will be mounted (L: for the source LUN and M: for the target one).

Note The target device is also mounted in order to get information on the physical disk name on which the system mounts the LUN. Use the command stkassign info M: to display the target disk name.



Volume	Disposition	Type	Système de fich...	Statut	Capacité	Espace libre	% Libres
	Partition	De base	FAT	Sain (Configura...	39 Mo	31 Mo	79 %
	Partition	De base		Sain	2,28 Go	2,28 Go	100 %
ECAM_NBU (F:)	Partition	De base	NTFS	Sain	2,28 Go	2,27 Go	99 %
NBU_src (L:)	Partition	De base	NTFS	Sain	2,28 Go	2,15 Go	94 %
NT (C:)	Partition	De base	NTFS	Sain (Système)	1,98 Go	1,50 Go	75 %
2000 (D:)	Partition	De base	NTFS	Sain (Démarrer)	1,98 Go	73 Mo	3 %

Disque 0
De base
3,99 Go
Connecté

39 Mo FAT
Sain (Configuration EISA

NT (C:)
1,98 Go NTFS
Sain (Système)

2000 (D:)
1,98 Go NTFS
Sain (Démarrer)

Disque 3
De base
2,28 Go
Connecté

ECAM_NBU (F:)
2,28 Go NTFS
Sain

Disque 4
De base
2,28 Go
Connecté

NBU_src (L:)
2,28 Go NTFS
Sain

Disque 5
De base
2,28 Go
Connecté

2,28 Go
Sain

Figure 7. Manager Console Interface — Disk Management.

4.2 NETWORKER CONFIGURATION

Launch the NetWorker Administrator Interface in order to use the backup utilities. Go to Start > Programs > NetWorker Group > NetWorker Administrator.

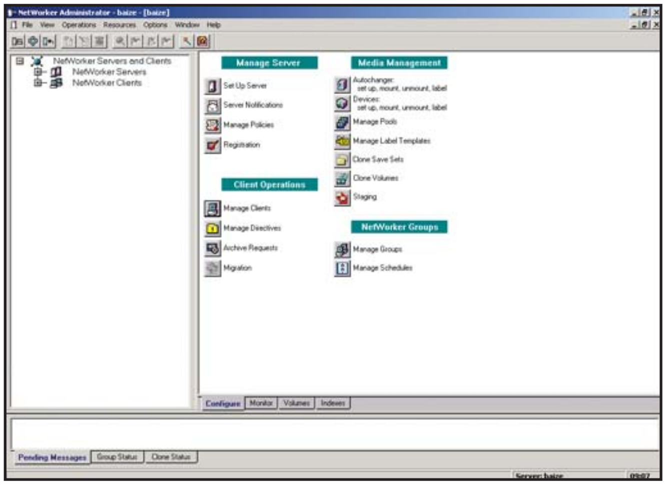


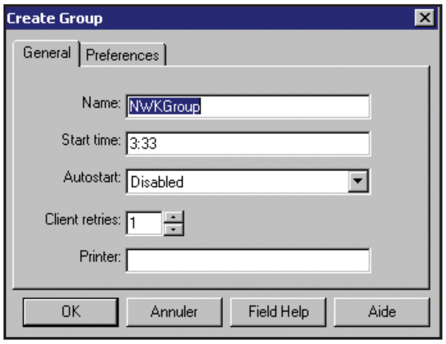
Figure 8. The NetWorker Administration Interface.

The NetWorker Server is installed on the backup server. In this configuration the Master Server is also a client. The NetWorker client application has to be installed on the production server.

Configuration steps

The NetWorker Backup tool must be configured manually. Perform the following:

> Create a new group. Use the NetWorker interface > NetWorker Groups > Manage Groups, go to the menu Operations > Create… and complete the fields with the appropriate information.



Create Group

Name: NWKGroup

Start time: 3:33

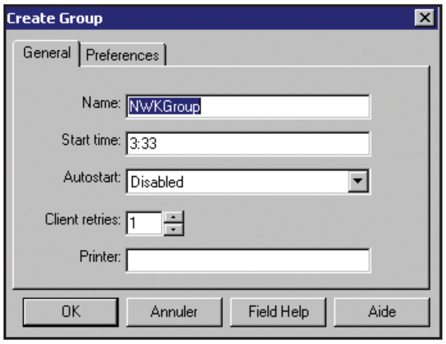
Autostart: Disabled

Client retries: 1

Printer:

Figure 9. NetWorker Group Creation Interface.

>Create a new label template for the pool to be created. Use the NetWorker interface > Media Management > Manage Label Templates, go to the menu Operations > Create… and complete the fields with the appropriate information.



Create Group

Name: NWKGroup

Start time: 3:33

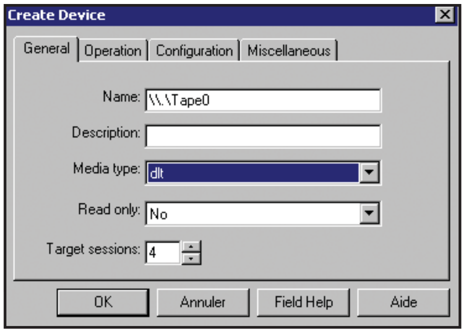
Autostart: Disabled

Client retries: 1

Printer:

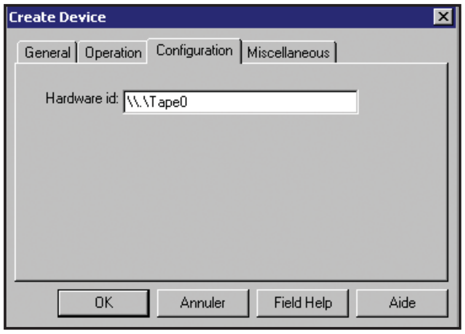
Figure 10. NetWorker Label Template Creation Interface.

The label template will be used to label the device for the pool that will be used to do the backup. The field "Next" will be automatically completed during creation.



Name: \\.\Tape0
Description:
Media type: dlt
Read only: No
Target sessions: 4

Figure 11. NetWorker Device Creation Interface.



Hardware id: \\.\Tape0

Figure 12. NetWorker Device Configuration.

> Create a new device for the pool to be created. Use the NetWorker interface > Media Management > Devices: set up, mount, umount, label, go to the menu Operations > Create… and complete the fields with the appropriate information (**Figure 11**).

Then, click on the Configuration tab and complete the Hardware id field (**Figure 12**).

Before configuring the new device, make sure that the tape drive is correctly configured in terms of hardware. Use the Management Console to see the list of hardware connected to the server.

```
C:\Program Files\nsr\bin>inquire
scsidev@0.0.0:ST340016A               3.05|Disk, \\.\PHYSICALDRIVE0
scsidev@1.0.0:ATAPI-CDROM-DRIVE-52MAX 52CA|CD-ROM
scsidev@6.20.0:STK      9500              1127|Disk

C:\Program Files\nsr\bin>jbverify

 Jbverify is running on host baize

 no enabled jukeboxes found in database.
 Processing stand-alone devices...

 Processing \\.\Tape0
  no block size found for this device: \\.\Tape0!
 Finished processing \\.\Tape0

 Finished processing stand-alone devices.

*******************************************************************
                  Summary report of jbverify
                  ======= ====== == ========
Hostname   Device Handle    Blocksize  Jukebox         Drv No. Status
____   _______    _____  ____          ____ ___
baize      \\.\Tape0         N/A        N/A              N/A      Pass
*******************************************************************
 Exiting jbverify successfully.
```

> Create the new pool in charge of the backup operation. Use the NetWorker interface > Media Management > Manage Pools, go to the menu Operations > Create… and complete the fields with the appropriate information (**Figure 13**).

Do not complete the Clients field yet. The client must be configured for the pool first. Use the NetWorker interface > Media Management > Manage Pools, go to the menu Operations > Create … and complete the fields with the appropriate information.

> Create the client that will have save sets to be backed up. In the Single Backup Server configuration, create the baize client. In the Production server-less backup configuration, create the neste client. Use the NetWorker interface > Media Management > Manage Pools, go to the menu Operations > Create … and complete the fields with the appropriate information (**Figure 14**).

Now that the client is configured for the pool NWKPool, modify the pool and complete the Clients field with the client name.

> Specify a device used by the Default group for the NetWorker indexes and bootstrap to be backed up (cannot be the same media as the one used to do the data backups, a disk space can be specified).

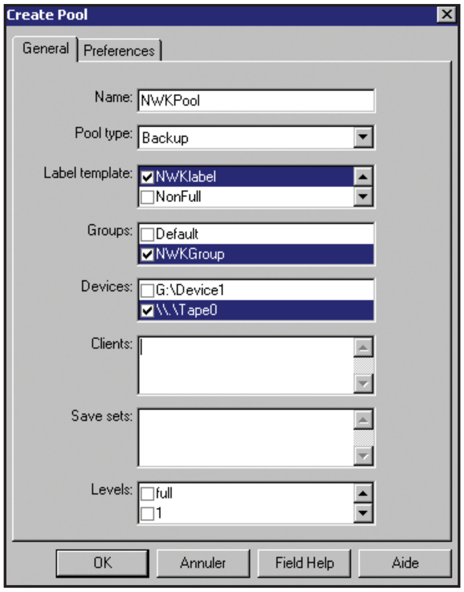
Note **For the single-server configuration, we use P:\ for the target device that will receive the snapped data. So the NetWorker group has to backup P:\ The target LUNs cannot be mounted on the same device letter. Use whatever free letter. For the dedicated backup server architecture, we use P:\ You can use whatever free device letter.**

Single server

Baize is considered as the production and the backup server.

Additional backup server

The NetWorker client must be installed on the production server. In this configuration with the dedicated backup server, Neste is the production server. The server Baize must be specified as the NetWorker server.



Name: NWKPool

Pool type: Backup

Label template: ☑ NWKlabel
□ NonFull

Groups: □ Default
☑ NWKGroup

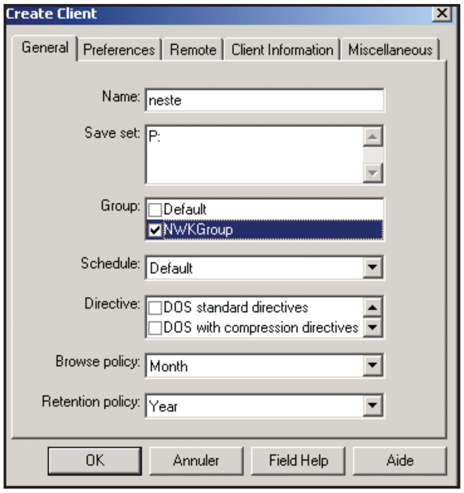
Devices: □ G:\Device1
☑ \\.\Tape0

Clients:

Save sets:

Levels: □ full
□ 1

Figure 13. NetWorker Pool Creation Interface.



Name: neste

Save set: P:

Group: □ Default
☑ NWKGroup

Schedule: Default

Directive: □ DOS standard directives
□ DOS with compression directives

Browse policy: Month

Retention policy: Year

Figure 14. NetWorker Client Creation Interface.

5 THE MAIN PROCESSES

The whole backup process is composed of three main processes:

> The release process: release the target LUNs for them to be cleaned for use
> The SnapShot process: snap the source to the target LUNs
> The backup process: backup from the SVA disks to the tapes.

The main script is in **Appendix A: Main Backup Script (No Additional Backup Server)** for the single production and backup server and in **Appendix B: Main Backup Script (Dedicated Backup Server)** for the configuration with the additional backup server.

The principle is the same while using a single production and backup server and using a dedicated backup server. The main difference with the second architecture is that all the operations are launched from the backup server.

All these processes are automated and use several scripts. Each process uses the lunconfig.lst file to get all the needed information about the source and target devices. Please refer to **Appendix C: LUNs Configuration File (No Additional Backup Server)** or to **Appendix D: LUNs Configuration Files (Dedicated Backup Server)**.

5.1 DEVELOPMENT OVERVIEW

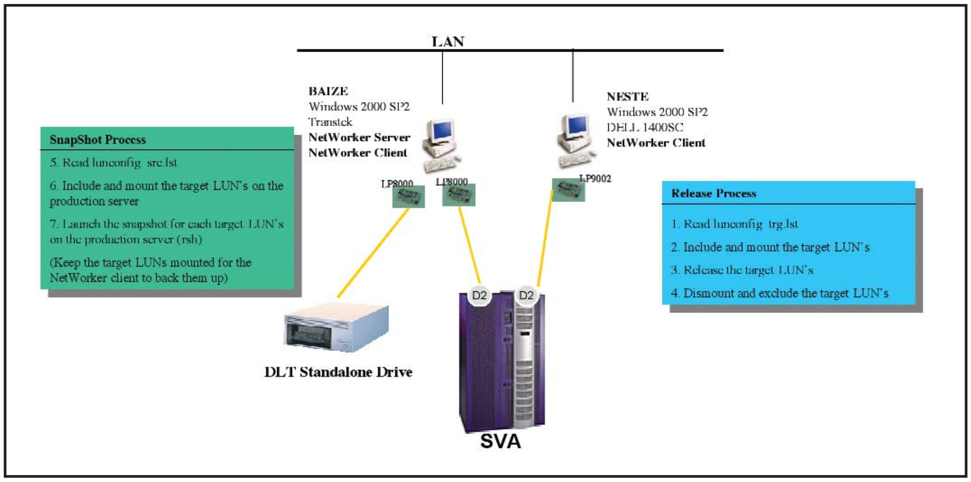

LAN
BAIZE
Windows 2000 SP2
Transtek
NetWorker Server
NetWorker Client
NESTE
Windows 2000 SP2
DELL 1400SC
NetWorker Client
SnapShot Process
5. Read lunconfig src.lst
6. Include and mount the target LUN's on the production server
7. Launch the snapshot for each target LUN's on the production server (rsh)
(Keep the target LUNs mounted for the NetWorker client to back them up)
LP8000
LP8000
LP9002
Release Process
1. Read lunconfig trg.lst
2. Include and mount the target LUN's
3. Release the target LUN's
4. Dismount and exclude the target LUN's
D2
D2
DLT Standalone Drive
SVA

Figure 15. Processes development overview (single-server).

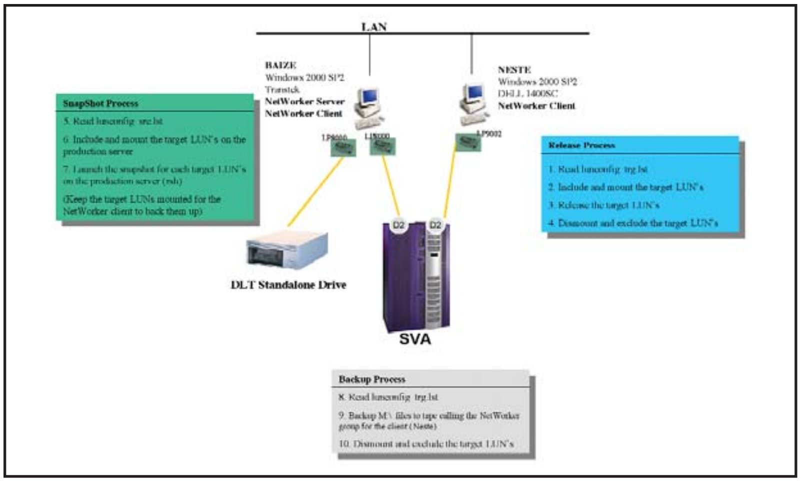

LAN
BAIZE
Windows 2000 SP2
Transtek
NetWorker Server
NetWorker Client
NESTE
Windows 2000 SP2
DELL 1400SC
NetWorker Client
SnapShot Process
5. Read lunconfig_src.lst
6. Include and mount the target LUN's on the production server
7. Launch the snapshot for each target LUN's on the production server (rsh)
(Keep the target LUNs mounted for the NetWorker client to back them up)
LP8000
LP8000
LP9002
Release Process
1. Read lunconfig_trg.lst
2. Include and mount the target LUN's
3. Release the target LUN's
4. Dismount and exclude the target LUN's
D2
D2
DLT Standalone Drive
SVA
Backup Process
8. Read lunconfig_trg.lst
9. Backup M:\ files to tape calling the NetWorker group for the client (Neste)
10. Dismount and exclude the target LUN's

Figure 16. Processes development overview (dedicated backup server).

5.2 BACKUP EXECUTION OVERVIEW

For all the mentioned scripts, please refer to **Appendix C, Appendix D, Appendix E, Appendix F and Appendix G.**

5.2.1 Single server

For the single production/backup server, after all the configuration steps are completed, perform the following:

> Choose a directory where all the custom scripts will be located and initialize the STK_SCRIPTS_HOME environment variable.

> Copy all the scripts for this configuration in the directory you have chosen. In these scripts the environment variable STK_SCRIPTS_HOME is initialized to D:\STK_scripts.

- ***lunconfig.lst:*** file containing the LUNs configuration for SVA Path software and Windows 2000 disk partitions.
- ***backup_script.cmd:*** main script that launches all the processes
- ***release_luns.cmd and release.cmd:*** includes, mounts, releases, dismounts and excludes the target LUNs
- ***snap_luns.cmd and snap.cmd:*** includes, mounts, snaps, dismounts and excludes the target LUNs
- ***backup_luns.cmd:*** includes, mounts, backs up dismounts and excludes the target LUNs
- ***mount_luns.cmd and mount.cmd:*** includes and mounts the target LUNs specified in the lunconfig file
- ***umount_luns and umount.cmd:*** dismounts and excludes the target LUNs listed in the lunconfig file.

> Change these scripts each time an absolute path is given to reflect your own organization.

> Change the lunconfig file to reflect your configuration.

> To perform a manual backup, launch the backup_script.cmd script. Please refer to **Appendix H: Example of Successful Backup (No Additional Backup Server)** for a detailed log of a successful backup.

Note Do not forget to put the *stkassign* executable in the directory where all the scripts are copied. This

command is used to mount, dismount, synchronize or get information on the disks.

5.2.2 Dedicated backup server

For the architecture with the additional backup server, after all the configuration steps completed, perform the following:

> Choose a directory where all the custom scripts will be located and initialize the STK_SCRIPTS_HOME environment variable either on the data host or on the backup host.

> Copy all the scripts for this configuration in the directory you have chosen. In these scripts, the environment variable STK_SCRIPTS_HOME is initialized to D:\STK_scripts.

> On the production server:

- ***lunconfig_src.lst:*** file containing the LUNs configuration for SVA Path software and Windows 2000 disk partitions to perform the snapshot locally
- ***snap_luns.cmd and snap.cmd:*** includes, mounts, snaps, dismounts and excludes the target LUNs
- ***umount_luns and umount.cmd:*** dismounts and excludes the target LUNs listed in the lunconfig file

> On the backup server:

- ***lunconfig_trg.lst:*** file containing the LUNs configuration for SVA Path software and Windows 2000 disk partitions to perform the snapshot locally
- ***backup_script.cmd:*** main script that launches all the processes
- ***release_luns.cmd and release.cmd:*** includes, mounts, releases, dismounts and excludes the target LUNs
- ***backup_luns.cmd:*** includes, mounts, backs up dismounts and excludes the target LUNs
- ***mount_luns.cmd and mount.cmd:*** includes and mounts the target LUNs specified in the lunconfig file

> Change these scripts each time an absolute path is given to reflect your own organization.

> Change the lunconfig file to reflect your configuration.

> To perform a manual backup, launch the backup_script.cmd script. Please refer to **Appendix I: Example of Successful Backup (Dedicated Backup Server)** for the detailed logs of a successful backup.

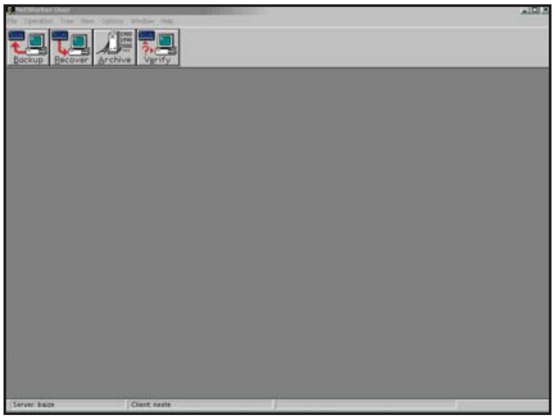
Note **Do not forget to put the stkassign executable in the directory where the scripts are duplicated both on the data and the backup hosts. This command is used to mount, dismount, synchronize or get information on the disks.**

5.3 HOW TO RESTORE

The procedure to restore files or directories is the same using either the single-server configuration or the dedicated backup server architecture.

To restore a file or directory, perform the following:

> Launch the NetWorker User Interface in order to use the backup utilities. Go to Start > Programs > NetWorker Group > NetWorker User.

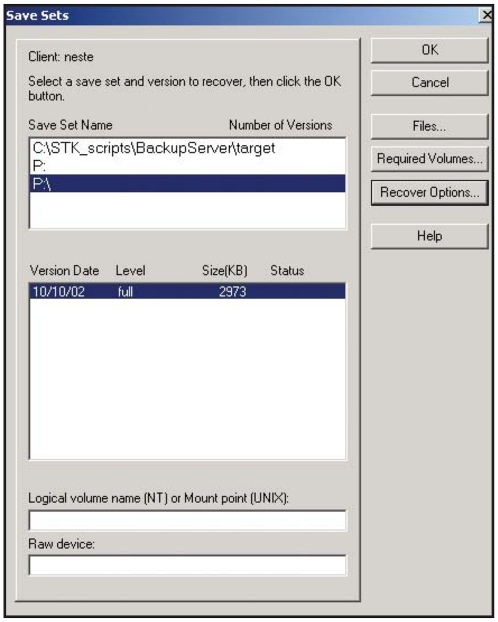


Server: baize

Client: neste

Figure 17. The NetWorker User Interface.

> Go to menu Operation > Save Sets Recovery… Choose the client name of the host that performed the backup (here it is Neste) and the client where the restore will be done (on Neste).



Client: neste

Select a save set and version to recover, then click the OK button.

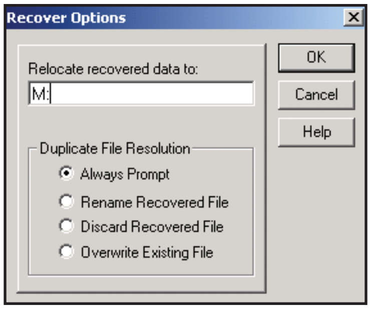
Save Set Name	Number of Versions
C:\STK_scripts\BackupServer\target	
P:	
P:\	

Version Date	Level	Size(KB)	Status
10/10/02	full	2973	

Logical volume name (NT) or Mount point (UNIX):

Raw device:

Figure 18. The NetWorker User Interface.



Relocate recovered data to: M:

Duplicate File Resolution

- ● Always Prompt
- ○ Rename Recovered File
- ○ Discard Recovered File
- ○ Overwrite Existing File

Figure 19. The Backup, Archive and Restore NetWorker Client interface.

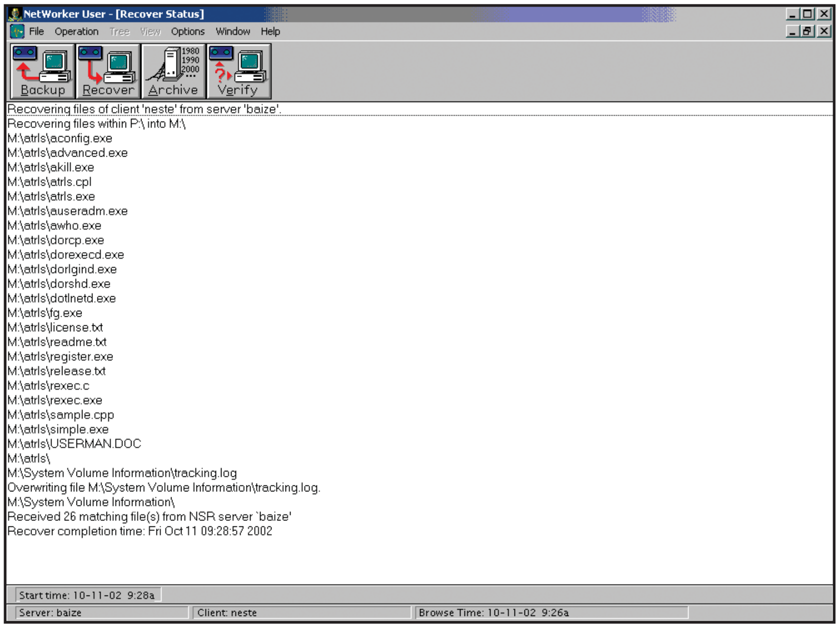
> Then, choose the Save Set and version of backup to be restored. Click on the Restore Options… button (**Figure 19**).

> Complete the restore options with the destination paths of the restore

- For the single server configuration: As the backup is performed using the snapshot target LUNs, the destination path of the restore is not the original location. So select the right path. Here, the M:\ directory was backed up using its snapshot on P:\, so the source for the restore is P:\ and the destination M:\.
- For the dedicated backup server configuration: As the backup is performed on the backup server, the target partition is mounted on any available drive letter. The restore destination path must be on the production server, which must be mounted to be able to perform the restore, M:\.

> Then, initiate the restore by clicking on OK and the client will browse the server to get backup information back. The restore operation is launched.

> The following window is displayed to view the restore status.



```
Recovering files of client 'neste' from server 'baize'.
Recovering files within P:\ into M:\
M:\atrls\aconfig.exe
M:\atrls\advanced.exe
M:\atrls\akill.exe
M:\atrls\atrls.cpl
M:\atrls\atrls.exe
M:\atrls\auseradm.exe
M:\atrls\awho.exe
M:\atrls\dorcp.exe
M:\atrls\dorexecd.exe
M:\atrls\dorlgind.exe
M:\atrls\dorshd.exe
M:\atrls\dotlnetd.exe
M:\atrls\fg.exe
M:\atrls\license.txt
M:\atrls\readme.txt
M:\atrls\register.exe
M:\atrls\release.txt
M:\atrls\rexec.c
M:\atrls\rexec.exe
M:\atrls\sample.cpp
M:\atrls\simple.exe
M:\atrls\USERMAN.DOC
M:\atrls\
M:\System Volume Information\tracking.log
Overwriting file M:\System Volume Information\tracking.log.
M:\System Volume Information\
Received 26 matching file(s) from NSR server `baize'
Recover completion time: Fri Oct 11 09:28:57 2002
```

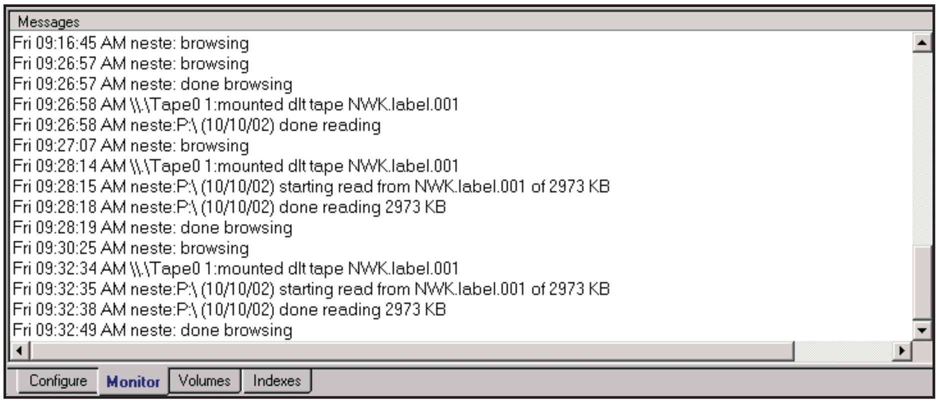
Start time: 10-11-02 9:28a

Server: baize | Client: neste | Browse Time: 10-11-02 9:26a

Figure 20. Restore Status Interface.

> The operation can be successful or not. If so, refer to the messages displayed to see what is the problem.

On the backup server, the monitor displays messages during the restore operation as it performs the restore for the NetWorker client.



Messages

```
Fri 09:16:45 AM neste: browsing
Fri 09:26:57 AM neste: browsing
Fri 09:26:57 AM neste: done browsing
Fri 09:26:58 AM \\.\Tape0 1:mounted dlt tape NWK.label.001
Fri 09:26:58 AM neste:P:\ (10/10/02) done reading
Fri 09:27:07 AM neste: browsing
Fri 09:28:14 AM \\.\Tape0 1:mounted dlt tape NWK.label.001
Fri 09:28:15 AM neste:P:\ (10/10/02) starting read from NWK.label.001 of 2973 KB
Fri 09:28:18 AM neste:P:\ (10/10/02) done reading 2973 KB
Fri 09:28:19 AM neste: done browsing
Fri 09:30:25 AM neste: browsing
Fri 09:32:34 AM \\.\Tape0 1:mounted dlt tape NWK.label.001
Fri 09:32:35 AM neste:P:\ (10/10/02) starting read from NWK.label.001 of 2973 KB
Fri 09:32:38 AM neste:P:\ (10/10/02) done reading 2973 KB
Fri 09:32:49 AM neste: done browsing
```

Figure 21. NetWorker Server Monitor during the Restore Operation.

Note The goal of the present document is to present a backup/restore solution based on a particular configuration. This is not the only solution. To perform the backup operation, there is no need of the NetWorker client Neste. The SnapShot operation on the production server is performed remotely from the backup server. It is possible to mount, snap and dismount the target LUNs on the production server and then, mount them on the backup server to back up data on the tape. The backup operation is considered as performed by the NetWorker server. In order to restore data, a script must be used to mount the source LUNs on the backup server, restore files and dismount the LUNs. In our case, the backup operation is considered as performed by the production server although it is the NetWorker server that realizes the operation. And the restore operation is launched using the NetWorker utilities.

6 CRITICAL ANALYSIS

Advantages	Drawbacks
Simple: there is only one SVA command per file system.	The snapshot command is launched by the backup server but executed by the client.
The data can be backed up or backed up to tape using another server that will mount the SVA snapshot.	
All the other advantages of a cold backup remain.	

SnapShot software works by creating different "views" of data rather than copying the data itself. By reducing the need to physically move data for duplication, SnapShot software creates copies, almost instantly. With SnapShot's ultra-fast duplication capability, backup operations are not on the critical path any more and could be run in parallel with other processes.

6.1 HOW TO SNAPSHOT DATA ON THE BACKUP SERVER?

In this section, we'll describe how to backup the "Backup server" without using SVAPATH. But in this architecture, the backup may be performed either by:

> The production server: You need to install the media server and connect a storage pool to this server. This method consumes the CPU of the production server.

> The network: This configuration is a standard configuration but the most drawback of the backup. In this case you consume CPU and network.

The only solution is SnapShot. To execute the snapshot we need to stop or suspend the application during approximately 2–5 seconds, no matter how much data is copied. You save a lot of time.

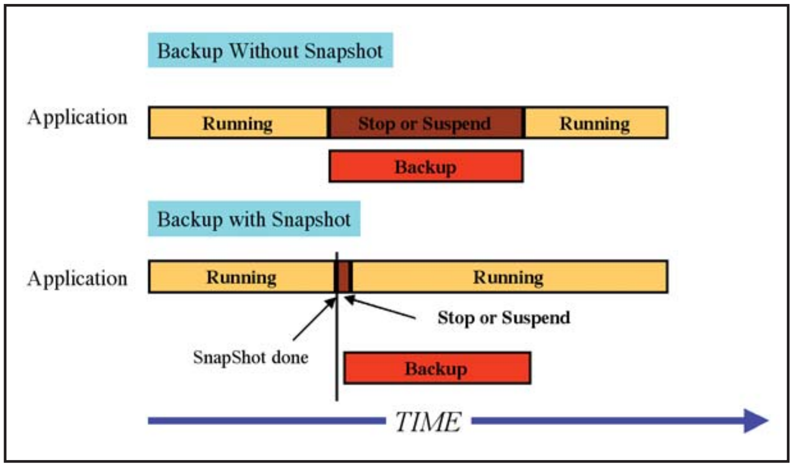


Figure 22. The SnapShot Real Advantage.

7 APPENDICES

This section contains a copy of each script used in this project.

APPENDIX A: MAIN BACKUP SCRIPT (NO ADDITIONAL BACKUP SERVER)

backup_script.cmd

```
setlocal enableextensions
@echo off

@REM ********************************************************************************
@REM This script is used to perform an automated backup using the SVA snapshot fea-
ture
@REM and calling the Legato NetWorker utility to backup data on a tape (DLT drive)
@REM
@REM All the custom scripts used are located in directory
C:\STK_scripts\SingleServer\
@REM For further information on the detailed steps, please refer to file
@REM C:\STK_scripts\SingleServer\master.log
@REM ********************************************************************************

@REM
@REM Releasing target LUNs
@REM =====================

@echo
@echo RELEASING TARGET LUNS
@call C:\STK_scripts\SingleServer\release_luns.cmd C:\STK_scripts\SingleServer\luncon-
fig.lst

@REM
@REM Snapping LUNs from source to target
@REM ==================================

@echo
@echo SNAPPING LUNS FROM SOURCE TO TARGET
@call C:\STK_scripts\SingleServer\snap_luns.cmd C:\STK_scripts\SingleServer\luncon-
fig.lst

@REM
@REM Calling the backup policy
@REM =========================

@echo
@echo CALLING BACKUP UTILITY
@call C:\STK_scripts\SingleServer\backup_luns.cmd C:\STK_scripts\SingleServer\luncon-
fig.lst

ENDLOCAL

pause
```

APPENDIX B: MAIN BACKUP SCRIPT (DEDICATED BACKUP SERVER)

```
@echo off
setlocal enableextensions


@REM ********************************************************************************
@REM This script is used to perform an automated backup using the SVA snapshot fea-
ture
@REM and calling the Legato NetWorker utility to backup data on a tape (DLT drive)
@REM
@REM All the custom scripts used are located in directory D:\STK_scripts
@REM For further information on the detailed steps, please refer to file
@REM D:\STK_scripts\BackupServer\target\master.log
@REM ********************************************************************************

@REM
@REM Releasing target LUNs
@REM =====================

@echo
@echo Releasing target LUNs....................
@call C:\STK_scripts\BackupServer\target\release_luns.cmd
C:\STK_scripts\BackupServer\target\lunconfig_trg.lst

@REM
@REM Snapping LUNs from source to target
@REM ===================================

@echo
@echo Snapping LUNs from source to target..................
rsh NESTE D:\STK_scripts\BackupServer\source\snap_luns.cmd
D:\STK_scripts\BackupServer\source\lunconfig_src.lst

@REM
@REM Calling the backup policy
@REM =========================

@echo
@echo Calling the backup utility.................
@call C:\STK_scripts\BackupServer\target\backup_luns.cmd
C:\STK_scripts\BackupServer\target\lunconfig_trg.lst

ENDLOCAL

pause
```

APPENDIX C: LUNS CONFIGURATION FILE (NO ADDITIONAL BACKUP SERVER)

lunconfig.lst

```
#
# This is the configuration file for the LUNs to Snap on the source server
#
# Fields are separated by tabs!
#
# Field Description:
#
# 1. The first field is the UNIQUE identifier of each line of this file
#    It is not used by the script at that time. Use SVALUN_<LUN Number>.
# 2. The second field is LUN number from SVA Path (use setsp to display the LUNs)
# 3. The third field is the snapshot source partition letter
# 4. The fourth field is the snapshot TARGET partition letter (set this parameter
# carefully!!!)
# 5. The fifth field is the physical disk name for the target partition
#    Use “stkassign list” to display the currently mapped partitions
#
# examples are given below:
#
#              Target
# Target       SVAPath Src       Target   Target_Device
# SVA LUN      LUN#
SVALUN_9       26       M:       P:       \Device\HarddiskVolume9
```

APPENDIX D: LUNS CONFIGURATION FILES (DEDICATED BACKUP SERVER)

lunconfig_src.lst

```
#
# This is the configuration file for the LUNs to Snap on the source server
#
# Fields are separated by tabs!
#
# Field Description:
#
# 1. The first field is the UNIQUE identifier of each line of this file
#    It is not used by the script at that time. Use SVALUN_<LUN Number>.
# 2. The second field is LUN number from SVA Path (use setsp to display the LUNs)
# 3. The third field is the snapshot source partition letter
# 4. The fourth field is the snapshot TARGET partition letter (set this parameter
carefully!!!)
# 5. The fifth field is the physical disk name for the target partition
#    Use “stkassign list” to display the currently mapped partitions
#
# examples are given below:
#
#              Target
# Target       SVAPath Src       Target   Target_Device
# SVA LUN      LUN#
SVALUN_9       15      M:        P:       \Device\HarddiskVolume12
```

lunconfig_trg.lst

```
#
# This is the configuration file for the LUNs to Backup on the target server
#
# Fields are separated by tabs!
#
# Field Description:
#
# 1. The first field is the UNIQUE identifier of each line of this file
#    It is not used by the script at that time. Use SVALUN_<LUN Number>.
# 2. The second field is LUN number from SVA Path (use setsp to display the LUNs)
# 3. The third field is the snapshot source partition letter
# 4. The fourth field is the snapshot TARGET partition letter (set this parameter
carefully!!!)
# 5. The fifth field is the physical disk name for the target partition
#    Use “stkassign list” to display the currently mapped partitions
#
# examples are given below:
#
#              Target
# Target       SVAPath Src       Target   Target_Device
# SVA LUN      LUN#
SVALUN_9       26      M:        P:       \Device\HarddiskVolume9
```

APPENDIX E: RELEASE LUNS SCRIPTS

release_luns.cmd

```
@echo off
SETLOCAL ENABLEEXTENSIONS

REM
REM This script is used to perform a release on the target luns on the backup server
REM


REM
REM Update next line to reflect the absolute path of SVAA CLI Batch File
REM On your configuration
REM Use double quotes if spaces are part of the pathnames
REM ==================================================================

set sibadmin=”C:\Program Files\StorageTek\Shared Virtual Array Administrator
3.1.0\bin\sibadmin.bat”

REM
REM Check Usage
REM we need STK_SCRIPTS_HOME to be set properly
REM We need the LUN Configuration File as an input parameter
REM ===========

if [%1] EQU [] goto s_usage
if [%2] NEQ [] goto s_usage

if [%STK_SCRIPTS_HOME%] EQU [] goto s_usage
set local_run_path=%STK_SCRIPTS_HOME%

if NOT EXIST %local_run_path%\release.cmd goto s_usage
goto s_start

:s_usage

echo syntax error
echo Set STK_SCRIPTS_HOME environment variable properly.
echo usage: %0 LUN_configuration_file_name
goto s_end

:s_start

set lun_config_file=%1

REM
REM Setting default file names
REM =========================

set master_log=master.log

REM
REM Get Date and Time for Logging
REM ============================

for /F “tokens=1*” %%P in (‘date /T’) do (set d_day=%%P %%Q)
for /F “tokens=1*” %%P in (‘time /T’) do (set d_date=%d_day% %%P)
```

```
echo . >> %local_run_path%\%master_log%
echo . >> %local_run_path%\%master_log%
echo ========================================================== >>
%local_run_path%\%master_log%

echo ****** %0 starting on [%d_date%] >> %local_run_path%\%master_log%

echo. >> %local_run_path%\%master_log%

set lun_config_file=%1

REM
REM For Each LUN:
REM - Mount, Snap, Check and Unmount
REM ===============================

REM [ delims is a tab in the following loop ]

FOR /F “tokens=1,2,3,4,5* delims= “ %%G in (‘findstr /V ^# %lun_config_file% 2^>nul’)
do (call %local_run_path%\release.cmd %%H %%J %%K)

echo. >> %local_run_path%\%master_log%

echo ************************************************ >>
%local_run_path%\%master_log%

echo. >> %local_run_path%\%master_log%

:a_end

for /F “tokens=1*” %%P in (‘date /T’) do (set d_day=%%P %%Q)
for /F “tokens=1*” %%P in (‘time /T’) do (set d_date=%d_day% %%P)

echo. >> %local_run_path%\%master_log%
echo ****** %0 ending on  [%d_date%] >> %local_run_path%\%master_log%
ENDLOCAL

:s_end
```

Note This script is used for the single-server configuration and for the production / backup configuration.

release.cmd

```
@echo off
setlocal enableextensions

REM
REM Check Usage - we need 4 parameters
REM - 1 - SVA Path Target LUN Number
REM - 2 - Target Partition Letter
REM - 3 - Target Physical Disk Name
REM =================================

if [%3] EQU [] goto s_usage
if [%4] NEQ [] goto s_usage
goto s_start

:s_usage

echo syntax error
echo usage: ^<%0 SVAPath_lun_number target_drive_letter target_physical_disk_name^>
goto s_end

:s_start
set svapath_target_LUN=%1
set target_letter=%2
set target_disk=%3

REM
REM Processing LUN
REM ==============

echo. >> %local_run_path%\%master_log%

echo Performing the release..................................
echo ****** Processing LUN: >> %local_run_path%\%master_log%
echo  SVAPath Target LUN:         %svapath_target_LUN% >>
%local_run_path%\%master_log%
echo  Target Paritition Letter:   %target_letter% >> %local_run_path%\%master_log%
echo  Target Disk Name:           %target_disk% >> %local_run_path%\%master_log%

REM
REM Include and mount Target LUNs
REM =============================

echo - Including Target LUN... >> %local_run_path%\%master_log% 2>&1
setsp -l %svapath_target_LUN% -e 0 >> %local_run_path%\%master_log% 2>&1
if %ERRORLEVEL% NEQ 0 goto s_error

echo - Mounting Target Partition... >> %local_run_path%\%master_log% 2>&1
%local_run_path%\stkassign mount %target_letter% %target_disk% >>
%local_run_path%\%master_log% 2>&1
if %ERRORLEVEL% NEQ 0 goto s_error

REM
REM Format Target LUNs - avoid catching side effects on NTFS
REM =======================================================

label %target_letter% NWKtrg >> %local_run_path%\%master_log% 2>&1
echo - Formatting LUNs...  >> %local_run_path%\%master_log% 2>&1
@(
echo NWKtrg
echo Y
) | %SystemRoot%\system32\format %target_letter% /FS:NTFS /Q /V:NWKtrg >>
%local_run_path%\%master_log% 2>&1
if %ERRORLEVEL% NEQ 0 goto s_error

REM
REM Unmount and exclude Target LUNs (Target Partition Letter becomes available to NT)
REM ================================================================================

echo - Syncing and Unmounting Target Partition... >> %local_run_path%\%master_log%
2>&1
%local_run_path%\stkassign sync %target_letter% >> %local_run_path%\%master_log% 2>&1
%local_run_path%\stkassign umount %target_letter% >> %local_run_path%\%master_log%
2>&1
if %ERRORLEVEL% NEQ 0 goto s_error

REM echo - Excluding Target LUN... >> %local_run_path%\%master_log% 2>&1
setsp -l %svapath_target_LUN% -e 1 >> %local_run_path%\%master_log% 2>&1
if %ERRORLEVEL% NEQ 0 goto s_error
```

```
goto s_end

:s_error

echo
echo ==== ERROR encountered!!! See messages above. ****** >> %local_run_path%\%mas-
ter_log%

:s_end

echo ****** Target LUNs Released. ****** >> %local_run_path%\%master_log%

ENDLOCAL
```

APPENDIX F: SNAPSHOT LUNS SCRIPTS

snap_luns.cmd

```
@echo off
SETLOCAL ENABLEEXTENSIONS

REM
REM Check Usage
REM we need STK_SCRIPTS_HOME to be set properly
REM We need the LUN Configuration File as an input parameter
REM ===========

if [%1] EQU [] goto s_usage
if [%2] NEQ [] goto s_usage
if [%STK_SCRIPTS_HOME%] EQU [] goto s_usage

set local_run_path=%STK_SCRIPTS_HOME%

if NOT EXIST %local_run_path%\snap.cmd goto s_usage
goto s_start

:s_usage

echo syntax error
echo Set STK_SCRIPTS_HOME environment variable properly.
echo usage: %0 _LUN_configuration_file_name >> %local_run_path%\%master_log%
goto s_end

:s_start
set lunconfig_file=%1
cd %local_run_path%

REM
REM Setting default file names
REM ==========================
set master_log=master.log

REM
REM Get Date and Time for Logging
REM =============================

for /F “tokens=1*” %%P in (‘date /T’) do (set d_day=%%P %%Q)
for /F “tokens=1*” %%P in (‘time /T’) do (set d_date=%d_day% %%P)

echo . >> %local_run_path%\%master_log%
echo . >> %local_run_path%\%master_log%
echo ========================================================== >>
%local_run_path%\%master_log%

echo ****** %0 starting on [%d_date%] >> %local_run_path%\%master_log%

echo. >> %local_run_path%\%master_log%


REM
REM For Each LUN:
REM - Mount, Snap, Check and Unmount
REM ================================

REM [ delims is a tab in the following loop ]
```

```
FOR /F “tokens=1,2,3,4,5* delims= “ %%G in (‘findstr /V ^# %lunconfig_file% 2^>nul’)
do (call %local_run_path%\snap.cmd %%H %%I %%J %%K)

echo. >> %local_run_path%\%master_log%

echo ************************************************ >>
%local_run_path%\%master_log%

echo. >> %local_run_path%\%master_log%

:a_end

for /F “tokens=1*” %%P in (‘date /T’) do (set d_day=%%P %%Q)
for /F “tokens=1*” %%P in (‘time /T’) do (set d_date=%d_day% %%P)

echo. >> %local_run_path%\%master_log%
echo ****** %0 ending on  [%d_date%] >> %local_run_path%\%master_log%
ENDLOCAL

:s_end
```

snap.cmd

```
@echo off
setlocal enableextensions

REM
REM Update next line to reflect the absolute path of SVAA CLI Batch File
REM Use double quotes if spaces are part of the pathnames
REM ===================================================================
set sibadmin=”D:\Program Files\StorageTek\Shared Virtual Array Administrator
3.1.0\bin\sibadmin.bat”

REM
REM Check Usage - we need 4 parameters
REM - 1 - SVA Path Target LUN Number
REM - 2 - Source Partition Letter
REM - 3 - Target Partition Letter
REM - 4 - Target Physical Disk Name
REM ==================================

if [%4] EQU [] goto s_usage
if [%5] NEQ [] goto s_usage
if [%STK_SCRIPTS_HOME%] EQU [] goto s_usage
goto s_start

:s_usage

echo syntax error
echo Set STK_SCRIPTS_HOME environment variable properly.
echo usage: ^<%0 SVAPath_lun_number source_drive_letter target_drive_letter
target_physical_disk_name^>
goto s_end

:s_start

set local_run_path=%STK_SCRIPTS_HOME%
set svapath_target_LUN=%1
set source_letter=%2
set target_letter=%3
set target_disk=%4

REM
REM Processing LUN
REM ==============
echo. >> %local_run_path%\%master_log%
echo Performing the snapshot............................
echo ****** Processing LUN: >> %local_run_path%\%master_log%
echo  SVAPath Target LUN:         %svapath_target_LUN% >>
%local_run_path%\%master_log%
echo  Source Partition Letter:    %source_letter% >> %local_run_path%\%master_log%
echo  Target Paritition Letter:   %target_letter% >> %local_run_path%\%master_log%
echo  Target Disk Name:           %target_disk% >> %local_run_path%\%master_log%

REM
REM Include and mount Target LUNs
REM =============================
echo - Including Target LUN... >> %local_run_path%\%master_log% 2>&1
setsp -l %svapath_target_LUN% -e 0 >> %local_run_path%\%master_log% 2>&1
if %ERRORLEVEL% NEQ 0 goto s_error

echo - Mounting Target Partition... >> %local_run_path%\%master_log% 2>&1
%local_run_path%\stkassign mount %target_letter% %target_disk% >>
%local_run_path%\%master_log% 2>&1
if %ERRORLEVEL% NEQ 0 goto s_error

REM
REM Snapping From Source to Target
REM ==============================
echo - Snapping LUNs... >> %local_run_path%\%master_log% 2>&1
call %sibadmin% snap -source %source_letter% -target %target_letter% >>
%local_run_path%\%master_log% 2>&1
if %ERRORLEVEL% NEQ 0 goto s_error


goto s_end

:s_error

echo ==== ERROR encountered!!! See messages above. ****** >> %local_run_path%\%mas-
ter_log%

:s_end

echo ****** Target LUNs Snapped. ****** >> %local_run_path%\%master_log%

ENDLOCAL
```

APPENDIX G: BACKUP LUNS SCRIPTS

backup_luns.cmd

```
@echo off
SETLOCAL ENABLEEXTENSIONS

REM
REM Update next line to reflect the absolute path of NetWorker bpbackup command
REM Use double quotes if spaces are part of the pathnames
REM ==================================================================
set nsrsave=”C:\Program Files\nsr\bin\savegrp.exe”

if [%STK_SCRIPTS_HOME%] EQU [] goto s_usage
set local_run_path=%STK_SCRIPTS_HOME%

goto s_start

:s_usage

echo syntax error
echo Set STK_SCRIPTS_HOME environment variable properly.
echo usage: %0 LUN_configuration_file_name
goto s_end

:s_start

set lunconfig_file=%1

REM
REM Setting default file names
REM =========================

set master_log=master.log

REM
REM Get Date and Time for Logging
REM ============================

for /F “tokens=1*” %%P in (‘date /T’) do (set d_day=%%P %%Q)
for /F “tokens=1*” %%P in (‘time /T’) do (set d_date=%d_day% %%P)

echo . >> %local_run_path%\%master_log%
echo . >> %local_run_path%\%master_log%
echo ========================================================== >>
%local_run_path%\%master_log%

echo ****** %0 NWK backup starting on [%d_date%] >> %local_run_path%\%master_log%

echo. >> %local_run_path%\%master_log%


call %nsrsave% -G NWKGroup
rsh NESTE D:\STK_scripts\BackupServer\source\umount_luns.cmd
D:\STK_scripts\BackupServer\source\lunconfig_src.lst


for /F “tokens=1*” %%P in (‘date /T’) do (set d_day=%%P %%Q)
for /F “tokens=1*” %%P in (‘time /T’) do (set d_date=%d_day% %%P)

echo. >> %local_run_path%\%master_log%
```

```
echo ****** %0 ending on  [%d_date%] >> %local_run_path%\%master_log%
ENDLOCAL

:s_end
```

mount_luns.cmd

```
@echo off
SETLOCAL ENABLEEXTENSIONS

REM
REM This script is used to perform a mount the target luns on the backup server
REM


REM
REM Update next line to reflect the absolute path of SVAA CLI Batch File
REM On your configuration
REM Use double quotes if spaces are part of the pathnames
REM ===================================================================

set sibadmin=”C:\Program Files\StorageTek\Shared Virtual Array Administrator
3.1.0\bin\sibadmin.bat”

REM
REM Check Usage
REM we need STK_SCRIPTS_HOME to be set properly
REM We need the LUN Configuration File as an input parameter
REM ===========

if [%1] EQU [] goto s_usage
if [%2] NEQ [] goto s_usage

if [%STK_SCRIPTS_HOME%] EQU [] goto s_usage
set local_run_path=%STK_SCRIPTS_HOME%

if NOT EXIST %local_run_path%\umount.cmd goto s_usage
goto s_start

:s_usage

echo syntax error
echo Set STK_SCRIPTS_HOME environment variable properly.
echo usage: %0 LUN_configuration_file_name
goto s_end

:s_start

set lun_config_file=%1

REM
REM Setting default file names
REM ==========================

set master_log=master.log
set lun_config_file=%1

REM
REM For Each LUN:
```

```
REM - Mount, Snap, Check and Unmount
REM ================================

REM [ delims is a tab in the following loop ]
FOR /F “tokens=1,2,3,4,5* delims= “ %%G in (‘findstr /V ^# %lun_config_file% 2^>nul’)
do (call %local_run_path%\mount.cmd %%H %%J %%K)

echo. >> %local_run_path%\%master_log%

echo ************************************************ >>
%local_run_path%\%master_log%

echo. >> %local_run_path%\%master_log%

:a_end

ENDLOCAL

:s_end
```

mount.cmd

```
@echo off
setlocal enableextensions

REM
REM Update next line to reflect teh absolute path of SVAA CLI Batch File
REM Use double quotes if spaces are part of the pathnames
REM ===================================================================
set sibadmin=”C:\Program Files\StorageTek\Shared Virtual Array Administrator
3.1.0\bin\sibadmin.bat”

REM
REM Check Usage - we need 4 parameters
REM - 1 - SVA Path Target LUN Number
REM - 2 - Target Partition Letter
REM - 3 - Target Physical Disk Name
REM ==================================

if [%3] EQU [] goto s_usage
if [%4] NEQ [] goto s_usage
goto s_start

:s_usage

echo syntax error
echo usage: ^<%0 SVAPath_lun_number target_drive_letter target_physical_disk_name^>
goto s_end

:s_start

set svapath_target_LUN=%1
set target_letter=%2
set target_disk=%3

REM
REM Include Target LUN (SVAPath)
REM ============================
echo - Including Target LUN... >> %local_run_path%\%master_log% 2>&1
setsp -l %svapath_target_LUN% -e 0 >> %local_run_path%\%master_log% 2>&1
if %ERRORLEVEL% NEQ 0 goto s_error

REM
REM Mount Target LUN (Target Partition Letter becomes available to NT)
REM =================================================================

echo - Mounting Target Partition... >> %local_run_path%\%master_log% 2>&1
%local_run_path%\stkassign mount %target_letter% %target_disk% >>
%local_run_path%\%master_log% 2>&1
if %ERRORLEVEL% NEQ 0 goto s_error


goto s_end

:s_error

echo ==== ERROR encountered!!! See messages above. ****** >> %local_run_path%\%mas-
ter_log%

:s_end

echo ****** Target LUNs mounted. ****** >> %local_run_path%\%master_log%

ENDLOCAL
```

umount_luns.cmd

```
@echo off
SETLOCAL ENABLEEXTENSIONS

REM
REM This script is used to perform an unmount the target luns on the backup server
REM

REM
REM Update next line to reflect the absolute path of SVAA CLI Batch File
REM On your configuration
REM Use double quotes if spaces are part of the pathnames
REM ===================================================================

set sibadmin=”D:\Program Files\StorageTek\Shared Virtual Array Administrator
3.1.0\bin\sibadmin.bat”

REM
REM Check Usage
REM we need STK_SCRIPTS_HOME to be set properly
REM We need the LUN Configuration File as an input parameter
REM ===========

if [%1] EQU [] goto s_usage
if [%2] NEQ [] goto s_usage

if [%STK_SCRIPTS_HOME%] EQU [] goto s_usage
set local_run_path=%STK_SCRIPTS_HOME%

if NOT EXIST %local_run_path%\umount.cmd goto s_usage
goto s_start
```

```
:s_usage

echo syntax error
echo Set STK_SCRIPTS_HOME environment variable properly.
echo usage: %0 LUN_configuration_file_name
goto s_end

:s_start

set lun_config_file=%1

REM
REM Setting default file names
REM =========================

set master_log=master.log
set lun_config_file=%1

REM
REM For Each LUN:
REM - Check and Unmount
REM ==================

REM [ delims is a tab in the following loop ]
FOR /F “tokens=1,2,3,4,5* delims= “ %%G in (‘findstr /V ^# %lun_config_file% 2^>nul’)
do (call %local_run_path%\umount.cmd %%H %%J)

echo. >> %local_run_path%\%master_log%

echo *********************************************** >>
%local_run_path%\%master_log%

echo. >> %local_run_path%\%master_log%

:a_end
ENDLOCAL
:s_end
```

umount.cmd

```
@echo off
setlocal enableextensions

REM
REM Update next line to reflect the absolute path of SVAA CLI Batch File
REM Use double quotes if spaces are part of the pathnames
REM ==================================================================
set sibadmin=”D:\Program Files\StorageTek\Shared Virtual Array Administrator
3.1.0\bin\sibadmin.bat”

REM
REM Check Usage - we need 2 parameters
REM - 1 - SVAPath Target LUN
REM - 2 - Target Partition Letter
REM =================================

if [%2] EQU [] goto s_usage
if [%3] NEQ [] goto s_usage
```

```
goto s_start

:s_usage

echo syntax error
echo usage: ^<%0 SVAPath_lun_number target_drive_letter^>
goto s_end

:s_start

set svapath_target_LUN=%1
set target_letter=%2

REM
REM Un-Mount Target LUN (Target Partition Letter is no more available to NT)
REM =======================================================================
echo - Syncing and Unmounting Target Partition... >> %local_run_path%\%master_log%
2>&1
%local_run_path%\stkassign sync %target_letter% >> %local_run_path%\%master_log% 2>&1
%local_run_path%\stkassign umount %target_letter% >> %local_run_path%\%master_log%
2>&1
if %ERRORLEVEL% NEQ 0 goto s_error

REM
REM Exclude Target LUN (SVAPath)
REM ============================
echo - Excluding Target LUN... >> %local_run_path%\%master_log% 2>&1
setsp -l %svapath_target_LUN% -e 1 >> %local_run_path%\%master_log% 2>&1
if %ERRORLEVEL% NEQ 0 goto s_error

goto s_end

:s_error
echo ==== ERROR encountered!!! See messages above. ****** >> %local_run_path%\%mas-
ter_log%

:s_end

echo ****** Target LUN dismounted. ****** >> %local_run_path%\%master_log%
ENDLOCAL
```

APPENDIX H: EXAMPLE OF SUCCESSFUL BACKUP (NO ADDITIONAL BACKUP SERVER)

master.log

```
==========================================================
***** C:\STK_scripts\SingleServer\release_luns.cmd starting on [Wed 10/09/2002
2:10p]

****** Processing LUN:
      SVAPath Target LUN:        26
      Target Paritition Letter:  P:
      Target Disk Name:          \Device\HarddiskVolume9
- Including Target LUN...
- Mounting Target Partition...
“P:” is now mapped to:
\Device\HarddiskVolume9
- Formatting LUNs...
The type of the file system is NTFS.
Enter current volume label for drive M: NWKtrg

WARNING, ALL DATA ON NON-REMOVABLE DISK
DRIVE P: WILL BE LOST!
Proceed with Format (Y/N)? Y
QuickFormatting 7036M
Creating file system structures.
Format complete.
   7205120 KB total disk space.
   7166620 KB are available.
- Syncing and Unmounting Target Partition...
Internal Disk Name = \\.\P:

Done
“P:” unmapped.
- Excluding Target LUN...
****** Target LUNs Released. ******

************************************************

****** C:\STK_scripts\SingleServer\release_luns.cmd ending on  [Wed 10/09/2002
2:10p]
.
.
==========================================================
****** C:\STK_scripts\SingleServer\snap_luns.cmd starting on [Wed 10/09/2002  2:10p]

****** Processing LUN:
      SVAPath Target LUN:        26
      Source Partition Letter:   L:
      Target Paritition Letter:  P:
      Target Disk Name:          \Device\HarddiskVolume9
- Including Target LUN...
- Mounting Target Partition...
“P:” is now mapped to:
\Device\HarddiskVolume9
- Snapping LUNs...
Starting CLI...
SIB9863I: SnapShot is completed.
- Syncing and Unmounting Target Partition...
Internal Disk Name = \\.\P:

Done
```

```
“P:” unmapped.
- Excluding Target LUN...
****** Target LUNs Snapped. ******
************************************************

****** C:\STK_scripts\SingleServer\snap_luns.cmd ending on  [Wed 10/09/2002  2:10p]
.
.
==========================================================
****** C:\STK_scripts\SingleServer\backup_luns.cmd NWK backup starting on [Wed
10/09/2002  2:10p]

- Including Target LUN...
- Mounting Target Partition...
“P:” is now mapped to:
\Device\HarddiskVolume9
****** Target LUNs mounted. ******

************************************************

- Syncing and Unmounting Target Partition...
Internal Disk Name = \\.\P:

Done
“P:” unmapped.
- Excluding Target LUN...
****** Target LUN dismounted. ******

************************************************
****** C:\STK_scripts\SingleServer\backup_luns.cmd ending on  [Wed 10/09/2002  2:16p]
```

APPENDIX I: EXAMPLE OF SUCCESSFUL BACKUP (DEDICATED BACKUP SERVER)

master.log (on the production server)

```
===========================================================
*D:\STK_scripts\BackupServer\source\snap_luns.cmd starting on [jeu. 10/10/2002
18:35]


****** Processing LUN:
      SVAPath Target LUN:           15
      Source Partition Letter:      M:
      Target Paritition Letter:     P:
      Target Disk Name:             \Device\HarddiskVolume12
- Including Target LUN...
- Mounting Target Partition...
"P:" is now mapped to:
\Device\HarddiskVolume12
- Snapping LUNs...
Starting CLI...
SIB9863I: SnapShot is completed.
****** Target LUNs Snapped. ******

*************************************************


****** D:\STK_scripts\BackupServer\source\snap_luns.cmd ending on  [jeu. 10/10/2002
18:35]
- Syncing and Unmounting Target Partition...
Internal Disk Name = \\.\P:

Done
"P:" unmapped.
- Excluding Target LUN...
****** Target LUN dismounted. ******
```

master.log (on the backup server)

```
===========================================================
***** C:\STK_scripts\BackupServer\target\release_luns.cmd starting on [Thu 10/10/2002
6:39p]

****** Processing LUN:
      SVAPath Target LUN:           26
      Target Paritition Letter:     P:
      Target Disk Name:             \Device\HarddiskVolume9
- Including Target LUN...
- Mounting Target Partition...
"P:" is now mapped to:
\Device\HarddiskVolume9
- Formatting LUNs...
The type of the file system is NTFS.
Enter current volume label for drive P: NWKtrg

WARNING, ALL DATA ON NON-REMOVABLE DISK
DRIVE P: WILL BE LOST!
Proceed with Format (Y/N)? Y
QuickFormatting 7036M
Creating file system structures.
Format complete.
```

```
   7205120 KB total disk space.
   7166620 KB are available.
- Syncing and Unmounting Target Partition...
Internal Disk Name = \\.\P:

Done
“P:” unmapped.
****** Target LUNs Released. ******

************************************************

****** C:\STK_scripts\BackupServer\target\release_luns.cmd ending on  [Thu 10/10/2002
6:39p]
.
.
==========================================================
****** C:\STK_scripts\BackupServer\target\backup_luns.cmd NWK backup starting on [Thu
10/10/2002  6:39p]

****** C:\STK_scripts\BackupServer\target\backup_luns.cmd ending on  [Thu 10/10/2002
6:41p]
```

After the release of the target LUNs, the SnapShot is performed on the data host via a rsh command.

ABOUT STORAGETEK®

Storage Technology Corporation (NYSE: STK), a $2 billion worldwide company with headquarters in Louisville, CO, has been delivering a broad range of storage management solutions designed for IT professionals for over 30 years. StorageTek offers solutions that are easy to manage, integrate well with existing infrastructures and allow universal access to data across servers, media types and storage networks. StorageTek's practical and safe storage solutions for tape automation, disk storage systems and storage integration, coupled with a global services network, provide IT professionals with confidence and know-how to manage their entire storage management ecosystem today and in the future.

StorageTek products are available through a worldwide network. For more information, visit www.storagetek.com, or call 1.800.275.4785 or 01.303.673.2800.

WORLD HEADQUARTERS

Storage Technology Corporation
One StorageTek Drive
Louisville, Colorado 80028 USA
1.800.877.9220 or 01.303.673.5151

MA 9128 C 04/04