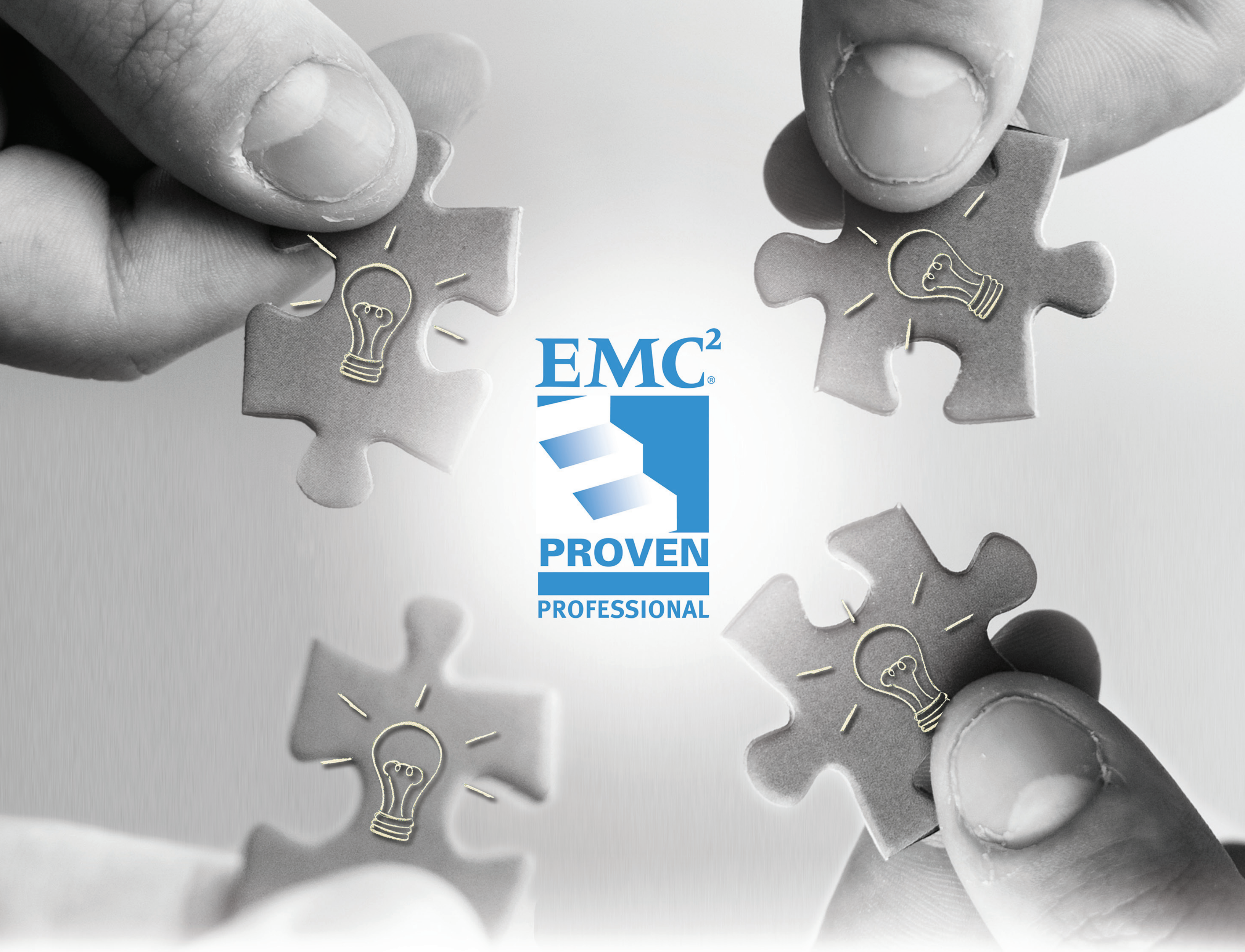


DD BOOST IMPLEMENTATION WITH NETWORKER

Crescenzo Oliviero
Advisory System Engineer
EMC Computer Systems Italia S.p.A
crescenzo.oliviero@emc.com

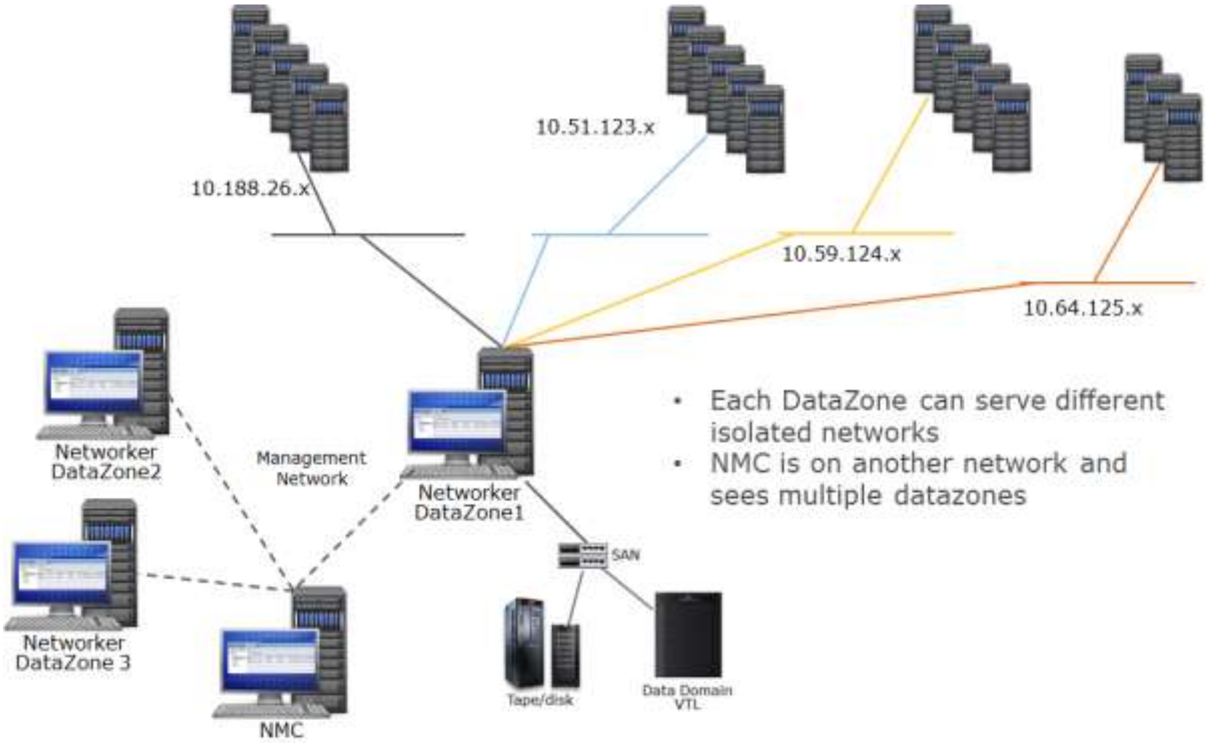
EMC²

Table of Contents

Disclaimer: The views, processes, or methodologies published in this article are those of the author. They do not necessarily reflect EMC Corporation’s views, processes, or methodologies.

Introduction

Enterprise customer environments are often implemented with complex network infrastructures where different networks are isolated for security reasons. A typical enterprise customer has a single NetWorker® Management Console (NMC) that manages different datazones and Data Domain® systems as seen in the figure below (only one datazone is completely expanded for sake of simplicity).



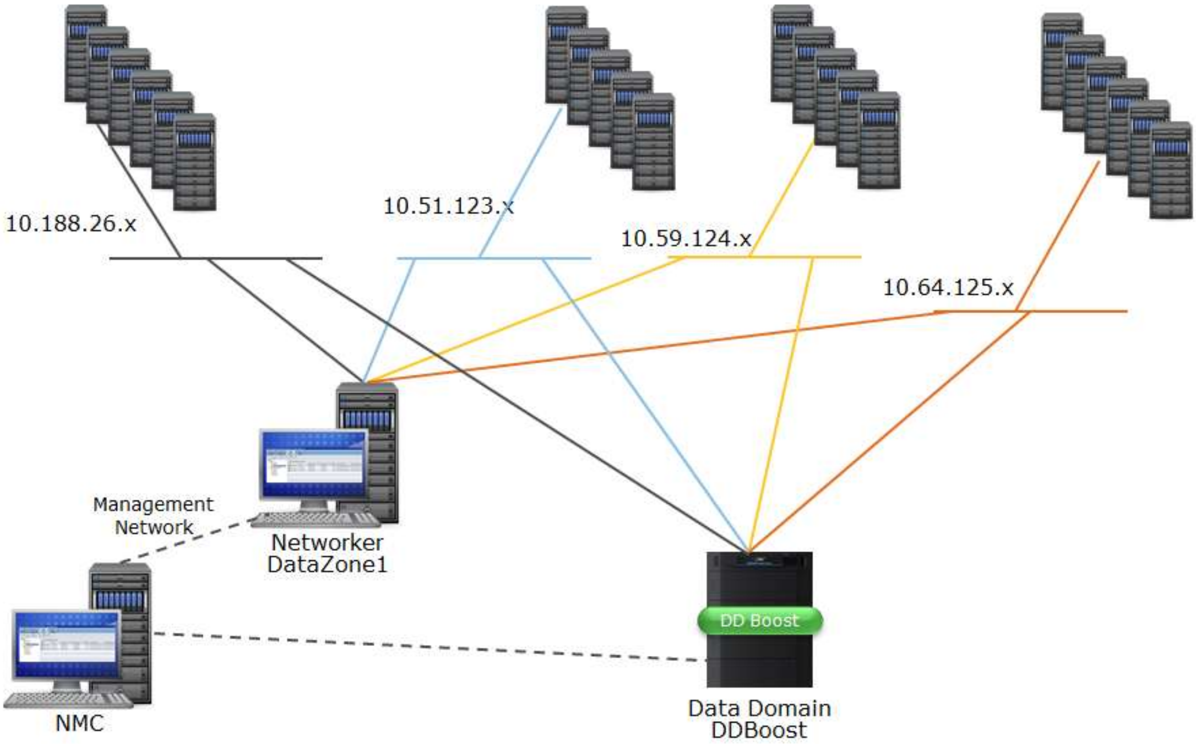
- Each DataZone can serve different isolated networks
- NMC is on another network and sees multiple datazones

In environments with only one datazone and NMC is implemented on NetWorker server itself or does not have complex network environment, the standard procedure to implement DD Boost is fine. Using the wizard, Boost backup will start in 5 minutes.

In environments where there are different isolated VLANs where clients reside, NMC will not be able to see all networks where data will flow and wizard configuration could fail.

To configure Boost devices, NMC must see both NetWorker servers and Data Domain systems over the backup network where the Boost device will be enabled and data will flow. For security reasons, this is typically not possible, and interferes with Boost device implementation.

The following picture shows the desired configuration where NMC is confined on the management network.

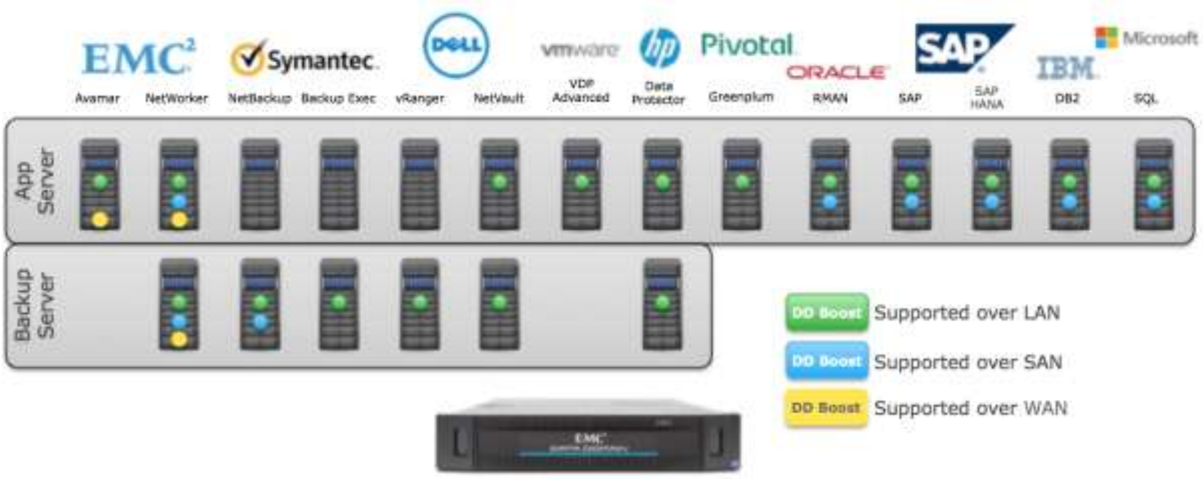


A procedure described later in this article will permit DD Boost device creation on different networks, enabling source deduplication that will avoid network congestion and reduce backup windows.

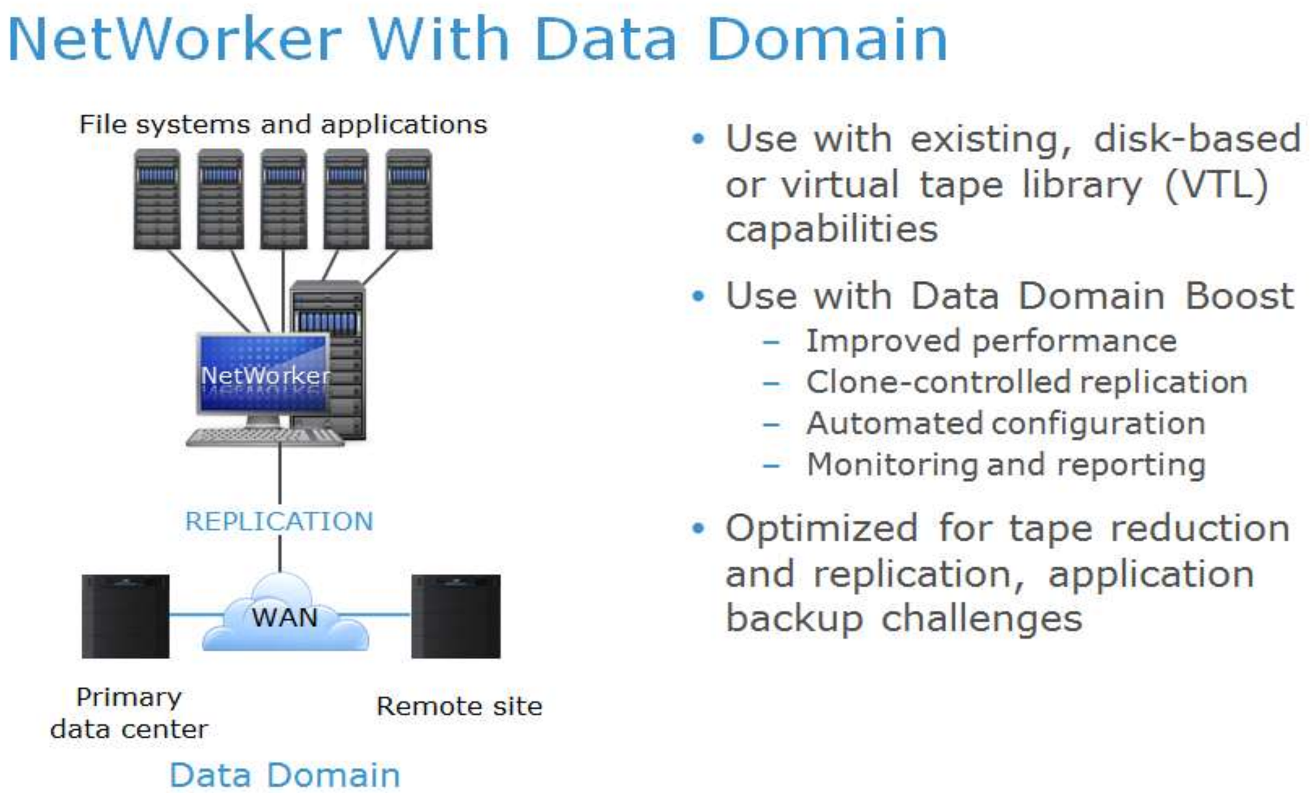
Why DD Boost is so important

DD Boost technology is important for many reasons. First, it can be implemented for performance purposes: it is already available for different environments and is able to shrink backup windows and reduce network traffic by up to 95%.

The figure below depicts what can be done today even without using NetWorker.



Moreover, using Data Domain with NetWorker will provide many other benefits as shown in the following figure.

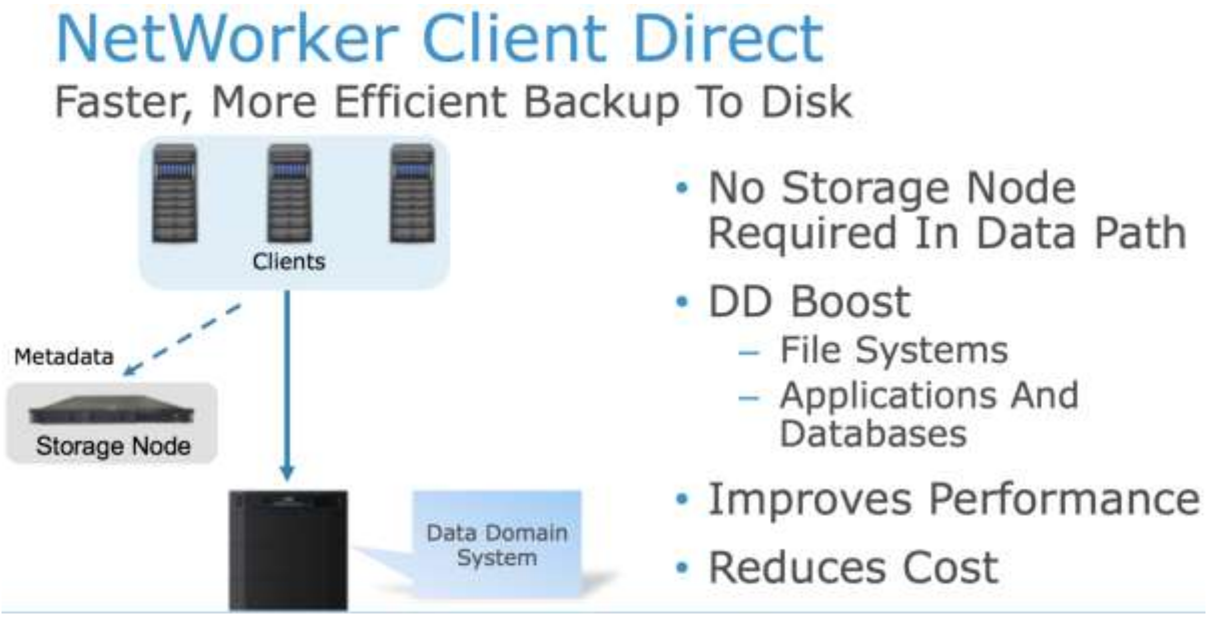


NetWorker With Data Domain

- Use with existing, disk-based or virtual tape library (VTL) capabilities
- Use with Data Domain Boost
  - Improved performance
  - Clone-controlled replication
  - Automated configuration
  - Monitoring and reporting
- Optimized for tape reduction and replication, application backup challenges

The most important feature is enablement of client direct feature; this will implement source deduplication anywhere!

Client direct is described below:



NetWorker Client Direct

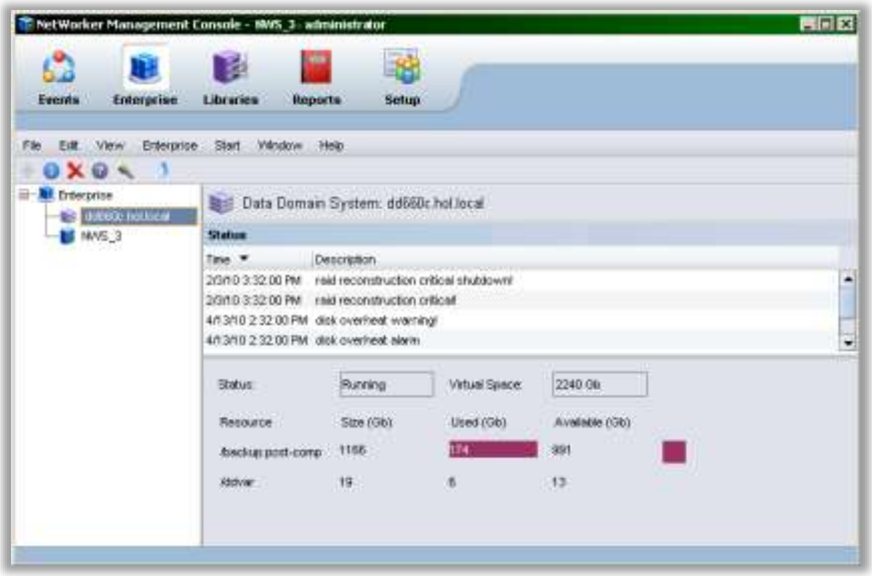
Faster, More Efficient Backup To Disk

- No Storage Node Required In Data Path
- DD Boost
  - File Systems
  - Applications And Databases
- Improves Performance
- Reduces Cost

Any client will send backup data directly to DataDomain device by-passing storage nodes, this will result in removal of any bottleneck increasing backup speed.

Easy to manage: DD monitoring from NMC

Integrating Data Domain with NMC enables the Data Domain monitoring feature as backup storage as seen below (SNMP). This reduces management effort since that backup manager will operate only on one console:



Data Domain System: dd660c.hol.local

Status

Time	Description
2/3/10 3:32:00 PM	raid reconstruction critical shutdown!
2/3/10 3:32:00 PM	raid reconstruction critical!
4/13/10 2:32:00 PM	disk overheat warning!
4/13/10 2:32:00 PM	disk overheat alarm

Status: Running Virtual Space: 2240 Gb

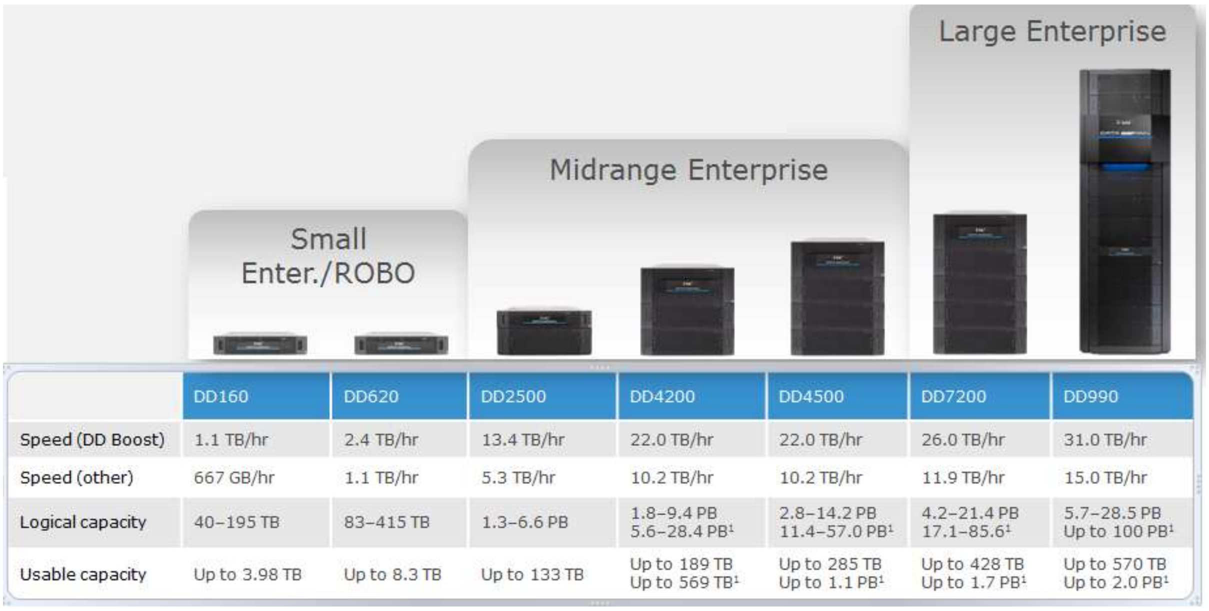
Resource	Size (Gb)	Used (Gb)	Available (Gb)
/backup post-comp	1165	174	991
/ddvar	19	6	13

This integration provides the status of the Data Domain device, the utilization, how much space is left, if it is fine, and so on. No other tools are required to gain visibility of the Data Domain device.

This feature simplifies use of Data Domain and relieves you of having to manage a virtual tape library, create drives, manage tapes, and more.

Backup and restore performance

The following figure depicts how DD Boost improves Data Domain performance and reduces backup windows of the entire infrastructure.


Large Enterprise
Midrange Enterprise
Small
Enter./ROBO

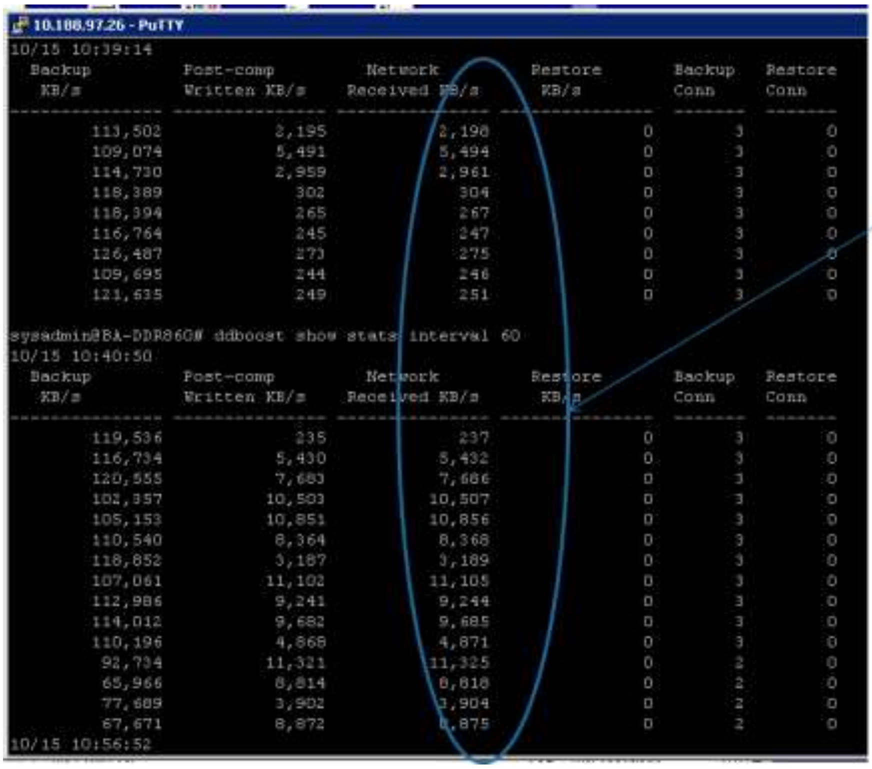
	DD160	DD620	DD2500	DD4200	DD4500	DD7200	DD990
Speed (DD Boost)	1.1 TB/hr	2.4 TB/hr	13.4 TB/hr	22.0 TB/hr	22.0 TB/hr	26.0 TB/hr	31.0 TB/hr
Speed (other)	667 GB/hr	1.1 TB/hr	5.3 TB/hr	10.2 TB/hr	10.2 TB/hr	11.9 TB/hr	15.0 TB/hr
Logical capacity	40–195 TB	83–415 TB	1.3–6.6 PB	1.8–9.4 PB 5.6–28.4 PB[1]	2.8–14.2 PB 11.4–57.0 PB[1]	4.2–21.4 PB 17.1–85.6[1]	5.7–28.5 PB Up to 100 PB[1]
Usable capacity	Up to 3.98 TB	Up to 8.3 TB	Up to 133 TB	Up to 189 TB Up to 569 TB[1]	Up to 285 TB Up to 1.1 PB[1]	Up to 428 TB Up to 1.7 PB[1]	Up to 570 TB Up to 2.0 PB[1]

DD Boost provides a performance improvement greater than 100% (more than doubling Data Domain performance), without impacting client performance or including hardware.

Network Traffic reduction and Source Deduplication

DD Boost deduplicates most of the data at source, close to where data lives, reducing network traffic, lowering CPU consumption, and shrinking backup windows.

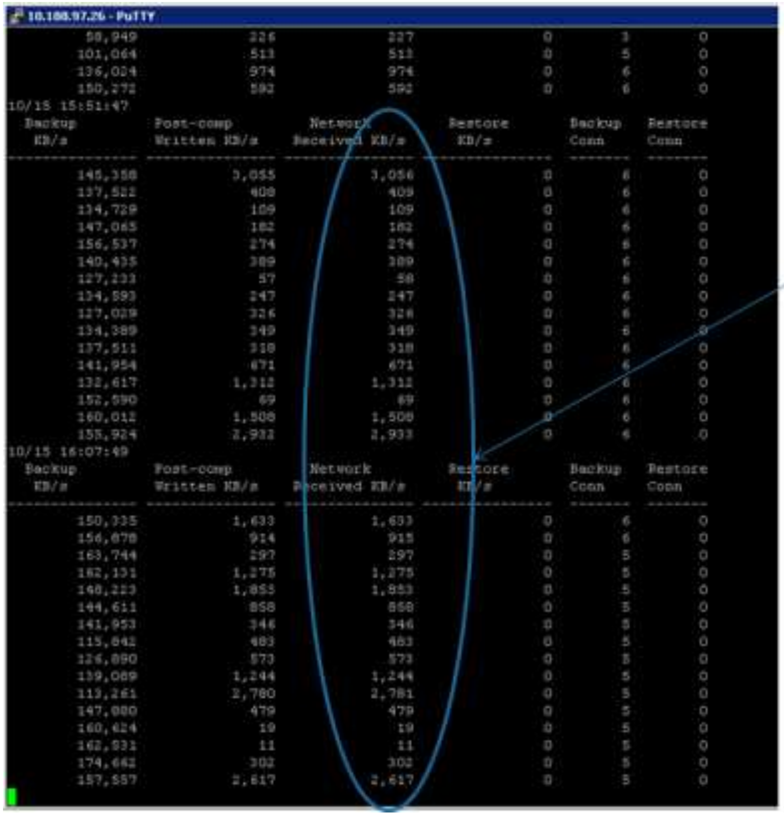
The following figure shows a test conducted during a Proof of Concept (POC). Data are sent to Data Domain system over the network. This is the first backup of virtual machines on a VMware farm.



```
10.188.97.26 - PuTTY
10/15 10:39:14
  Backup          Post-comp          Network          Restore         Backup   Restore
   KB/s          Written KB/s    Received KB/s         KB/s           Conn     Conn
--------------- --------------- --------------- ---------------   -------  -------
        113,502           2,195           2,198               0         3        0
        109,074           5,491           5,494               0         3        0
        114,730           2,959           2,961               0         3        0
        118,389             302             304               0         3        0
        118,394             265             267               0         3        0
        116,764             245             247               0         3        0
        126,487             273             275               0         3        0
        109,695             244             246               0         3        0
        121,635             249             251               0         3        0

sysadmin@BA-DDR860# ddboost show stats interval 60
10/15 10:40:50
  Backup          Post-comp          Network          Restore         Backup   Restore
   KB/s          Written KB/s    Received KB/s         KB/s           Conn     Conn
--------------- --------------- --------------- ---------------   -------  -------
        119,536             235             237               0         3        0
        116,734           5,430           5,432               0         3        0
        120,555           7,683           7,686               0         3        0
        102,357          10,503          10,507               0         3        0
        105,153          10,851          10,856               0         3        0
        110,540           8,364           8,368               0         3        0
        118,852           3,187           3,189               0         3        0
        107,061          11,102          11,105               0         3        0
        112,986           9,241           9,244               0         3        0
        114,012           9,682           9,685               0         3        0
        110,196           4,868           4,871               0         3        0
         92,734          11,321          11,325               0         2        0
         65,966           8,814           8,818               0         2        0
         77,689           3,902           3,904               0         2        0
         67,671           8,872           8,875               0         2        0
10/15 10:56:52
```

Data Transferred
Over the Network

Even at first backup, data sent over network are less than 10% of the total amount. At second backup—done 4 hours later—the amount of data sent over the LAN is negligible.



```
10.188.97.26 - PuTTY
         58,949             226             227               0         3        0
        101,064             513             513               0         5        0
        136,024             974             974               0         6        0
        150,272             592             592               0         6        0
10/15 15:51:47
  Backup          Post-comp          Network          Restore         Backup   Restore
   KB/s          Written KB/s    Received KB/s         KB/s           Conn     Conn
--------------- --------------- --------------- ---------------   -------  -------
        145,358           3,055           3,056               0         6        0
        137,522             408             409               0         6        0
        134,729             109             109               0         6        0
        147,065             182             182               0         6        0
        156,537             274             274               0         6        0
        140,435             389             389               0         6        0
        127,233              57              58               0         6        0
        134,593             247             247               0         6        0
        127,029             326             326               0         6        0
        134,389             349             349               0         6        0
        137,511             318             318               0         6        0
        141,954             671             671               0         6        0
        132,617           1,312           1,312               0         6        0
        152,590              69              69               0         6        0
        160,012           1,508           1,508               0         6        0
        155,924           2,932           2,933               0         6        0
10/15 16:07:49
  Backup          Post-comp          Network          Restore         Backup   Restore
   KB/s          Written KB/s    Received KB/s         KB/s           Conn     Conn
--------------- --------------- --------------- ---------------   -------  -------
        150,335           1,633           1,633               0         6        0
        156,878             914             915               0         6        0
        163,744             297             297               0         5        0
        162,131           1,275           1,275               0         5        0
        148,223           1,855           1,853               0         5        0
        144,611             858             858               0         5        0
        141,953             346             346               0         5        0
        115,842             483             483               0         5        0
        126,890             573             573               0         5        0
        139,089           1,244           1,244               0         5        0
        113,261           2,780           2,781               0         5        0
        147,880             479             479               0         5        0
        160,424              19              19               0         5        0
        162,931              11              11               0         5        0
        174,662             302             302               0         5        0
        157,557           2,617           2,617               0         5        0
```

Data Transferred
Over the Network: less than
1/100 th of the source data

This is a key reason why DD Boost is so important! No other backup software+backup applicance can do the same on every client on the infrastructure.

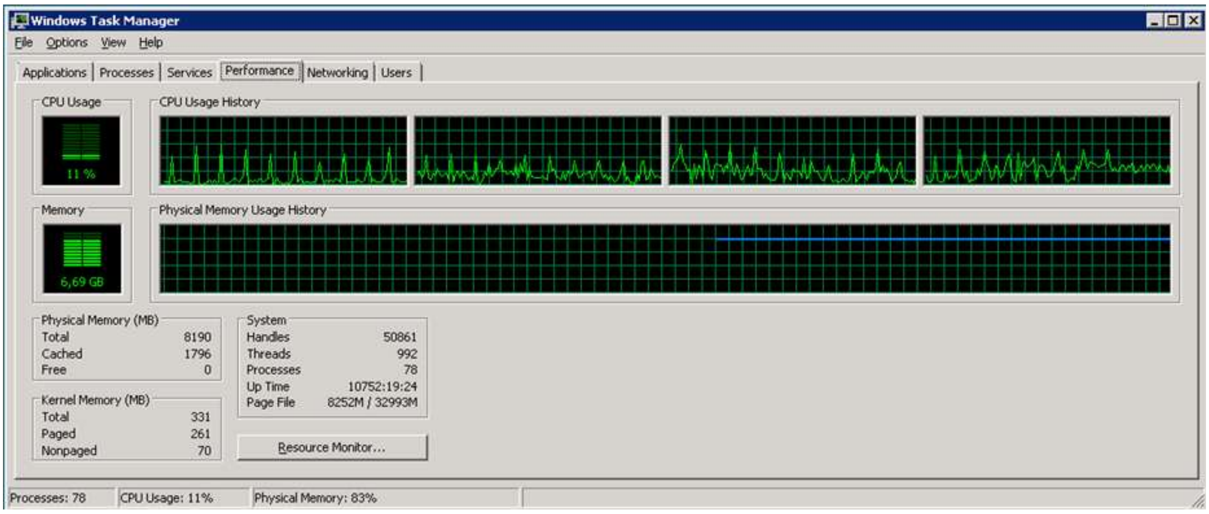
Network traffic reduction is vital in virtualized environments where consolidation could lead to many virtual machines being backed up at the same time, on the same ESX, using the same Gbit Ethernet connection to the backup infrastructure.

The above result is easily repeatable with a client installed within a virtual machine (this case is necessary when the virtual machine uses Raw Device Mapping which is not compatible with VADP snapshots), on physical systems, or on machines containing applications such as databases.

Client CPU consumption and performance

Source deduplication reduces CPU consumption on the client compared to a standard backup procedure where all data will flow to the backup device.

The screenshot below is an example of CPU consumption during a backup session.



Physical Memory (MB)
Total 8190
Cached 1796
Free 0

Kernel Memory (MB)
Total 331
Paged 261
Nonpaged 70

System
Handles 50861
Threads 992
Processes 78
Up Time 10752:19:24
Page File 8252M / 32993M

Processes: 78 | CPU Usage: 11% | Physical Memory: 83%

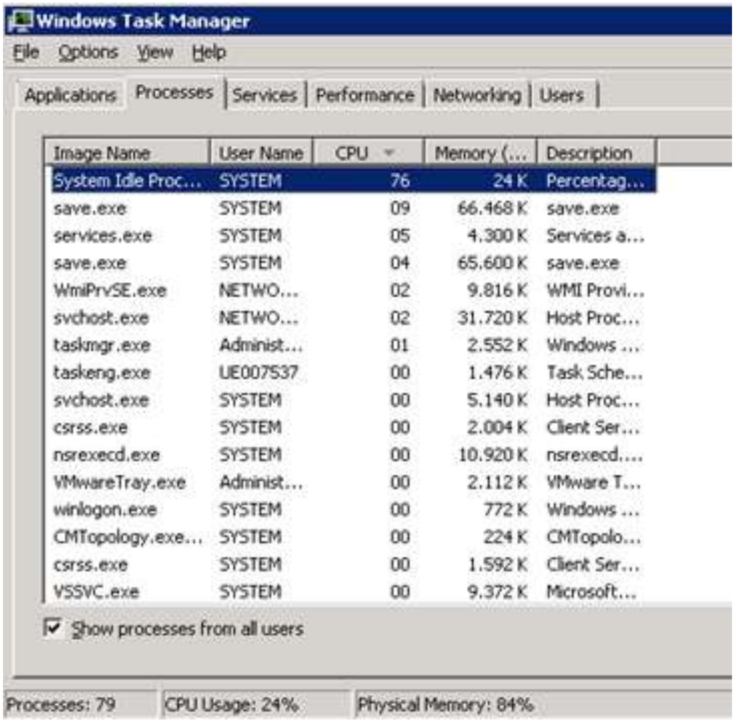
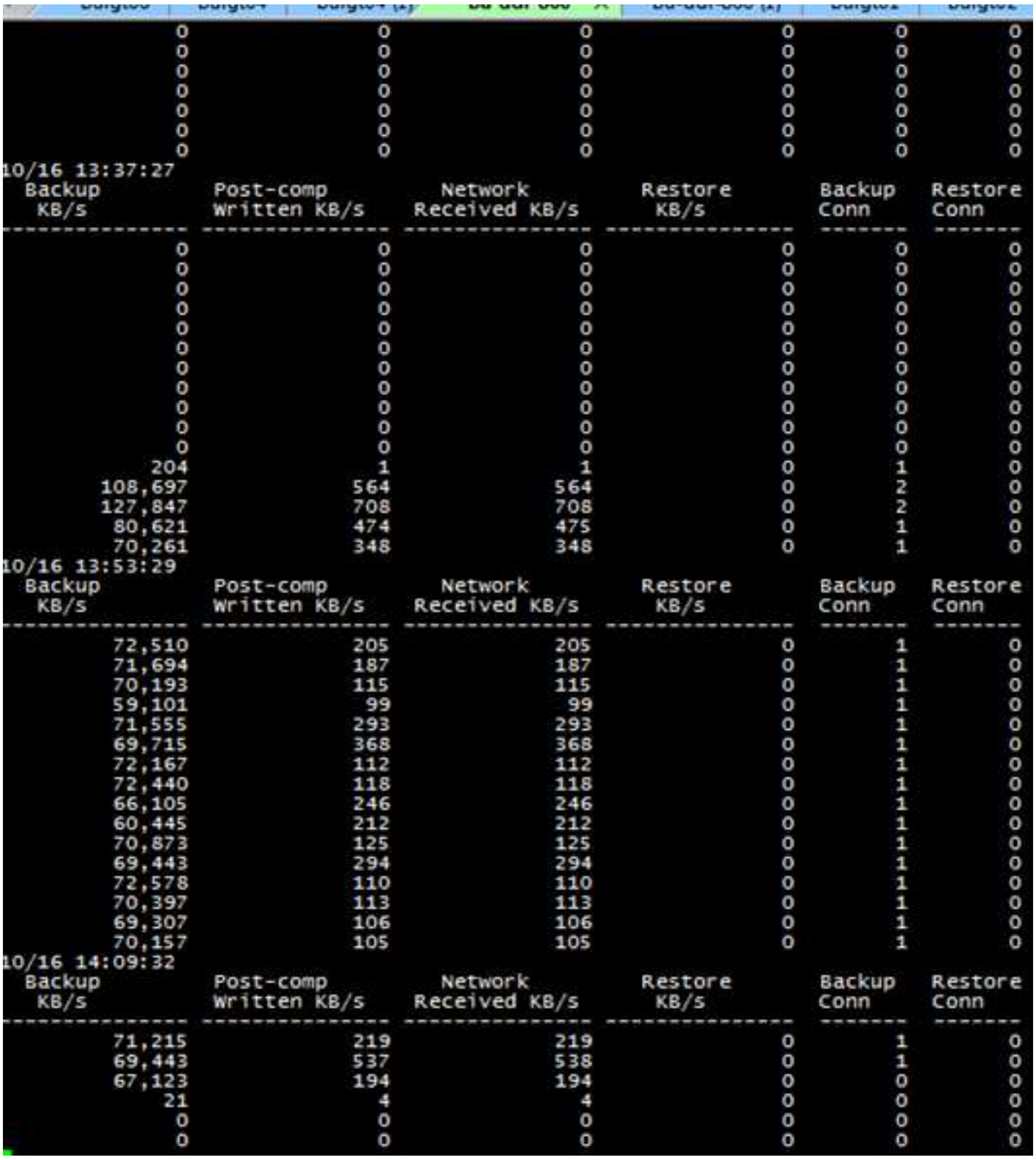

Windows Task Manager
File Options View Help
Applications Processes Services Performance Networking Users

Image Name	User Name	CPU	Memory (...	Description
System Idle Proc...	SYSTEM	76	24 K	Percentag...
save.exe	SYSTEM	09	66.468 K	save.exe
services.exe	SYSTEM	05	4.300 K	Services a...
save.exe	SYSTEM	04	65.600 K	save.exe
WmiPrvSE.exe	NETWO...	02	9.816 K	WMI Provi...
svchost.exe	NETWO...	02	31.720 K	Host Proc...
taskmgr.exe	Administ...	01	2.552 K	Windows ...
taskeng.exe	UE007537	00	1.476 K	Task Sche...
svchost.exe	SYSTEM	00	5.140 K	Host Proc...
csrss.exe	SYSTEM	00	2.004 K	Client Ser...
nsrexecd.exe	SYSTEM	00	10.920 K	nsrexecd....
VMwareTray.exe	Administ...	00	2.112 K	VMware T...
winlogon.exe	SYSTEM	00	772 K	Windows ...
CMTopology.exe...	SYSTEM	00	224 K	CMTopolo...
csrss.exe	SYSTEM	00	1.592 K	Client Ser...
VSSVC.exe	SYSTEM	00	9.372 K	Microsoft...

☑ Show processes from all users

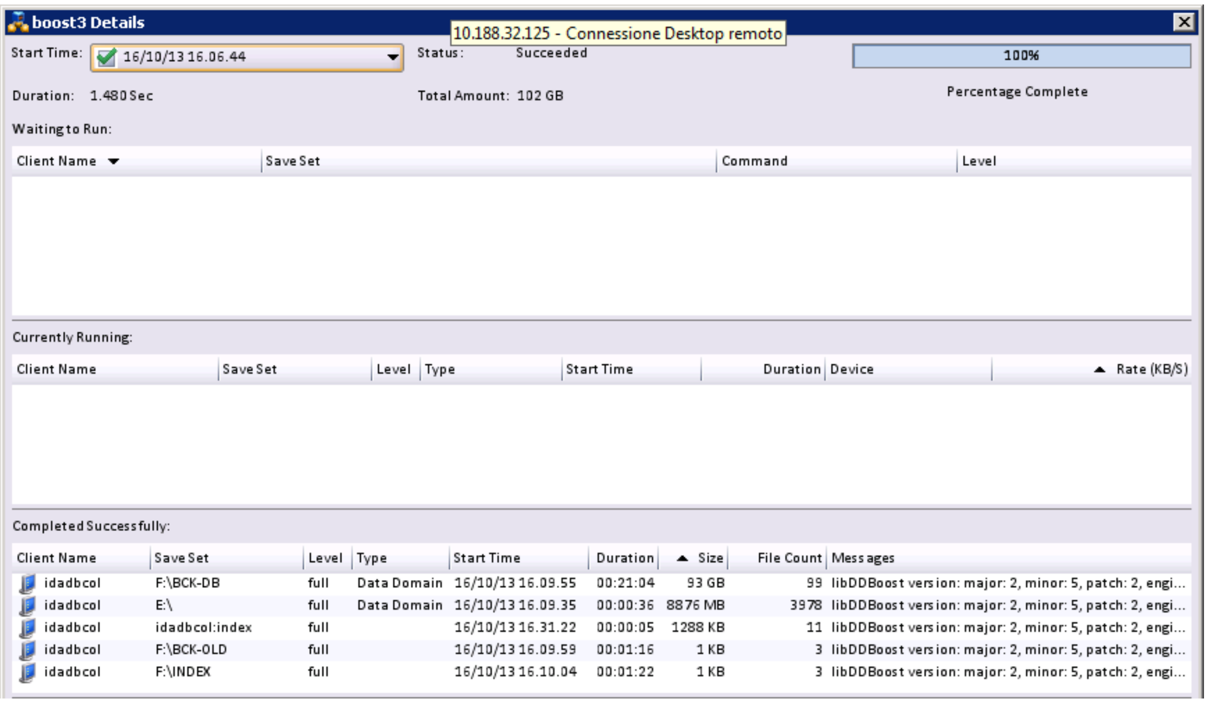
Processes: 79 CPU Usage: 24% Physical Memory: 84%

Processes save.exe (two parallel backup jobs) is consuming less than 13% CPU on this Windows server sending out about 100MB/s precompressed data using only a few KB/s of effective network bandwitdth (2nd backup session where deduplication is working a lot).



```
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
10/16 13:37:27
  Backup          Post-comp        Network        Restore       Backup  Restore
   KB/s          Written KB/s   Received KB/s      KB/s         Conn    Conn
-------------- -------------- -------------- -------------- ------- -------
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
           204              1              1              0       1       0
       108,697            564            564              0       2       0
       127,847            708            708              0       2       0
        80,621            474            475              0       1       0
        70,261            348            348              0       1       0
10/16 13:53:29
  Backup          Post-comp        Network        Restore       Backup  Restore
   KB/s          Written KB/s   Received KB/s      KB/s         Conn    Conn
-------------- -------------- -------------- -------------- ------- -------
        72,510            205            205              0       1       0
        71,694            187            187              0       1       0
        70,193            115            115              0       1       0
        59,101             99             99              0       1       0
        71,555            293            293              0       1       0
        69,715            368            368              0       1       0
        72,167            112            112              0       1       0
        72,440            118            118              0       1       0
        66,105            246            246              0       1       0
        60,445            212            212              0       1       0
        70,873            125            125              0       1       0
        69,443            294            294              0       1       0
        72,578            110            110              0       1       0
        70,397            113            113              0       1       0
        69,307            106            106              0       1       0
        70,157            105            105              0       1       0
10/16 14:09:32
  Backup          Post-comp        Network        Restore       Backup  Restore
   KB/s          Written KB/s   Received KB/s      KB/s         Conn    Conn
-------------- -------------- -------------- -------------- ------- -------
        71,215            219            219              0       1       0
        69,443            537            538              0       1       0
        67,123            194            194              0       0       0
            21              4              4              0       0       0
             0              0              0              0       0       0
             0              0              0              0       0       0
```

The following figure displays the backup job summary.



Start Time: 16/10/13 16.06.44

Status: Succeeded

Percentage Complete: 100%

Duration: 1.480 Sec

Total Amount: 102 GB

Waiting to Run:

Client Name	Save Set	Command	Level

Currently Running:

Client Name	Save Set	Level	Type	Start Time	Duration	Device	Rate (KB/S)

Completed Successfully:

Client Name	Save Set	Level	Type	Start Time	Duration	Size	File Count	Messages
idadbcol	F:\BCK-DB	full	Data Domain	16/10/13 16.09.55	00:21:04	93 GB	99	libDDBoost version: major: 2, minor: 5, patch: 2, engi...
idadbcol	E:\	full	Data Domain	16/10/13 16.09.35	00:00:36	8876 MB	3978	libDDBoost version: major: 2, minor: 5, patch: 2, engi...
idadbcol	idadbcol:index	full		16/10/13 16.31.22	00:00:05	1288 KB	11	libDDBoost version: major: 2, minor: 5, patch: 2, engi...
idadbcol	F:\BCK-OLD	full		16/10/13 16.09.59	00:01:16	1 KB	3	libDDBoost version: major: 2, minor: 5, patch: 2, engi...
idadbcol	F:\INDEX	full		16/10/13 16.10.04	00:01:22	1 KB	3	libDDBoost version: major: 2, minor: 5, patch: 2, engi...

The figure shows that about 8GB backs up in 36 seconds (first save.exe) and 93GB of data backs up in about 21 minutes (second save.exe process).

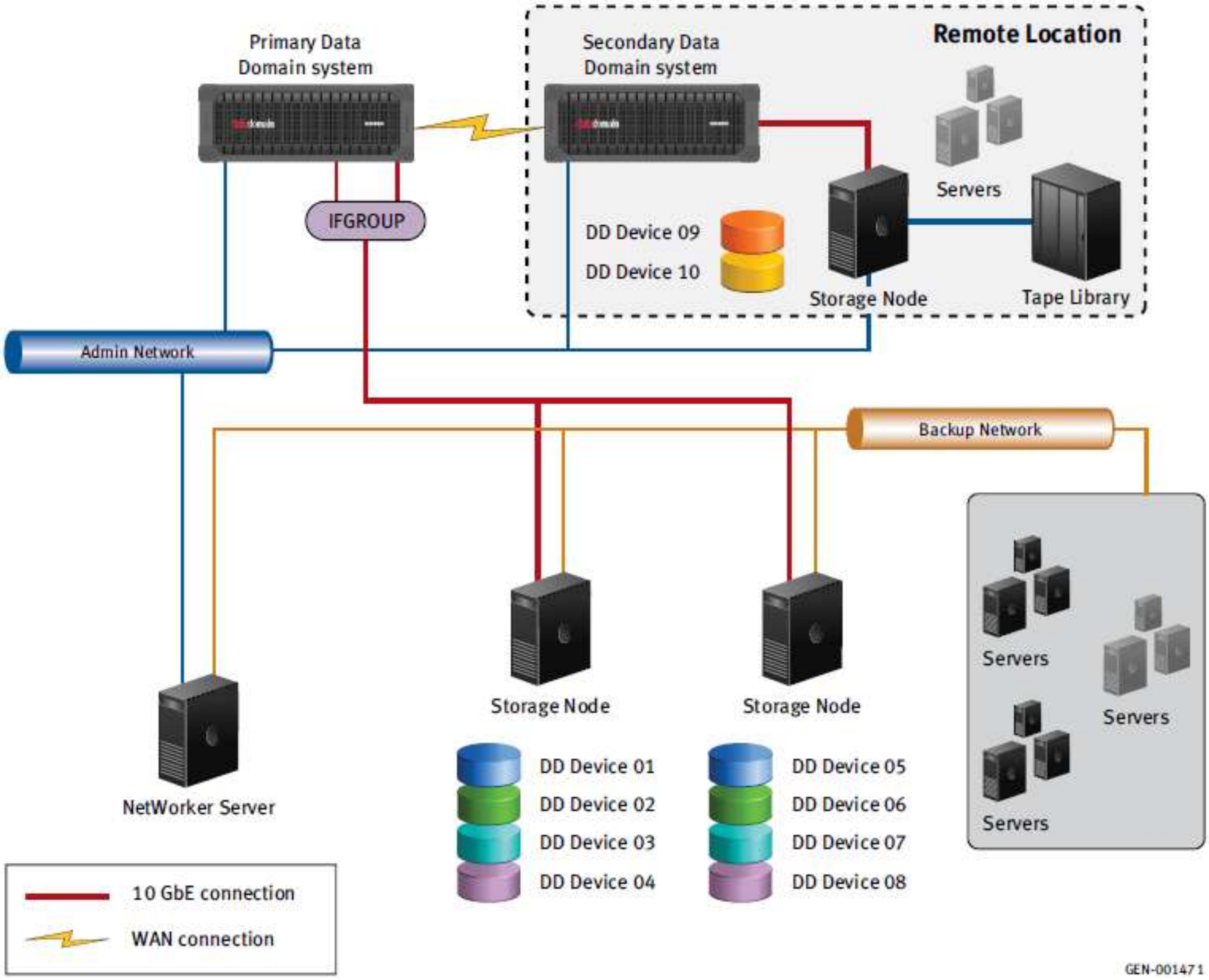
Clone controlled replication

Another benefit of DD Boost implementation with NetWorker is clone-controlled replication.

Cloning data, even to another site, is a simple clone policy job to be implemented only on NetWorker server. Behind the scenes, this will use Data Domain deduplication to reduce bandwidth requirements. NetWorker will know at any time that there are other copies of the data on another site on another Data Domain with the possibility of restoring data from the second copy.

As of NetWorker 8.1, the clone process will start at the end of each job reducing time to disaster recovery with an elegant and low-cost backup solution.

The figure below is an example of clone infrastructure (10 Gbit and storage nodes are not strictly required).



The gain here with DD Boost is that clone policy are done on NetWorker which knows where clones of data are and is able to restore it on a remote site or on a primary site if network bandwidth between sites is enough to comply with SLA.

Nothing has to be done on the Data Domain side (replication is managed by NetWorker).

Implementation Procedure

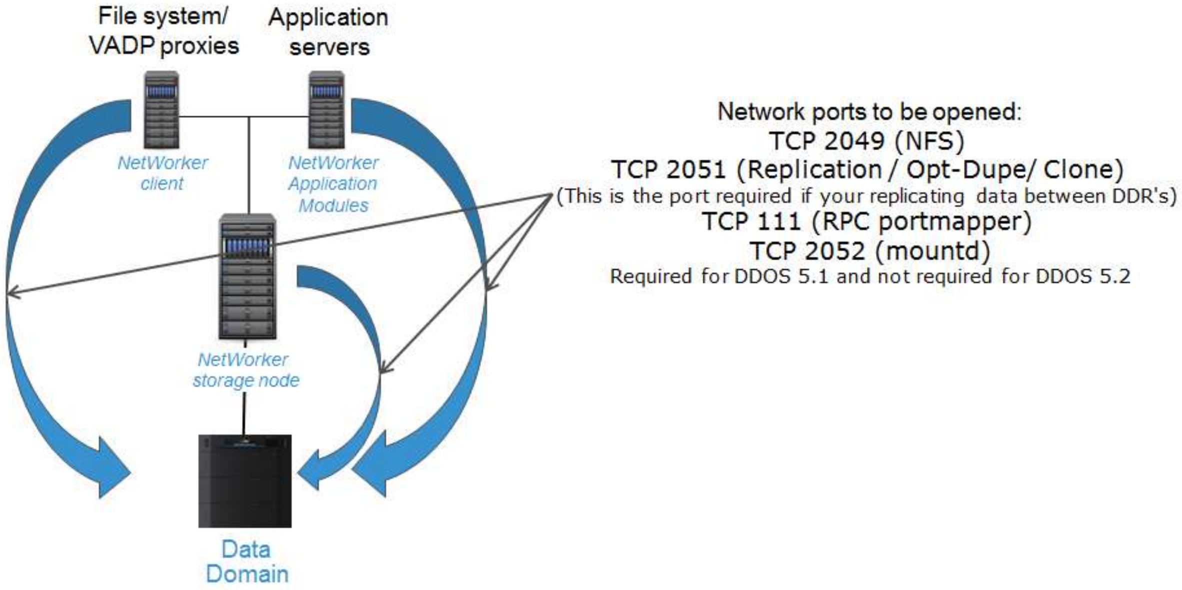
Network rules to be satisfied in order to implement DD Boost are shown below.

DD Boost Network Requirements

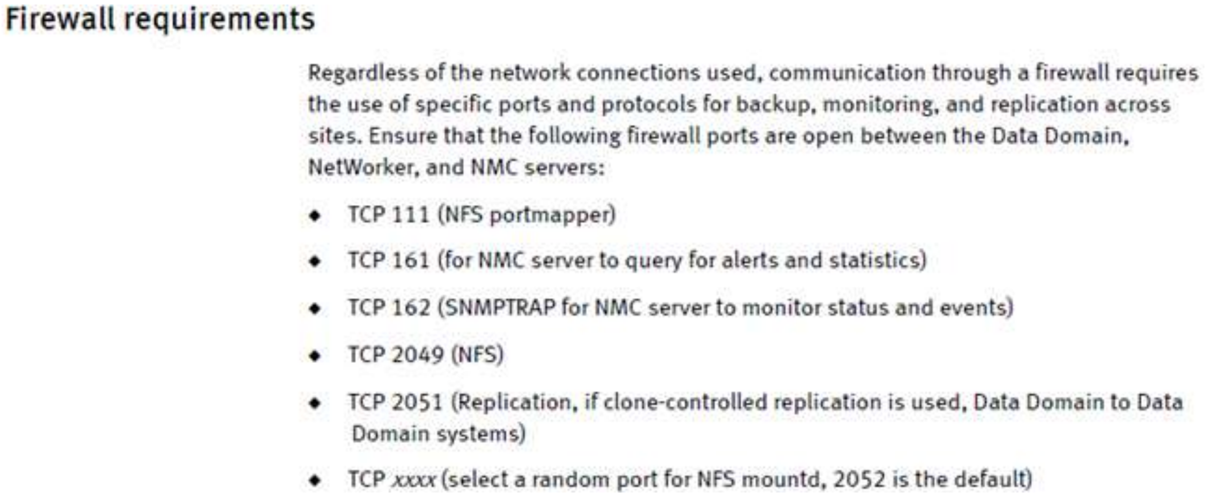
To realize full benefits of DD Boost implementation, each client must be able to see the DD Boost device directly. If this is satisfied, data will go directly to Data Domain, by-passing storage nodes and NetWorker server.

Clients, NetWorker servers, and storage nodes must be able to resolve the Data Domain IP, use DNS, or local host file to have this implemented.

Data flow is shown in the following figure where network requirement is also shown.


File system/ VADP proxies
Application servers
NetWorker client
NetWorker Application Modules
NetWorker storage node
Data Domain
Network ports to be opened:
TCP 2049 (NFS)
TCP 2051 (Replication / Opt-Dupe/ Clone)
(This is the port required if your replicating data between DDR's)
TCP 111 (RPC portmapper)
TCP 2052 (mountd)
Required for DDOS 5.1 and not required for DDOS 5.2

The following are the network ports that must be opened to ensure correct implementation of DD Boost among NetWorker server, Data Domain, and NMC.



Firewall requirements

Regardless of the network connections used, communication through a firewall requires the use of specific ports and protocols for backup, monitoring, and replication across sites. Ensure that the following firewall ports are open between the Data Domain, NetWorker, and NMC servers:

- TCP 111 (NFS portmapper)
- TCP 161 (for NMC server to query for alerts and statistics)
- TCP 162 (SNMPTRAP for NMC server to monitor status and events)
- TCP 2049 (NFS)
- TCP 2051 (Replication, if clone-controlled replication is used, Data Domain to Data Domain systems)
- TCP *xxxx* (select a random port for NFS mountd, 2052 is the default)

NMC is confined in a management network for security reasons and is typically is not able to access Data Domain over all the network where data will flow.

In any case, there must be a direct connection between clients and Data Domain to implement DD Boost with direct client.

Manual DD Boost implementation

- Enable DD Boost on Data Domain as usual

Add a DD Boost user

```
# user add username [password password]
```

For example, to add a user with a login name of crexboost and a password of abc123 with administrative privilege, enter:

```
# user add crexboost password abc123
```

The user must be configured in the application in order to connect to the Data Domain system.

Set the DD Boost user by entering:

```
# DD Boost set user-name crexboost
DD Boost user set to username
Previous user: none set
```

Note: Only one DD Boost user can be configured at a time for DD Boost access on a Data Domain system. The username and password must have already been set up on the Data Domain system using the `DD Boost CLI` command.

Enable DD Boost by entering:

```
# DD Boost enable
DD Boost enabled
```

- Create Storage Units

  On the Data Domain system, enter:

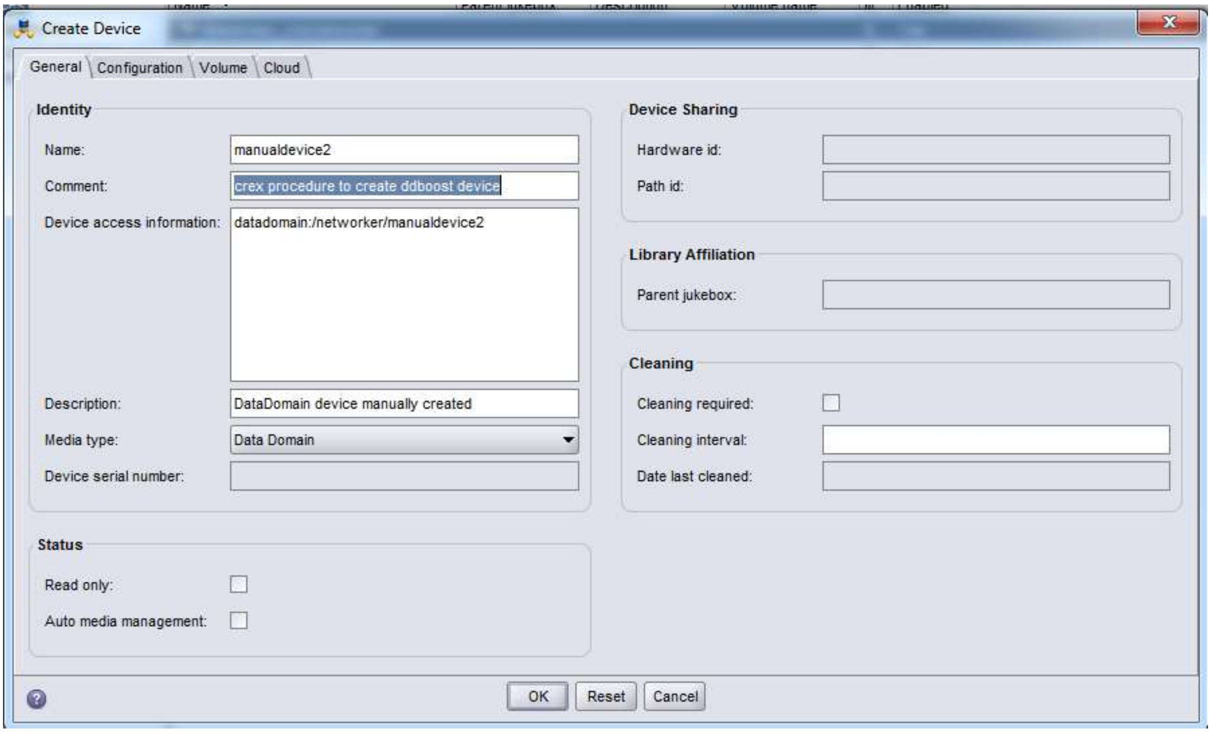
```
# DD Boost storage-unit create storage_unit_name-su
```

where each SU name is unique and tipically is the name of NetWorker datazone (NetWorker hostname).

Repeat the procedure for each Storage unit that is to be created.

- Log on to the NMC and go to the datazone where you are implementing DD Boost. Create a generic device paying attention to the following three parameters:

  1. Device access information: use DNS_Data Domain_Name:/StorageunitName/devicename
  2. Media type: must be Data Domain
  3. Remote user and password (the one you set on Data Domain side as DD Boost user)



Name: manualdevice2

Comment: crex procedure to create ddboost device

Device access information: datadomain:/networker/manualdevice2

Description: DataDomain device manually created

Media type: Data Domain

Device serial number:

Hardware id:

Path id:

Parent jukebox:

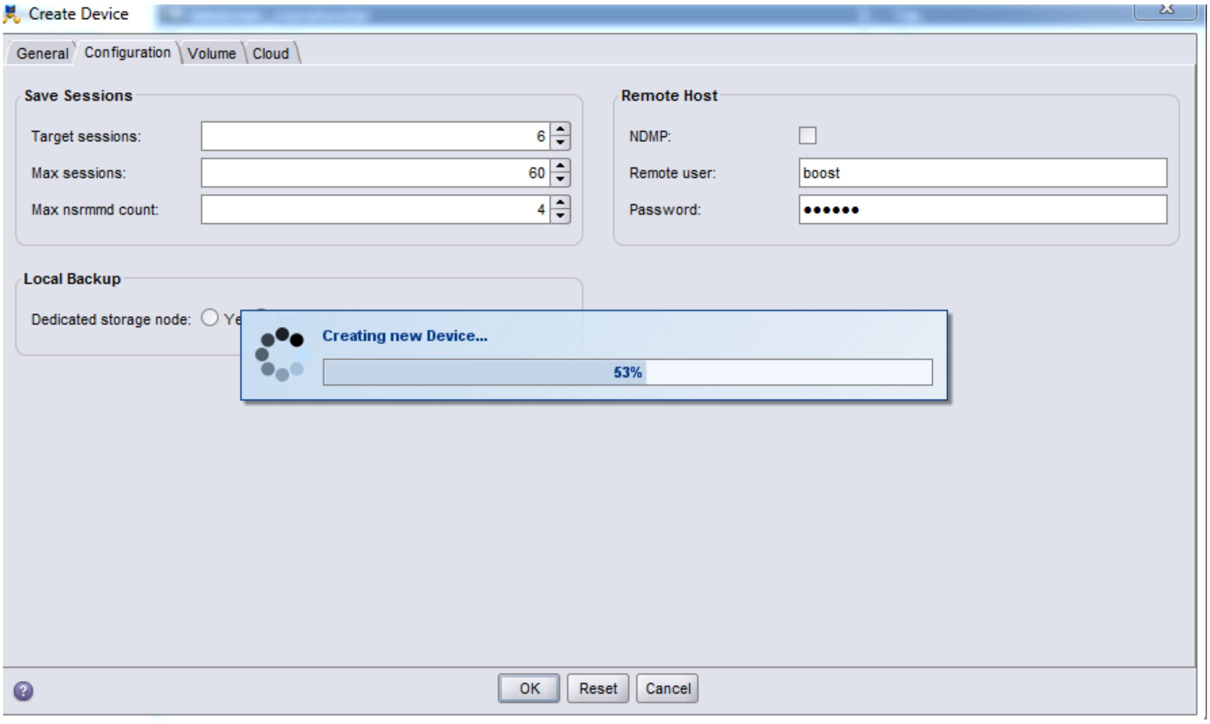
Cleaning required: □

Cleaning interval:

Date last cleaned:

Read only: □

Auto media management: □



Save Sessions

Target sessions: 6

Max sessions: 60

Max nsrmmd count: 4

Remote Host

NDMP: □

Remote user: boost

Password: ••••••

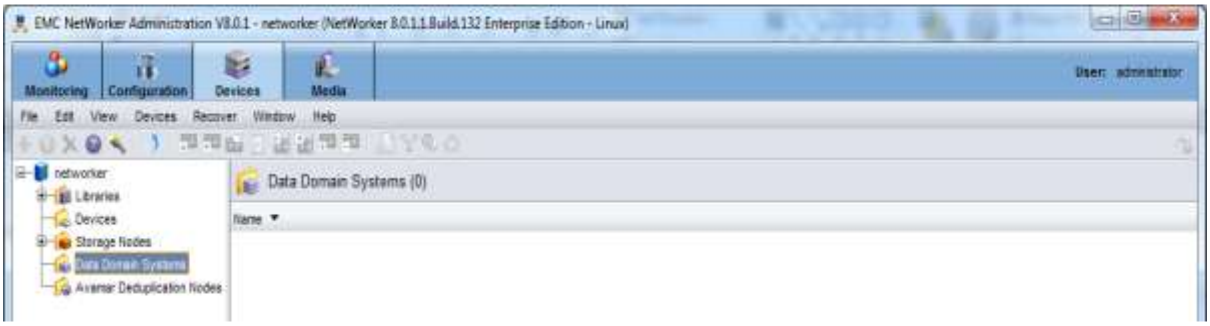
Local Backup

Dedicated storage node: ○ Ye

Creating new Device...

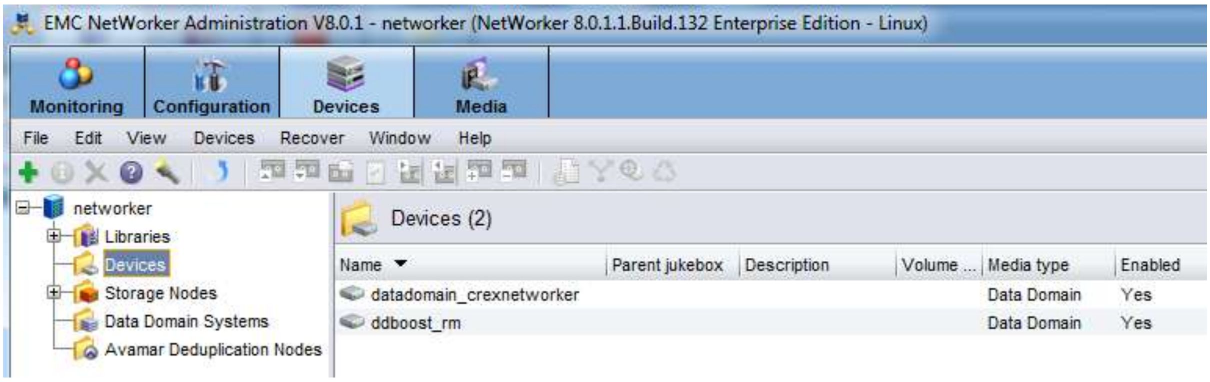
53%

At this point, you are able to see newer Data Domain device from NetWorker GUI.



EMC NetWorker Administration V8.0.1 - networker (NetWorker 8.0.1.1.Build.132 Enterprise Edition - Linux)

Data Domain Systems (0)

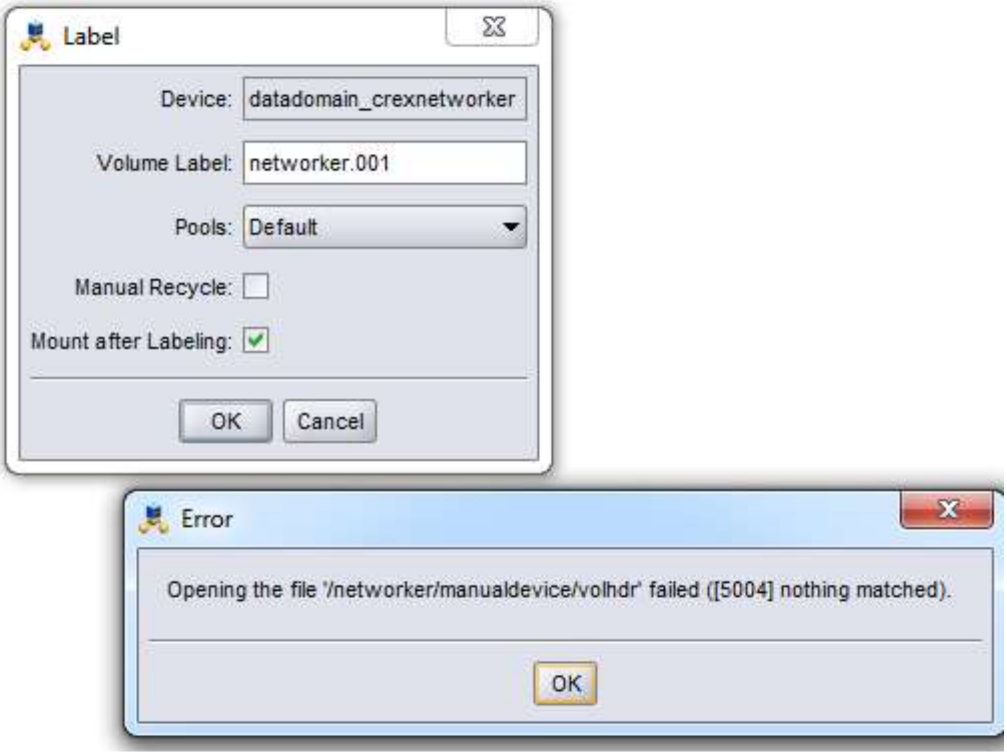


EMC NetWorker Administration V8.0.1 - networker (NetWorker 8.0.1.1.Build.132 Enterprise Edition - Linux)

Devices (2)

Name	Parent jukebox	Description	Volume ...	Media type	Enabled
datadomain_crexnetworker				Data Domain	Yes
ddboost_rm				Data Domain	Yes

You are not able to mount and label new devices at this time. Since we are not using a Wizard, we missed folders creation on the Data Domain side (though Wizard will do it automatically, NMC must see Data Domain).



Device: datadomain_crexnetworker

Volume Label: networker.001

Pools: Default

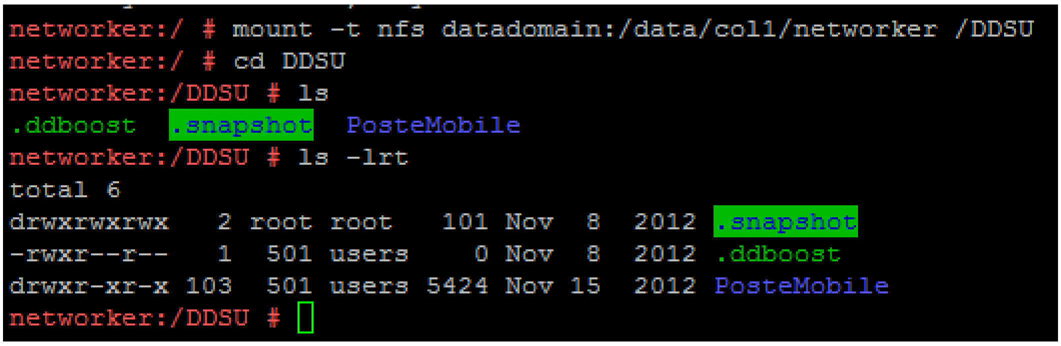
Manual Recycle: □

Mount after Labeling: ☑

Opening the file '/networker/manualdevice/volhdr' failed ([5004] nothing matched).

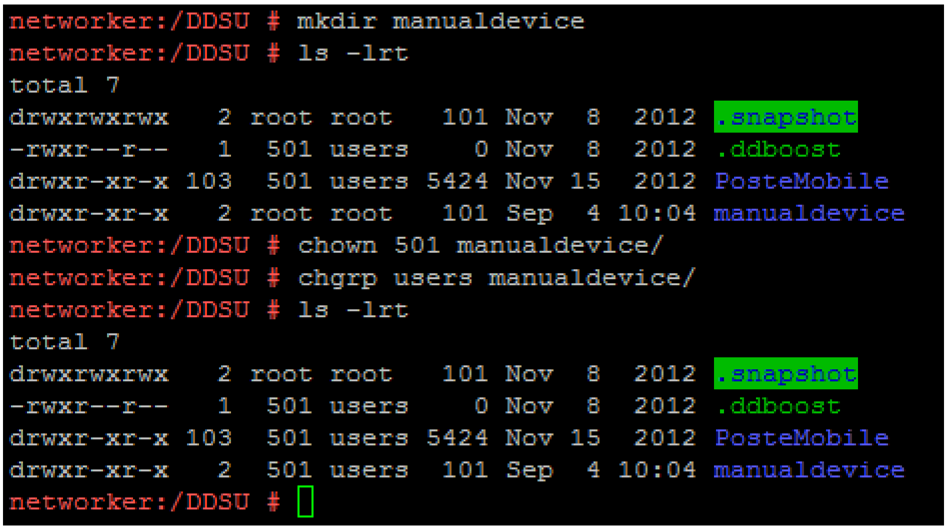
To solve this, mount via NFS or CIFS Data Domain folder and create folder structure manually. The following example is for a Linux box where network name of Data Domain is Data Domain, storage unit created on Data Domain is named NetWorker (dns name of NetWorker server is NetWorker), mounting it on local folder /DDSU:

Mount –t nfs Data Domain_name:/data/col1/StorageUnit /local_folder_to_mount



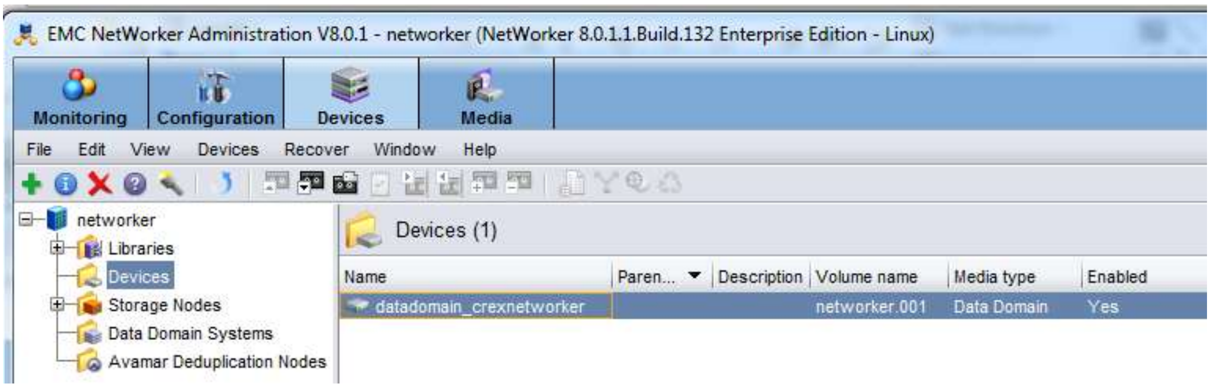
```
networker:/ # mount -t nfs datadomain:/data/col1/networker /DDSU
networker:/ # cd DDSU
networker:/DDSU # ls
.ddboost  .snapshot  PosteMobile
networker:/DDSU # ls -lrt
total 6
drwxrwxrwx   2 root root   101 Nov  8  2012 .snapshot
-rwxr--r--   1  501 users    0 Nov  8  2012 .ddboost
drwxr-xr-x 103  501 users 5424 Nov 15  2012 PosteMobile
networker:/DDSU # 
```

Create a folder with DD Boost device name (*mkdir manualdevice*) and change its owner to match DD Boost user.



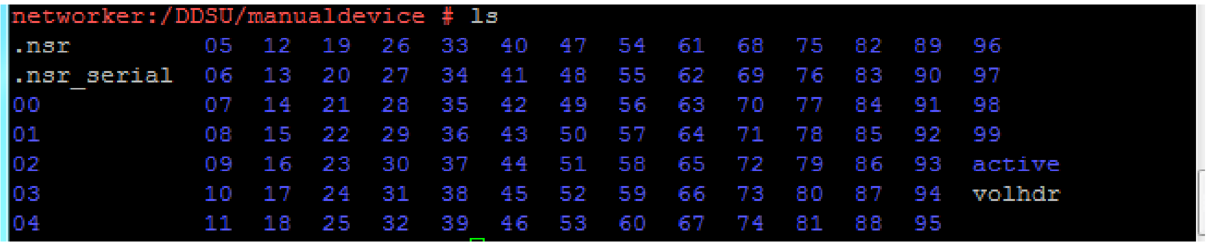
```
networker:/DDSU # mkdir manualdevice
networker:/DDSU # ls -lrt
total 7
drwxrwxrwx   2 root root   101 Nov  8  2012 .snapshot
-rwxr--r--   1  501 users    0 Nov  8  2012 .ddboost
drwxr-xr-x 103  501 users 5424 Nov 15  2012 PosteMobile
drwxr-xr-x   2 root root   101 Sep  4 10:04 manualdevice
networker:/DDSU # chown 501 manualdevice/
networker:/DDSU # chgrp users manualdevice/
networker:/DDSU # ls -lrt
total 7
drwxrwxrwx   2 root root   101 Nov  8  2012 .snapshot
-rwxr--r--   1  501 users    0 Nov  8  2012 .ddboost
drwxr-xr-x 103  501 users 5424 Nov 15  2012 PosteMobile
drwxr-xr-x   2  501 users  101 Sep  4 10:04 manualdevice
networker:/DDSU # 
```

You can now mount and label the newer device on the NetWorker side since you created the folder under storage unit.



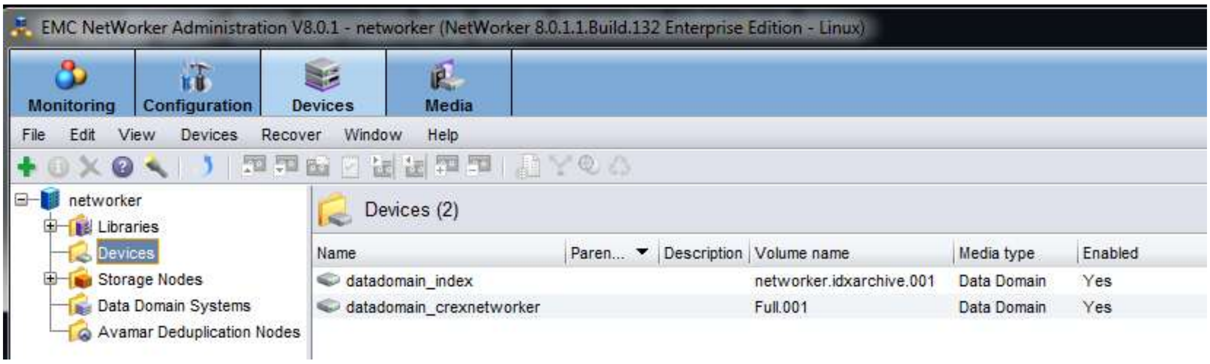
Name	Paren...	Description	Volume name	Media type	Enabled
datadomain_crexnetworker			networker.001	Data Domain	Yes

This is the effect on the Linux folder we mounted before. NetWorker created all Boost structure on it:



```
networker:/DDSU/manualdevice # ls
.nsr         05  12  19  26  33  40  47  54  61  68  75  82  89  96
.nsr_serial  06  13  20  27  34  41  48  55  62  69  76  83  90  97
00           07  14  21  28  35  42  49  56  63  70  77  84  91  98
01           08  15  22  29  36  43  50  57  64  71  78  85  92  99
02           09  16  23  30  37  44  51  58  65  72  79  86  93  active
03           10  17  24  31  38  45  52  59  66  73  80  87  94  volhdr
04           11  18  25  32  39  46  53  60  67  74  81  88  95
```

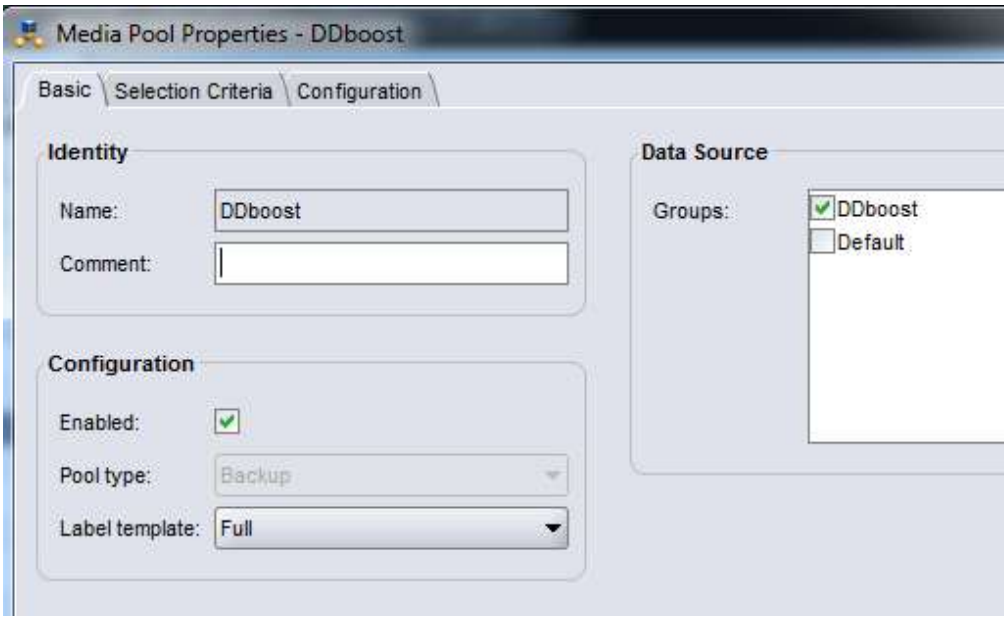
You can now label the device and create a pool to write data on it.



EMC NetWorker Administration V8.0.1 - networker (NetWorker 8.0.1.1.Build.132 Enterprise Edition - Linux)
Monitoring
Configuration
Devices
Media
File
Edit
View
Devices
Recover
Window
Help
networker
Libraries
Devices
Storage Nodes
Data Domain Systems
Avamar Deduplication Nodes

Devices (2)

Name	Paren...	Description	Volume name	Media type	Enabled
datadomain_index			networker.idxarchive.001	Data Domain	Yes
datadomain_crexnetworker			Full.001	Data Domain	Yes



Identity

Name: DDboost

Comment:

Configuration

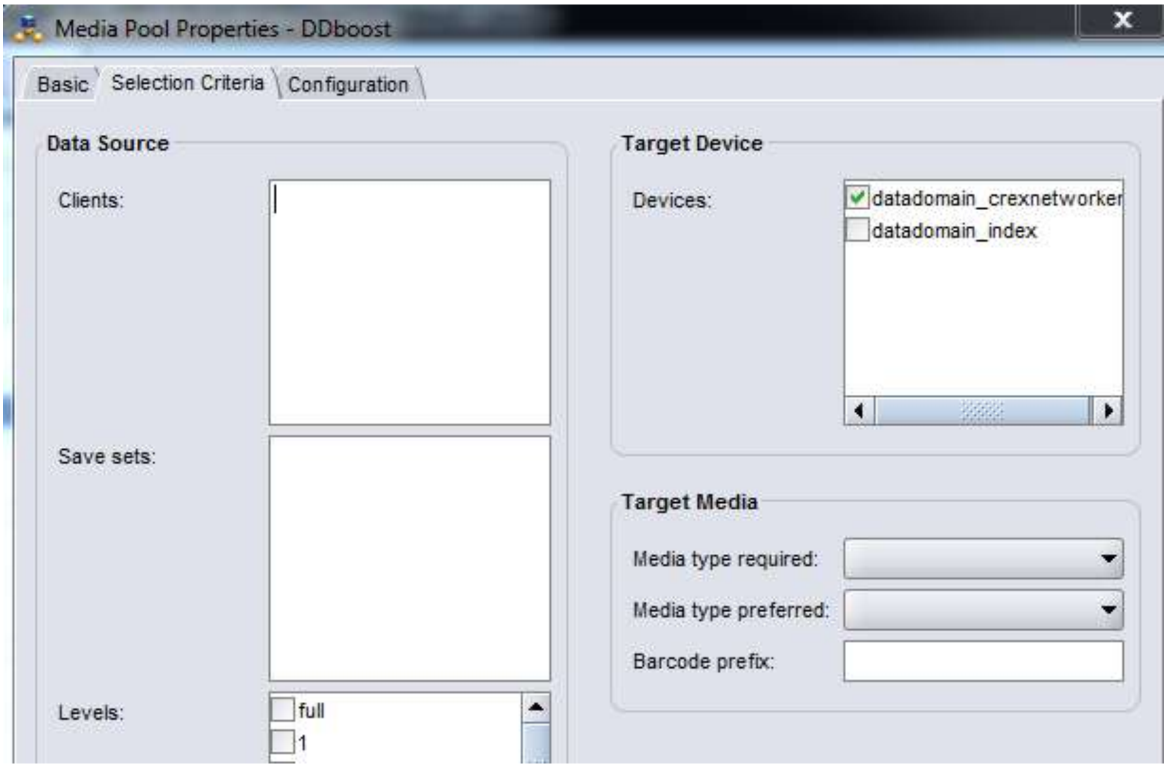
Enabled: ☑

Pool type: Backup

Label template: Full

Data Source

Groups: ☑ DDboost ☐ Default



Data Source

Clients:

Save sets:

Levels:
□ full
□ 1

Target Device

Devices:
☑ datadomain_crexnetworker
□ datadomain_index

Target Media

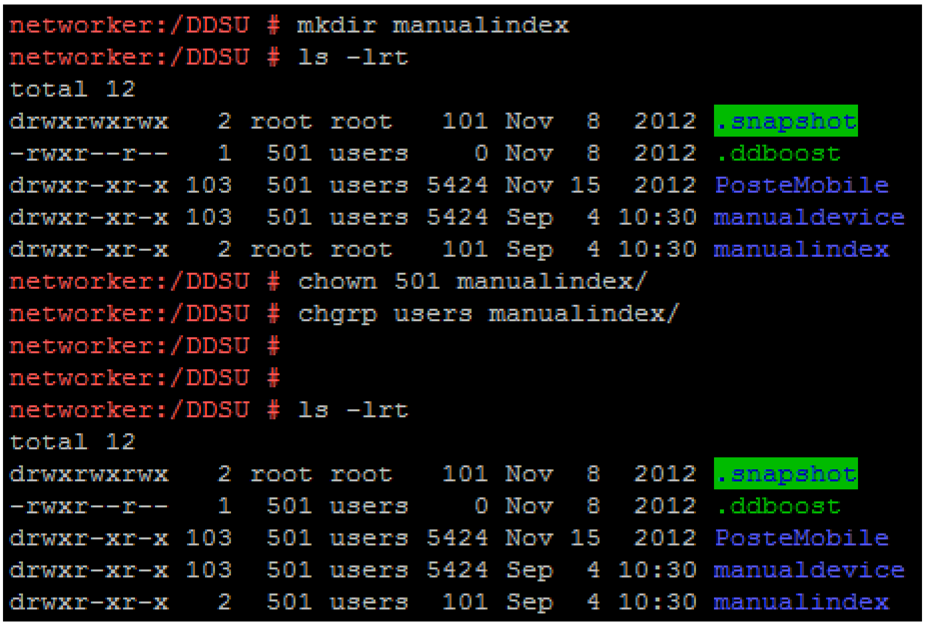
Media type required:

Media type preferred:

Barcode prefix:

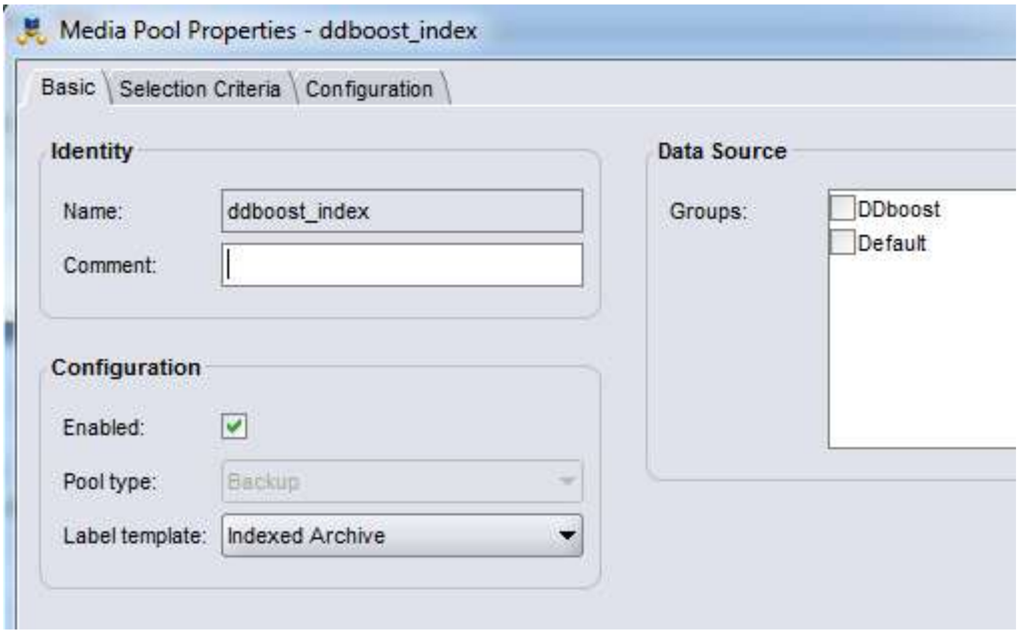
The same procedure applies when creating an indexing device on Data Domain.

Mkdir /DDSU/manualindex



```
networker:/DDSU # mkdir manualindex
networker:/DDSU # ls -lrt
total 12
drwxrwxrwx   2 root root   101 Nov  8  2012 .snapshot
-rwxr--r--   1  501 users     0 Nov  8  2012 .ddboost
drwxr-xr-x 103  501 users 5424 Nov 15  2012 PosteMobile
drwxr-xr-x 103  501 users 5424 Sep  4 10:30 manualdevice
drwxr-xr-x   2 root root   101 Sep  4 10:30 manualindex
networker:/DDSU # chown 501 manualindex/
networker:/DDSU # chgrp users manualindex/
networker:/DDSU #
networker:/DDSU #
networker:/DDSU # ls -lrt
total 12
drwxrwxrwx   2 root root   101 Nov  8  2012 .snapshot
-rwxr--r--   1  501 users     0 Nov  8  2012 .ddboost
drwxr-xr-x 103  501 users 5424 Nov 15  2012 PosteMobile
drwxr-xr-x 103  501 users 5424 Sep  4 10:30 manualdevice
drwxr-xr-x   2  501 users  101 Sep  4 10:30 manualindex
```

Following this, use the GUI to create a generic device as before, defining it as index pool.



Identity

Name: ddboost_index

Comment:

Configuration

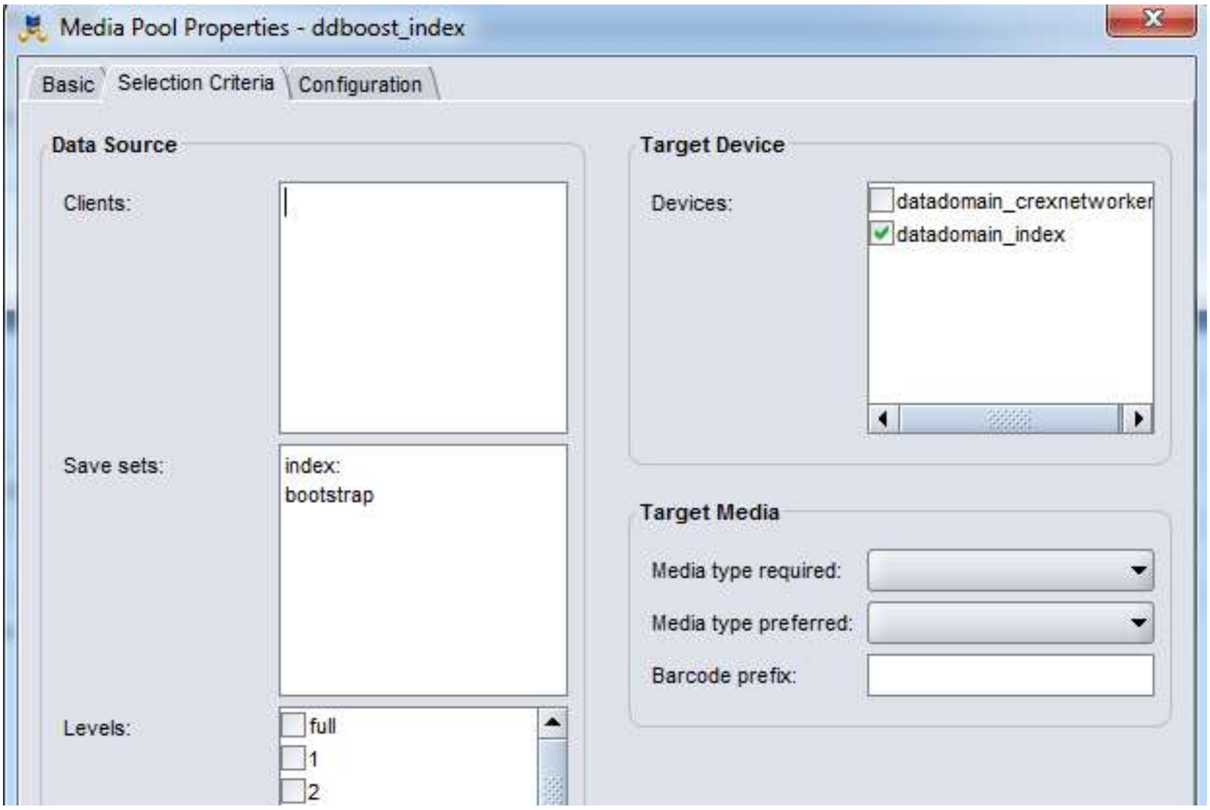
Enabled: ☑

Pool type: Backup

Label template: Indexed Archive

Data Source

Groups: □ DDboost
□ Default



Data Source

Clients:

Save sets: index:
bootstrap

Levels: □ full
□ 1
□ 2

Target Device

Devices: □ datadomain_crexnetworker
☑ datadomain_index

Target Media

Media type required:

Media type preferred:

Barcode prefix:

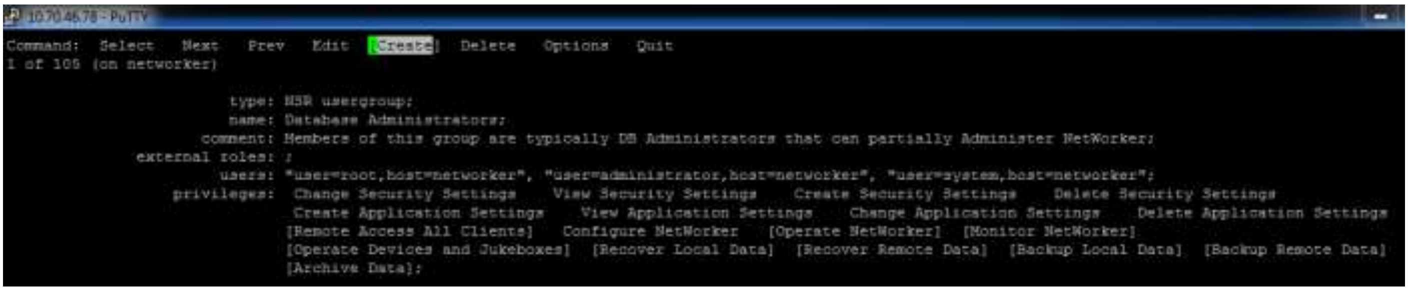
You are now able to back up data on the DD Boost device you have just created, indexing included!

Manual creation with nsradmin -visual

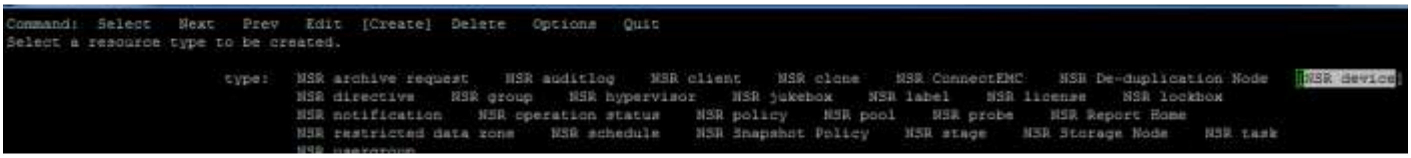
Another way to configure Boost device is via nsradmin command to be launched on NetWorker server. The procedure to follow is shown below:

Log on to NetWorker server and issue command: *nsradmin –visual*

You should land here where you can create a new device.

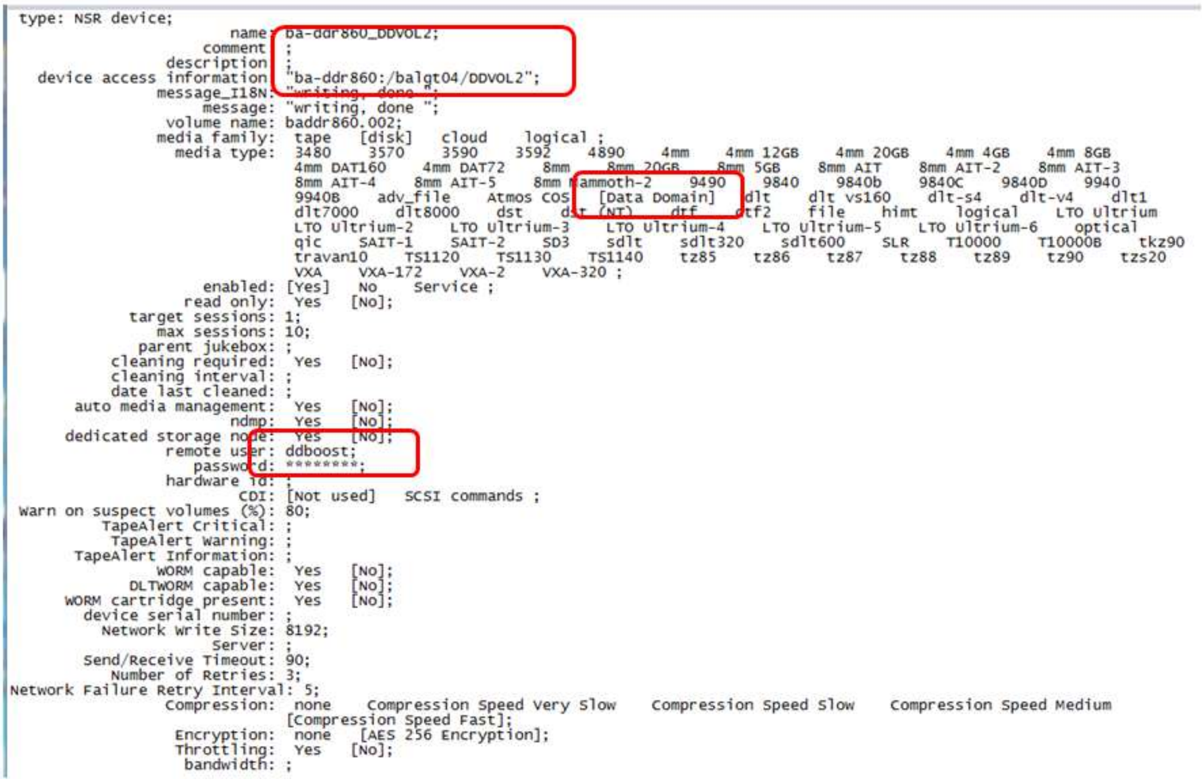


```
Command:  Select   Next   Prev   Edit  [Create]  Delete   Options   Quit
1 of 105 (on networker)

                 type: NSR usergroup;
                 name: Database Administrators;
              comment: Members of this group are typically DB Administrators that can partially Administer NetWorker;
       external roles: ;
                users: "user=root,host=networker", "user=administrator,host=networker", "user=system,host=networker";
           privileges:  Change Security Settings    View Security Settings    Create Security Settings    Delete Security Settings
                        Create Application Settings    View Application Settings    Change Application Settings    Delete Application Settings
                       [Remote Access All Clients]   Configure NetWorker   [Operate NetWorker]  [Monitor NetWorker]
                       [Operate Devices and Jukeboxes]  [Recover Local Data]  [Recover Remote Data]  [Backup Local Data]  [Backup Remote Data]
                       [Archive Data];
```



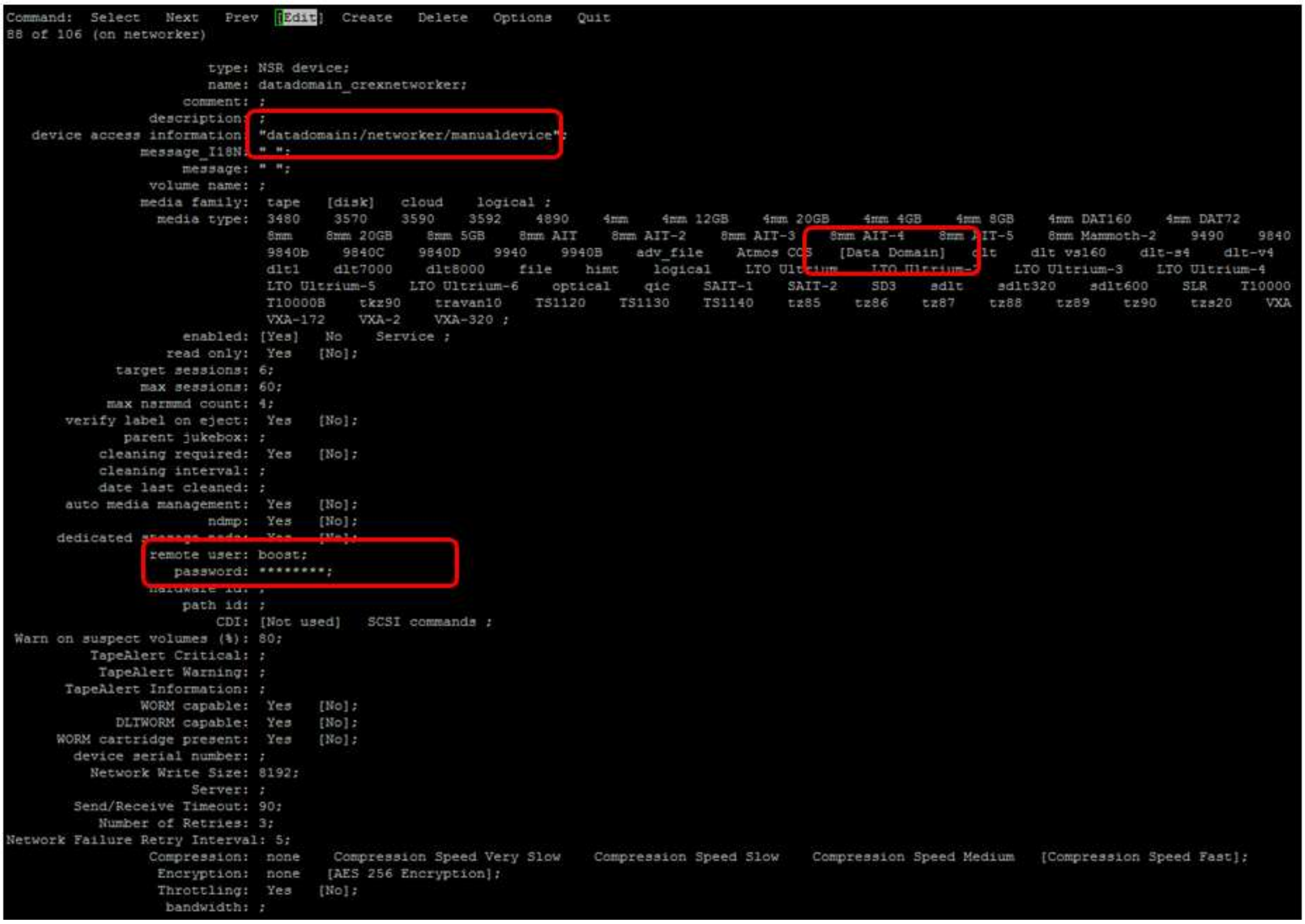
```
Command:  Select   Next   Prev   Edit  [Create]  Delete   Options   Quit
Select a resource type to be created.

                 type:   NSR archive request    NSR auditlog    NSR client    NSR clone    NSR ConnectEMC    NSR De-duplication Node   [NSR device]
                         NSR directive    NSR group    NSR hypervisor    NSR jukebox    NSR label    NSR license    NSR lockbox
                         NSR notification    NSR operation status    NSR policy    NSR pool    NSR probe    NSR Report Home
                         NSR restricted data zone    NSR schedule    NSR Snapshot Policy    NSR stage    NSR Storage Node    NSR task
                         NSR usergroup
```

What you find may differ slightly depending on NetWorker version, but the summary is the same. Shown here is what you see with version 7.6.x.



```
 type: NSR device;
                         name: ba-ddr860_DDVOL2;
                      comment: ;
                  description: ;
      device access information: "ba-ddr860:/balgt04/DDVOL2";
                 message_I18N: "writing, done ";
                      message: "writing, done ";
                  volume name: baddr860.002;
                 media family:  tape    [disk]    cloud    logical ;
                   media type:  3480     3570     3590     3592     4890     4mm     4mm 12GB     4mm 20GB     4mm 4GB     4mm 8GB
                                4mm DAT160     4mm DAT72     8mm     8mm 20GB     8mm 5GB     8mm AIT     8mm AIT-2     8mm AIT-3
                                8mm AIT-4     8mm AIT-5     8mm Mammoth-2     9490     9840     9840b     9840C     9840D     9940
                                9940B     adv_file     Atmos COS     [Data Domain]     dlt     dlt vs160     dlt-s4     dlt-v4     dlt1
                                dlt7000     dlt8000     dst     dst (NT)     dtf     dtf2     file     himt     logical     LTO Ultrium
                                LTO Ultrium-2     LTO Ultrium-3     LTO Ultrium-4     LTO Ultrium-5     LTO Ultrium-6     optical
                                qic     SAIT-1     SAIT-2     SD3     sdlt     sdlt320     sdlt600     SLR     T10000     T10000B     tkz90
                                travan10     TS1120     TS1130     TS1140     tz85     tz86     tz87     tz88     tz89     tz90     tzs20
                                VXA     VXA-172     VXA-2     VXA-320 ;
                      enabled: [Yes]    No    Service ;
                    read only:  Yes    [No];
              target sessions: 1;
                 max sessions: 10;
               parent jukebox: ;
            cleaning required:  Yes    [No];
            cleaning interval: ;
             date last cleaned: ;
       auto media management:  Yes    [No];
                         ndmp:  Yes    [No];
       dedicated storage node:  Yes    [No];
                  remote user: ddboost;
                     password: ********;
                  hardware id: ;
                          CDI: [Not used]    SCSI commands ;
 Warn on suspect volumes (%): 80;
           TapeAlert Critical: ;
            TapeAlert Warning: ;
        TapeAlert Information: ;
                 WORM capable:  Yes    [No];
              DLTWORM capable:  Yes    [No];
        WORM cartridge present:  Yes    [No];
         device serial number: ;
           Network Write Size: 8192;
                       Server: ;
        Send/Receive Timeout: 90;
            Number of Retries: 3;
Network Failure Retry Interval: 5;
                  Compression:  none    Compression Speed Very Slow    Compression Speed Slow    Compression Speed Medium
                               [Compression Speed Fast];
                   Encryption:  none    [AES 256 Encryption];
                   Throttling:  Yes    [No];
                    bandwidth: ;
```

Shown here is what you will see with version 8.0.x:



```
Command:  Select   Next   Prev  [Edit]  Create   Delete   Options   Quit
88 of 106 (on networker)

                        type: NSR device;
                        name: datadomain_crexnetworker;
                     comment: ;
                 description: ;
   device access information: "datadomain:/networker/manualdevice";
                message_I18N: " ";
                     message: " ";
                 volume name: ;
                media family:  tape   [disk]   cloud    logical ;
                  media type:  3480    3570    3590    3592    4890    4mm    4mm 12GB    4mm 20GB    4mm 4GB    4mm 8GB    4mm DAT160    4mm DAT72
                               8mm    8mm 20GB    8mm 5GB    8mm AIT    8mm AIT-2    8mm AIT-3    8mm AIT-4    8mm AIT-5    8mm Mammoth-2    9490    9840
                               9840b    9840C    9840D    9940    9940B    adv_file    Atmos COS    [Data Domain]    dlt    dlt vs160    dlt-s4    dlt-v4
                               dlt1    dlt7000    dlt8000    file    himt    logical    LTO Ultrium    LTO Ultrium-2    LTO Ultrium-3    LTO Ultrium-4
                               LTO Ultrium-5    LTO Ultrium-6    optical    qic    SAIT-1    SAIT-2    SD3    sdlt    sdlt320    sdlt600    SLR    T10000
                               T10000B    tkz90    travan10    TS1120    TS1130    TS1140    tz85    tz86    tz87    tz88    tz89    tz90    tzs20    VXA
                               VXA-172    VXA-2    VXA-320 ;
                     enabled: [Yes]   No    Service ;
                   read only:  Yes   [No];
             target sessions: 6;
                max sessions: 60;
             max nsrmmd count: 4;
       verify label on eject:  Yes   [No];
              parent jukebox: ;
           cleaning required:  Yes   [No];
           cleaning interval: ;
           date last cleaned: ;
       auto media management:  Yes   [No];
                        ndmp:  Yes   [No];
      dedicated storage node:  Yes   [No];
                 remote user: boost;
                    password: ********;
                 hardware id: ;
                     path id: ;
                         CDI: [Not used]   SCSI commands ;
 Warn on suspect volumes (%): 80;
          TapeAlert Critical: ;
           TapeAlert Warning: ;
       TapeAlert Information: ;
                WORM capable:  Yes   [No];
             DLTWORM capable:  Yes   [No];
      WORM cartridge present:  Yes   [No];
        device serial number: ;
          Network Write Size: 8192;
                      Server: ;
       Send/Receive Timeout: 90;
          Number of Retries: 3;
Network Failure Retry Interval: 5;
                 Compression:  none    Compression Speed Very Slow    Compression Speed Slow    Compression Speed Medium   [Compression Speed Fast];
                  Encryption:  none   [AES 256 Encryption];
                  Throttling:  Yes   [No];
                   bandwidth: ;
```

Parameters to consider are the same as before:

- Device access information

- Media type

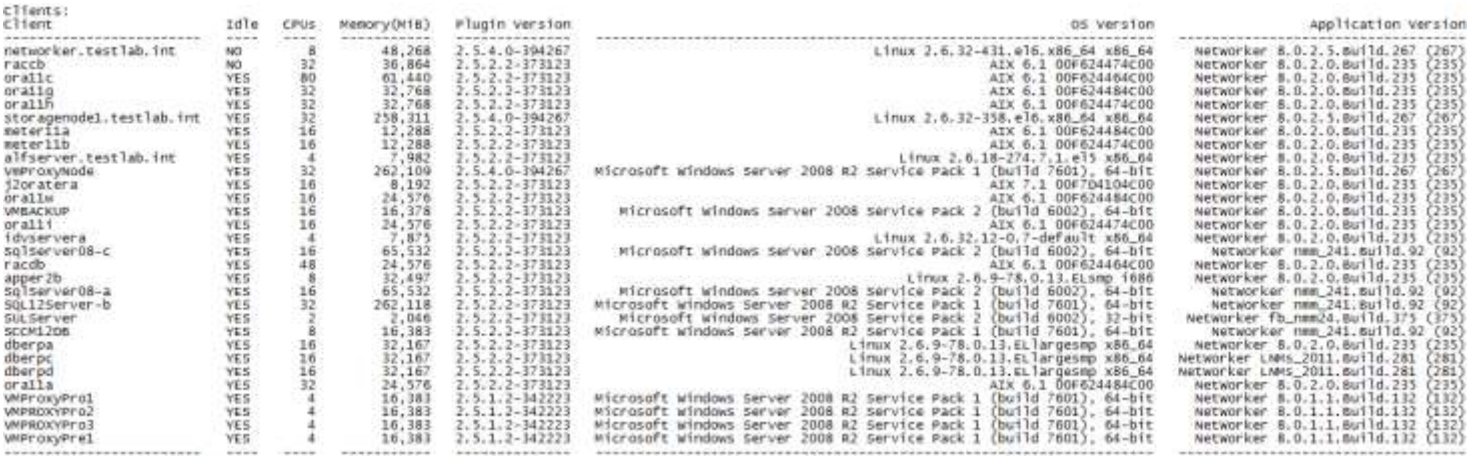
- Remote user (DD Boost one) and password

Procedure flows exactly the same as before with Data Domain folder creation, mounting and labelling newer device.

Data Flow control

There are different approaches to control that clients are sending data directly to Data Domain.

1. On the Data Domain side, the command: ***DD Boost show connection*** provides a list of the clients that are sending data directly to Data Domain via DD Boost protocol. An example is shown below:



```
Clients:
Client                    Idle  CPUs  Memory(MiB)  Plugin version                                                               OS Version                   Application Version
------------------------  ----  ----  -----------  --------------  ---------------------------------------------------------------------  -----------------------------------
networker.testlab.int     NO       8       48,268  2.5.4.0-394267                                     Linux 2.6.32-431.el6.x86_64 x86_64    Networker 8.0.2.5.Build.267 (267)
raccb                     NO      32       36,864  2.5.2.2-373123                                                   AIX 6.1 00F624474C00    Networker 8.0.2.0.Build.235 (235)
oral1c                    YES     80       61,440  2.5.2.2-373123                                                   AIX 6.1 00F624464C00    Networker 8.0.2.0.Build.235 (235)
oral1g                    YES     32       32,768  2.5.2.2-373123                                                   AIX 6.1 00F624484C00    Networker 8.0.2.0.Build.235 (235)
oral1h                    YES     32       32,768  2.5.2.2-373123                                                   AIX 6.1 00F624474C00    Networker 8.0.2.0.Build.235 (235)
storagenode1.testlab.int  YES     32      258,311  2.5.4.0-394267                                     Linux 2.6.32-358.el6.x86_64 x86_64    Networker 8.0.2.5.Build.267 (267)
meter11a                  YES     16       12,288  2.5.2.2-373123                                                   AIX 6.1 00F624484C00    Networker 8.0.2.0.Build.235 (235)
meter11b                  YES     16       12,288  2.5.2.2-373123                                                   AIX 6.1 00F624474C00    Networker 8.0.2.0.Build.235 (235)
alfserver.testlab.int     YES      4        7,982  2.5.2.2-373123                                        Linux 2.6.18-274.7.1.el5 x86_64    Networker 8.0.2.0.Build.235 (235)
VMProxyNode               YES     32      262,109  2.5.4.0-394267  Microsoft Windows Server 2008 R2 Service Pack 1 (build 7601), 64-bit    Networker 8.0.2.5.Build.267 (267)
j2oratera                 YES     16        8,192  2.5.2.2-373123                                                   AIX 7.1 00F7D4104C00    Networker 8.0.2.0.Build.235 (235)
oral1w                    YES     16       24,576  2.5.2.2-373123                                                   AIX 6.1 00F624484C00    Networker 8.0.2.0.Build.235 (235)
VMBACKUP                  YES     16       16,378  2.5.2.2-373123     Microsoft Windows Server 2008 Service Pack 2 (build 6002), 64-bit    Networker 8.0.2.0.Build.235 (235)
oral1i                    YES     16       24,576  2.5.2.2-373123                                                   AIX 6.1 00F624474C00    Networker 8.0.2.0.Build.235 (235)
idvservera                YES      4        7,875  2.5.2.2-373123                                     Linux 2.6.32.12-0.7-default x86_64    Networker 8.0.2.0.Build.235 (235)
sqlserver08-c             YES     16       65,532  2.5.2.2-373123     Microsoft Windows Server 2008 Service Pack 2 (build 6002), 64-bit      Networker nmm_241.Build.92 (92)
racdb                     YES     48       24,576  2.5.2.2-373123                                                   AIX 6.1 00F624464C00    Networker 8.0.2.0.Build.235 (235)
apper2b                   YES      8       32,497  2.5.2.2-373123                                         Linux 2.6.9-78.0.13.ELsmp i686    Networker 8.0.2.0.Build.235 (235)
sqlserver08-a             YES     16       65,532  2.5.2.2-373123     Microsoft Windows Server 2008 Service Pack 2 (build 6002), 64-bit      Networker nmm_241.Build.92 (92)
SQL12Server-b             YES     32      262,118  2.5.2.2-373123  Microsoft Windows Server 2008 R2 Service Pack 1 (build 7601), 64-bit      Networker nmm_241.Build.92 (92)
SULServer                 YES      2        2,046  2.5.2.2-373123     Microsoft Windows Server 2008 Service Pack 2 (build 6002), 32-bit   Networker fb_nmm24.Build.375 (375)
SCCM12DB                  YES      8       16,383  2.5.2.2-373123  Microsoft Windows Server 2008 R2 Service Pack 1 (build 7601), 64-bit      Networker nmm_241.Build.92 (92)
dberpa                    YES     16       32,167  2.5.2.2-373123                                  Linux 2.6.9-78.0.13.ELlargesmp x86_64    Networker 8.0.2.0.Build.235 (235)
dberpc                    YES     16       32,167  2.5.2.2-373123                                  Linux 2.6.9-78.0.13.ELlargesmp x86_64  Networker LNMS_2011.Build.281 (281)
dberpd                    YES     16       32,167  2.5.2.2-373123                                  Linux 2.6.9-78.0.13.ELlargesmp x86_64  Networker LNMS_2011.Build.281 (281)
oral1a                    YES     32       24,576  2.5.2.2-373123                                                   AIX 6.1 00F624484C00    Networker 8.0.2.0.Build.235 (235)
VMProxyPro1               YES      4       16,383  2.5.1.2-342223  Microsoft Windows Server 2008 R2 Service Pack 1 (build 7601), 64-bit    Networker 8.0.1.1.Build.132 (132)
VMPROXYPro2               YES      4       16,383  2.5.1.2-342223  Microsoft Windows Server 2008 R2 Service Pack 1 (build 7601), 64-bit    Networker 8.0.1.1.Build.132 (132)
VMPROXYPro3               YES      4       16,383  2.5.1.2-342223  Microsoft Windows Server 2008 R2 Service Pack 1 (build 7601), 64-bit    Networker 8.0.1.1.Build.132 (132)
VMProxyPre1               YES      4       16,383  2.5.1.2-342223  Microsoft Windows Server 2008 R2 Service Pack 1 (build 7601), 64-bit    Networker 8.0.1.1.Build.132 (132)
------------------------  ----  ----  -----------  --------------  ---------------------------------------------------------------------  -----------------------------------
```

2. On NetWorker Server, look at daemon.log which should display DDCL or direct client save with the client name you are backing up, An example is shown below.

```
[root@nynetworker logs]# nsr_render_log -achemp daemon.raw | grep "direct file save" | tail -50
10/16/2013 10:00:22 AM  1 3231405824 nsrmmd notice Save-set ID '207526885' (nyvba.nyclab.com:/data03) is using direct file save with Data Domain device 'nydatadomain_nynetworker'.
10/16/2013 10:00:55 AM  1 908764928 nsrmmd notice Save-set ID '190749719' (nynetworker:index:a1862bfb-00000004-51a92a76-51a92a75-00131e00-402cf2c9) is using direct file save with adv_file device 'Index'.
10/16/2013 06:30:36 PM  1 3231405824 nsrmmd notice Save-set ID '174003073' (nywin2012.nyclab.com:E:\) is using direct file save with Data Domain device 'nydatadomain_nynetworker'.
10/16/2013 06:30:44 PM  1 3231405824 nsrmmd notice Save-set ID '157225872' (nywin2012.nyclab.com:F:\) is using direct file save with Data Domain device 'nydatadomain_nynetworker'.
10/16/2013 06:30:51 PM  1 908764928 nsrmmd notice Save-set ID '140448666' (nynetworker:index:a4422e35-00000004-5213c63b-5213c63a-01341e00-402cf2c9) is using direct file save with adv_file device 'Index'.
10/16/2013 07:05:46 PM  1 3231405824 nsrmmd notice Save-set ID '123673224' (nywin2012.nyclab.com:E:\) is using direct file save with Data Domain device 'nydatadomain_nynetworker'.
```

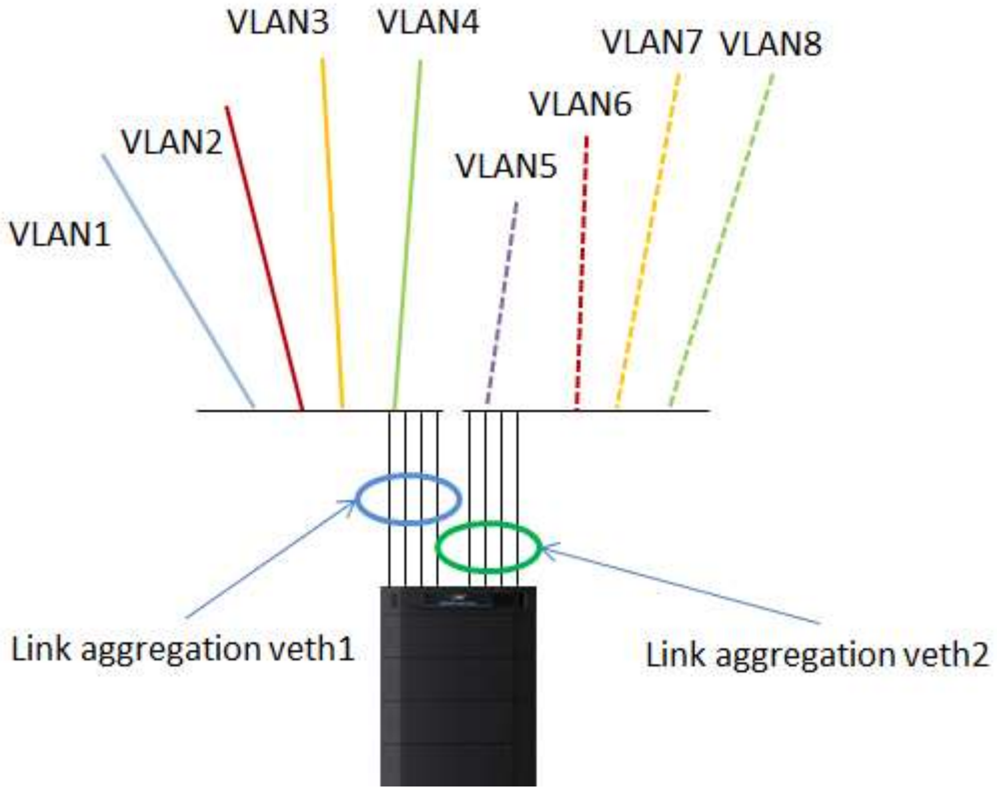
Data Domain Networking possibilities

Data Domain offers different methods to configure networking:

- Link aggregation control Protocol
- VLANs and VLAN Tagging
- IP aliasing

The most appropriate configuration is to use link aggregation (e.g. 2x10Gbit or 8x1Gbit) and, on the created channels, configure all IP addresses you need to go on different network VLANs.

The figure below shows an example of 8 Gbit connections aggregated into two different aggregations where it is possible to create VLANs you need for configuration.



It is not possible to use DD Boost ifgroup feature in this type of configuration.

Configuring Link Aggregation Control Protocol

Link Aggregation Protocol (LACP) is a bonding protocol used to provide active coordination or link status via “heartbeat” messages and failure handling of links that experience issues. LACP is limited to configuration on a single switch.

When setting up a LACP link, several parameters can be set that affect the performance of the bond; slow / fast timers, up / down timers, and the hash algorithm used.

- The slow / fast option determines how often a heartbeat message is sent to determine link status. The default for slow is 30 seconds. Fast changes the option to one second.
- The up / down option determines how long LACP will wait before reacting to a link state change. The value is set in milliseconds and in intervals of .9 seconds.
- The hash algorithm determines which variables will be used to perform link selection.

Note: It is very important to select the proper load balancing algorithm as the differing options can significantly affect data transfer rates and thereby, performance.

The *net aggregate* command creates a virtual interface with the specified physical interfaces and uses one of three aggregation modes. Select the mode that is compatible with the switch:

- xor-L2

Xor Layer 2 transmits packets based on static balanced mode aggregation with a Xor hash of Layer 2 (inbound and outbound MAC addresses).

- xor-L3L4

Xor Layer 3/4 transmits packets based on static balanced mode aggregation with a Xor hash of Layer 3 (inbound and outbound IP address) and Layer 4 (inbound and outbound port numbers).

1. [ ] Disable each of the interfaces that you plan to use as aggregation interfaces, such as *eth2a* and *eth3*, by entering:

```
# net disable eth3a
# net disable eth3b
```

2. [ ] To create a virtual interface, use the *net create* command and choose a virtual interface. For example:

```
# net create virtual veth1
```

3. [ ] Enter:

```
# net aggregate add <virtual-ifname> mode {roundrobin | lacp hash {xor-L2 |xor-
L2L3 | xorL3L4}interfaces physical-ifnamelist
```

For example, to create a virtual interface *veth1* from the two physical interfaces *eth3a* and *eth3b*, using the mode xor-L2, enter:

```
# net aggregate add veth1 mode lacp hash xor-L2 interfaces eth3a eth3b
```

Note: Optional command variables can be added to this command, as needed by network conditions. Those options are rate {fast / slow}, up / down {time}

---

4. [ ] To verify that the interface has been created, enter:

```
# net aggregate show
```

The output displays the name of the virtual interface, its hardware address, aggregation mode, and the ports that comprise the virtual interface.

Note: Do not assign an IP address at the time of creating VLAN interfaces on an aggregated interface. Assign an IP address at the time of VLAN configuration.

---

5. [ ] Assign an IP address to the new interface using this command:

```
# net config ifname ipaddr
```

where ifname is the name of the interface, which is veth1 in this example, and ipaddr is the interface's IP address.

Routing rules

To work properly, correct routing rules must be set on Data Domain to keep data traffic in/out on the same network to prevent asymmetric traffic blocked by default on the firewalls.

Below is an example of the configuration of originating box:

```
# net show settings
   port    enabled   DHCP   IP address      netmask         additional setting
   -----   -------   ----   -------------   -------------   ------------------
   eth0    yes       no     10.25.203.202   255.255.252.0
   eth1    yes       no     172.28.0.209    255.255.0.0
   eth2    yes       no     172.28.0.211    255.255.0.0
   veth0   no        n/a    n/a             n/a
   veth1   no        n/a    n/a             n/a
   veth2   no        n/a    n/a             n/a
   veth3   no        n/a    n/a             n/a
   -----   -------   ----   -------------   -------------   ------------------
```

This is the routing table before any changes have been made

```
# route show table
   Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
   10.25.200.0     0.0.0.0         255.255.252.0   U     0      0        0 eth0
   172.28.0.0      0.0.0.0         255.255.0.0     U     0      0        0 eth1
   172.28.0.0      0.0.0.0         255.255.0.0     U     0      0        0 eth2
   127.0.0.0       0.0.0.0         255.0.0.0       U     0      0        0 lo
   0.0.0.0         10.25.200.1     0.0.0.0         UG    0      0        0 eth0
```

In this example, we will be adding a route to 172.28.0.207 and want traffic bound for that address to egress Ethernet interface 2.

Use either of the following commands to add a route where a host uses a specific interface (please note the hostname is resolved before being inserted into the routing table):

```
# route add 172.28.0.207 eth2
```

Add a route to change the egress location:

```
# route add -host bob.chew.net eth2
# route show table
   Kernel IP routing table
   Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
   192.168.1.10    0.0.0.0         255.255.255.255 UH    0      0        0 eth2
   10.25.200.0     0.0.0.0         255.255.252.0   U     0      0        0 eth0
   172.28.0.0      0.0.0.0         255.255.0.0     U     0      0        0 eth1
   172.28.0.0      0.0.0.0         255.255.0.0     U     0      0        0 eth2
   127.0.0.0       0.0.0.0         255.0.0.0       U     0      0        0 lo
   0.0.0.0         10.25.200.1     0.0.0.0         UG    0      0        0 eth0
```

Configuring Link Aggregation (Active Only)

Note: It is important to select the proper load balancing algorithm as the differing options can significantly affect data transfer rates and thereby, performance.

The *`net aggregate`* command creates a virtual interface with the specified physical interfaces and uses one of three aggregation modes. Select the mode that is compatible with the switch:

- roundrobin

Roundrobin transmits packets in sequential order, from the first available link through the last in the aggregated group.

- xor-L2

Xor Layer 2 transmits packets based on static balanced mode aggregation with a XOR hash of Layer 2 (inbound and outbound MAC addresses).

- xor-L2L3

Xor Layer 2 transmits packets based on static balanced mode aggregation with a XOR hash of Layer 2 (inbound and outbound MAC addresses) and with a XOR hash of Layer 3 (inbound and outbound IP address).

- xor-L3L4

Xor Layer 3/4 transmits packets based on static balanced mode aggregation with a XOR hash of Layer 3 (inbound and outbound IP address) and Layer 4 (inbound and outbound port numbers).

1. [ ] Disable each of the interfaces that you plan to use as aggregation interfaces, such as `eth3a` and `eth3b`, by entering:

```
# net disable eth3a
# net disable eth3b
```

2. [ ] Enter:

```
# net aggregate add <virtual-ifname> mode {roundrobin | balanced hash {xor-L2
|xor-L2L3 | xorL3L4}interfaces physical-ifnamelist
```

For example, to create a virtual interface *veth1* from the two physical interfaces *eth3a* and *eth3b*, using the mode xor-L2, enter:

```
# net aggregate add veth1 mode xor-L2 interfaces eth3a eth3b
```

Note: Optional command variables can be added to this command, as needed by network conditions. Those options are rate {fast / slow}, up / down {time}

3. [ ] To verify that the interface has been created, enter:

```
# net aggregate show
```

The output displays the name of the virtual interface, its hardware address, aggregation mode, and the ports that comprise the virtual interface.

Note: Do not assign an IP address at the time of creating VLAN interfaces on an aggregated interface. Assign an IP address at the time of VLAN configuration.

4. [ ] Assign an IP address to the new interface using this command:

```
# net config ifname ipaddr
```

where *ifname* is the name of the interface, which is *veth1* in this example, and *ipaddr* is the interface’s IP address.

```
# net config veth1 192.168.2.78
```

VLANs and VLAN Tagging (802.1Q)

After creation of aggregations, you can define different IP addresses using VLANs and VLAN tagging mechanism.

VLANs are virtual networks, each with an independent IP subnet and broadcast domain.

VLAN Tagging is a networking method used for sharing a physical Ethernet network link by multiple independent logical networks (i.e. the same physical interface can be used for multiple VLANs). If multiple VLANs are set up on a single interface, tagging will automatically occur.

To configure a VLAN (and its associated tagging):

1. [ ] Configure the switch port that connects to the interface to receive and send VLAN traffic from the Data Domain interface. See the switch documentation for details on the configuration.

2. [ ] On the Data Domain system, enable the interface that you plan to use as the VLAN interface, such as *eth5b*, by entering:

   ```
   # net config eth5b up
   ```

3. [ ] Create the VLAN interface using either a physical port or a configured virtual port. The range for vlan-id is between 1 and 4098:

   ```
   # net create interface { physical-ifname | virtual-ifname } vlan vlan-id
   ```

   For example, to create a VLAN interface on physical interface *eth5b*, enter:

   ```
   # net create interface eth5b vlan 1
   ```

   A VLAN interface named *eth5b.1* is created.

4. [ ] Assign an IP address and netmask to the new interface using this command:

   ```
   # net config ifname ipaddr netmask mask
   ```

   where *ifname* is the name of the interface, which is *eth2b.1* in this example, *ipaddr* is the interface’s IP address, and mask is the corresponding netmask.

Note: DHCP cannot be used to assign an IP address to a VLAN.

5. [ ] To verify that the interface has been created, enter:

   ```
   # net show settings
      port     enabled DHCP IP address     netmask       additional setting
      -------  ------- ---- -------------- ------------- -------------
      eth5b.1  yes     no   192.168.11.156 255.255.252.0
   ```

   The abridged output above shows the name of the VLAN interface (eth5b.1), whether or not it is enabled.

IP Aliasing

IP aliasing is a configuration option that allows more than one address to be placed on a single interface. Physical or Virtual interfaces can accept an IP alias.

To create an IP alias (where basename is the physical interface and alias in the number of the alias):

```
# net create interface <basename> alias <number>
```

To configure an IP alias:

```
# net config eth0b:59 192.168.5.97 netmask 255.255.255.252
```

To destroy an IP alias on a configured interface:

```
# net destroy eth0b:159
```

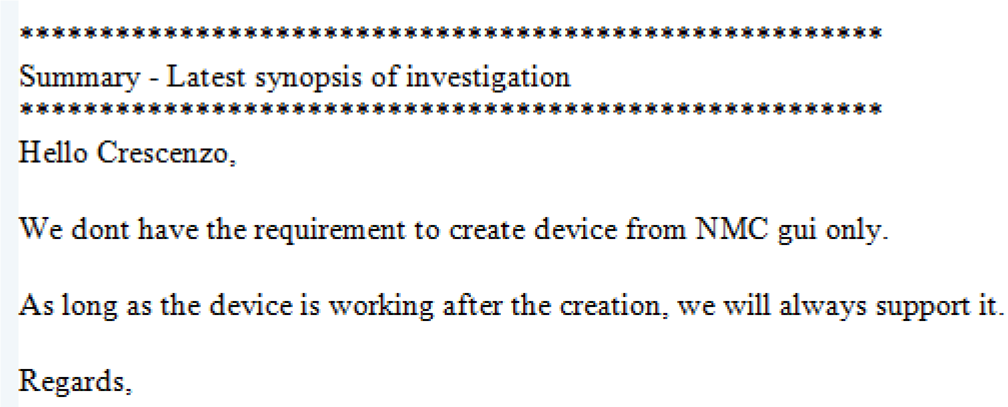
To add an IP alias to an already configured interface:

```
# net config <basename>
```

Support Statement

The procedure described is supported by EMC. There is no requirement about method used to create a device; if it works, it is supported: service request number: 56610904

The following is the Support answer.



****

Summary - Latest synopsis of investigation

****

Hello Crescenzo,

We dont have the requirement to create device from NMC gui only.

As long as the device is working after the creation, we will always support it.

Regards,