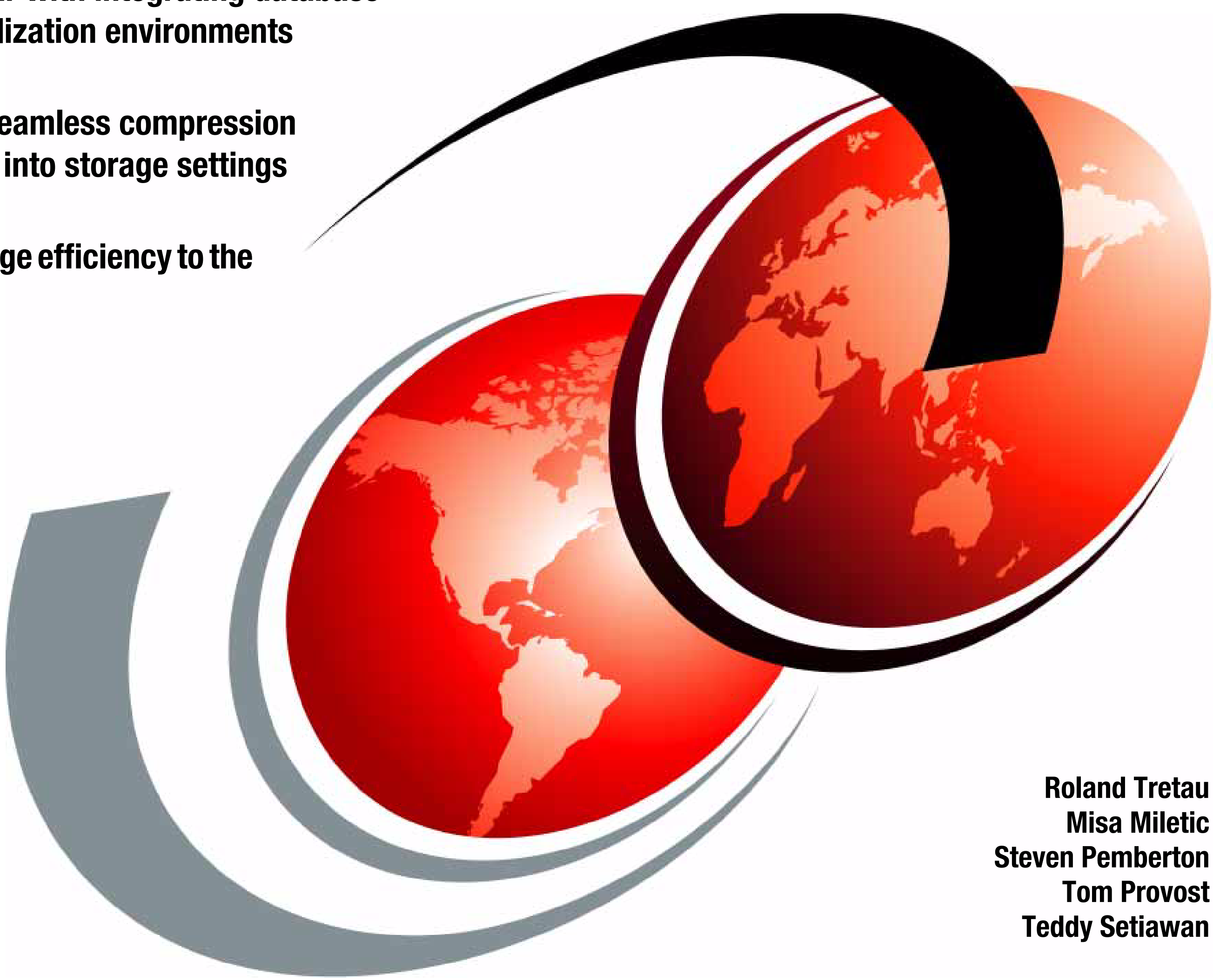
IBM

IBM Real Time Compression Appliance Application Integration Guide

Get familiar with integrating database and virtualization environments

See how seamless compression integrates into storage settings

Take storage efficiency to the next level



Roland Tretau
Misa Miletic
Steven Pemberton
Tom Provost
Teddy Setiawan

Redbooks

ibm.com/redbooks

International Technical Support Organization

Real Time Compression Appliance Application Integration Guide

July 2012

SG24-8029-00

Note: Before using this information and the product it supports, read the information in “Notices” on page xv.

First Edition (July 2012)

This edition applies to the IBM Real-time Compression Appliance Release 3.8 from October 2011.

Contents

Figures

Tables

Examples

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AFS®
AIX®
IBM®
Power®
ProtecTIER®
Redbooks®
Redbooks (logo) ®
System Storage®
Tivoli®
xSeries®

The following terms are trademarks of other companies:

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Snapshot, RAID-DP, Network Appliance, WAFL, SyncMirror, SnapVault, MultiStore, Data ONTAP, NetApp, and the NetApp logo are trademarks or registered trademarks of NetApp, Inc. in the U.S. and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Preface

Continuing its commitment to developing and delivering industry-leading storage technologies, IBM® is introducing the IBM Real-time Compression Appliances for NAS, an innovative new storage offering that delivers essential storage efficiency technologies, combined with exceptional ease of use and performance.

In an era when the amount of information, particularly in unstructured files, is exploding, but budgets for storing that information are stagnant, IBM Real-time Compression technology offers a powerful tool for better information management, protection, and access. IBM Real-time Compression can help slow the growth of storage acquisition, reducing storage costs while simplifying both operations and management. It also enables organizations to keep more data available for use rather than storing it offsite or on harder-to-access tape, so they can support improved analytics and decision making.

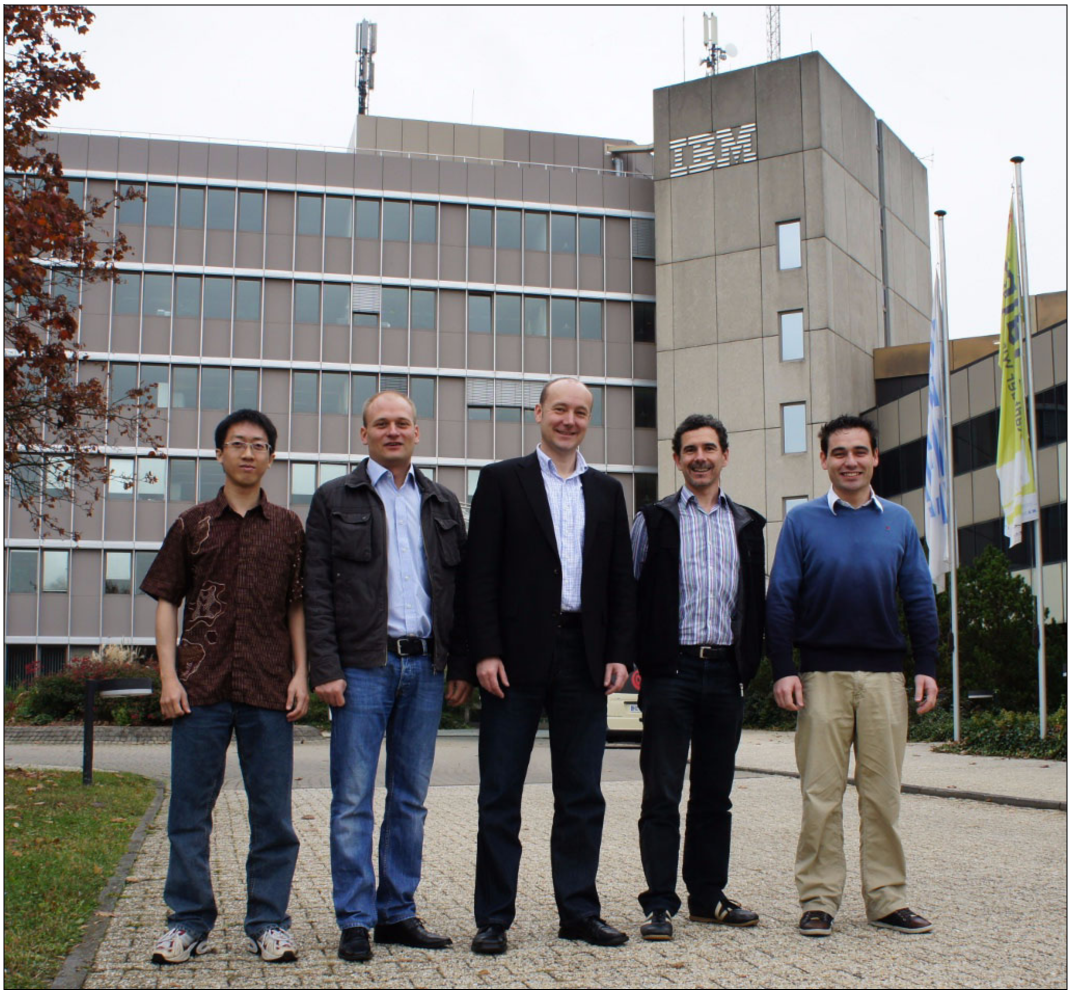
IBM Real-time Compression Appliances provide online storage optimization through real-time data compression, delivering dramatic cost reduction without performance degradation.

This IBM Redbooks® publication is an easy-to-follow guide that describes how to design solutions successfully using IBM Real-time Compression Appliances (IBM RTCAs). It explains best practices for RTCA solution design, application integration, and practical RTCA use cases. This is a companion book to *Introduction to IBM Real-time Compression Appliances*, SG24-7953. The publication is located at:

http://www.redbooks.ibm.com/abstracts/sg247953.html?Open

The team who wrote this book

This book was produced by a team of specialists from around the world working at the IBM European Storage Competence Center (ESCC) located in Mainz, Germany in close cooperation with the International Technical Support Organization (ITSO), San Jose, California, USA.



From left: Teddy Setiawan, Misa Miletic, Roland Tretau, Steven Pemberton, Tom Provost

Roland Tretau is an Information Systems professional with over 15 years of experience in the Information Technology (IT) industry. He has Engineering and Business Masters degrees and is the author of many storage-related IBM Redbooks publications. Roland has a solid background in project management, consulting, operating systems, storage solutions, enterprise search technologies, and data management.

Misa Miletic is a Client Technical Specialist for IBM Systems and Technology Group in Serbia. He supports business partners and customers from Serbia, Macedonia, Montenegro and Albania. Misa has more than seven years of expertise in the IT professional field, site implementations, customer support, solution design, and consulting end education. He joined IBM in 2008. His areas of expertise include designing storage area networks, storage and x86 servers, virtualization, backup solutions, planning disaster recovery, and high-availability solutions, mostly in complex environments. He has a Masters degree in Electrical Engineering from Belgrade University in Serbia.

Steven Pemberton is the Principal Engineer for storage with a large IBM GTS-managed account in Melbourne, Australia. Steven has broad experience as an IT solution architect, pre-sales specialist, consultant, instructor, and enterprise IT customer. Steven is the co-author of three previous IBM Redbooks publications.

Tom Provost is a Field Technical Sales Specialist for the IBM Systems and Technology Group in Belgium. Tom has three years of experience as an IT professional providing design, implementation, migration, and troubleshooting support for IBM System x, IBM System Storage®, storage software, and virtualization. He joined IBM in 2010.

Teddy Setiawan is an IT Specialist for IBM Global Technology Services in Indonesia. Teddy has five years of experience in technical services providing assessment, design, implementation, migration, and troubleshooting support for IBM Power® system, IBM AIX®, Linux, high availability, virtualization, IBM System Storage, and storage software.

Thanks to the following people for their contributions to this project:

Eyal Traitel
Manager, Development Support, IBM Real-time Compression
IBM Israel

Christian Prediger Appel
IBM Brazil

Jochen Erb
IBM Germany

Uwe Heinrich Mueller, Uwe Schweikhard
IBM Germany

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Obtain more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  ibm.com/redbooks

- Send your comments in an email to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/IBMRedbooks

- Follow us on Twitter:

  http://twitter.com/ibmredbooks

- Look for us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806

- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm

- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

Part 1

Solution design

In this part of the book, we provide the best practices for implementing the IBM Real-time Compression Appliance in several application environments. Based on these best practices you will be able to implement the IBM Real-time compression with as little impact as possible.

This is a companion book to *Introduction to IBM Real-time Compression Appliances*, SG24-7953, which is located at:

http://www.redbooks.ibm.com/abstracts/sg247953.html?Open

1

NAS / N series solution design

In this chapter, we explain some common architectures and the corresponding configurations of virtual network interfaces. Furthermore, we discuss typical Network Attached Storage (NAS) and N series architectures in combination with IBM Real-time Compression Appliance (RTCA) solutions.

Important: This chapter uses IBM System Storage N series as NAS systems. The different solution and design approaches can be easily applied to other NAS solutions that are available in the marketplace.

1.1 Introduction to the design examples

Figure 1-1 and Figure 1-2 serve as an introduction to facilitate an understanding of the design examples in this chapter. In these diagrams, we show the port assignment of the RTCA product (STN6500) and the N series (N5000).

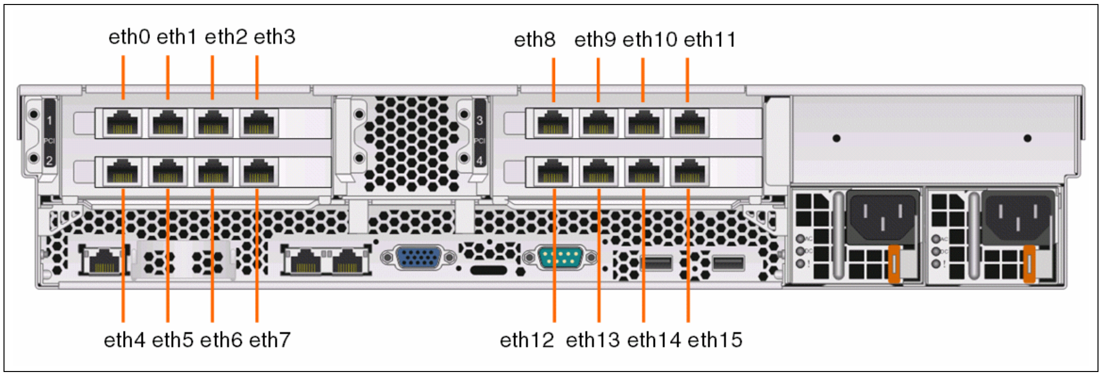


Figure 1-1 STN6500 port assignment

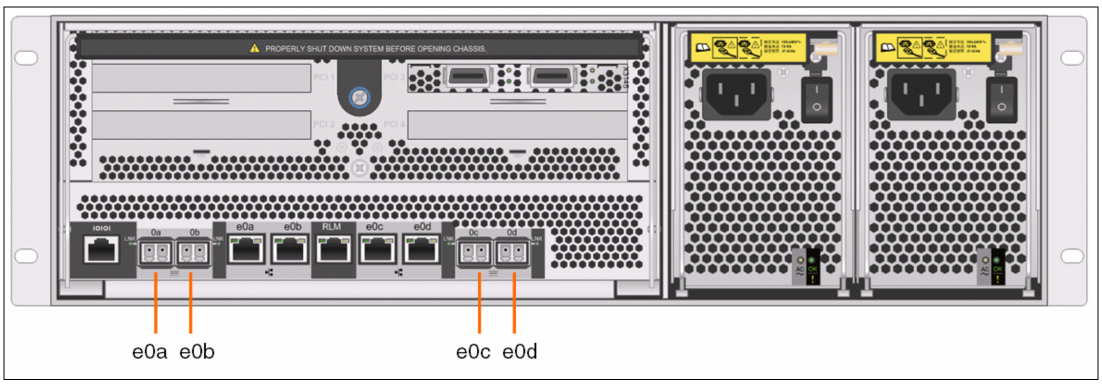


Figure 1-2 N series port assignment

1.2 N series single node solutions

The main idea for how to create a whole design solution and more possible ways for implementation is described in the companion IBM Redbooks publication *Introduction to IBM Real-time Compression Appliances*, SG24-7953, located at:

http://www.redbooks.ibm.com/abstracts/sg247953.html?Open

This section focuses on solutions using a single node N series controller.

1.2.1 N series and single RTCA product

The most basic configuration consists of a single NAS storage controller connected to a single RTCA product.

In Figure 1-3 on page 5, an N series single-node system is connected by a single link (from onboard port e0a) to the RTCA eth0 port, and eth1 is connected to the network switch. The ports eth0 and eth1 are configured as a bridge (STN-1 BR-0), with a bridge IP address in the same subnet as e0a.

Interfaces: There can be more than a single interface connected between the NAS storage controller and the RTCA product. However, in this configuration, the interfaces are independent of each other, and are not aggregated for load balancing or redundancy.

Being the simplest configuration possible, it does not provide any redundancy and is therefore not desirable for production use. Any failure in the network path, be it the network switch, its port, any of the network cables, the compression appliance ports, or the N series ports, will impact service to users.

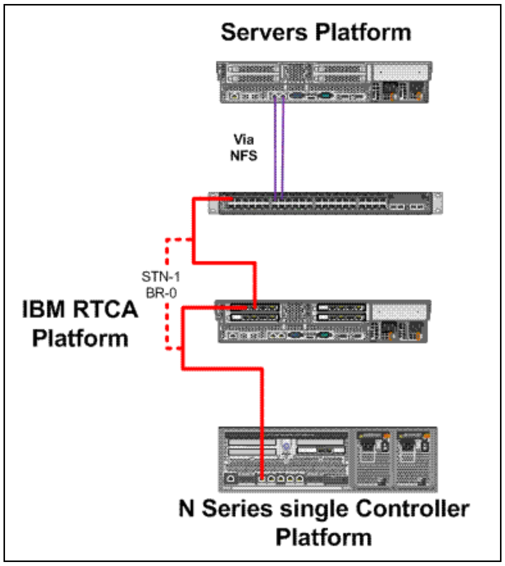


Figure 1-3 N series single node connected to a single RTCA product

The port connectivity that belongs to the solution in Figure 1-3 is illustrated in Table 1-1.

Table 1-1 N series single node connected to a single RTCA product - port connectivity

Switch/ port	RTCA/ port	Bond	Bridge	Bond	RTCA/ port	NAS port
SW1/ 9	STN1/ eth1	n/a	STN-1 BR-0	n/a	STN1/ eth0	NAS1 e0a

1.2.2 N series and single RTCA product, active/passive path

Similar to the configuration in the previous section, a single NAS controller is connected to a single RTCA product in Figure 1-4 on page 6. However, an additional link is added to provide redundancy to the networking path. The two links are aggregated into a single group. The N series single-node system is configured with an active/passive configuration, a single-mode virtual interface (VIF).

The compression appliance is configured with two active/passive bonds and a bridge that connects them. On the network switch side, the two ports are configured with the same parameters, and most importantly, both must be in the same subnet.

In this configuration, only one link is active at any particular time. If the NAS storage controller detects a fault in the active link, the standby link is activated and all traffic goes through the newly activated link.

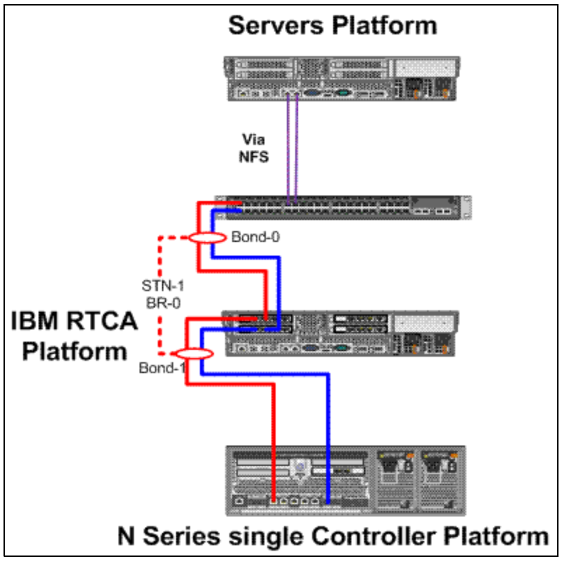


Figure 1-4 N series single-node, a single RTCA product, active/passive path failover

The port connectivity that belongs to the solution in Figure 1-4 is illustrated in Table 1-2.

Table 1-2 N series single-node, single RTCA product, active/passive path failover port connectivity

Switch/ port	RTCA/ port	Bond	Bridge	Bond	RTCA/ port	NAS port/ vif
SW1/ 9	STN1/ eth1	bond0 active/ passive	STN-1 BR-0	bond1 active/ passive	STN1/ eth0	NAS1 e0a/ vif-s
SW1/ 13	STN1/ eth5	bond0 active/ passive		bond1 active/ passive	STN1/ eth4	NAS1 e0c/ vif-s

The N series virtual interface configuration is illustrated in Example 1-1.

Example 1-1 Active/passive path failover for single systems: N series VIF configuration

```
NAS1> vif status
default: transmit 'IP Load balancing', VIF Type 'multi_mode', fail 'log'
vif-s: 1 link, transmit 'none', VIF Type 'single_mode' fail 'default'
     VIF Status    Up     Addr_set
    up:
        e0a: state up, since 31Mar2011 18:43:52 (00:00:23)
                mediatype: auto-1000t-fd-up
                flags: enabled
                input packets 13, input bytes 1062
                output packets 4, output bytes 168
                output probe packets 4, input probe packets 0
                strike count: 4 of 10
                up indications 1, broken indications 0
                drops (if) 0, drops (link) 0
                indication: up at 31Mar2011 18:43:52
```

```
                    consecutive 22, transitions 1
    down:
        e0c: state down, since 31Mar2011 18:43:52 (00:00:23)
                mediatype: auto-1000t-fd-up
                flags: enabled
                input packets 0, input bytes 0
                output packets 13, output bytes 570
                output probe packets 4, input probe packets 0
                strike count: 4 of 10
                up indications 1, broken indications 0
                drops (if) 0, drops (link) 0
                indication: up at 31Mar2011 18:43:52
                        consecutive 22, transitions 1
```

1.2.3 N series and active/passive path, RTCA product HA pair

By building on the previous examples and adding a compression appliance to the configuration, a single NAS storage controller is connected to two RTCA, STN-1 and STN-2 products in Figure 1-5. The N series single-node system is configured with an active/passive configuration, a single-mode virtual interface (VIF). A primary link is connected to one RTCA STN-1 product, and the other, the passive link, is connected to the second RTCA STN-2 product. The RTCA products are connected with a single link each to the switch.

This setup adds redundancy to the RTCA product itself. If any problem occurs, or maintenance is required, the service can fail over from the active RTCA STN-2 product to the standby RTCA STN-1 product. It is possible, and actually desirable, to connect each RTCA product to a separate network switch.

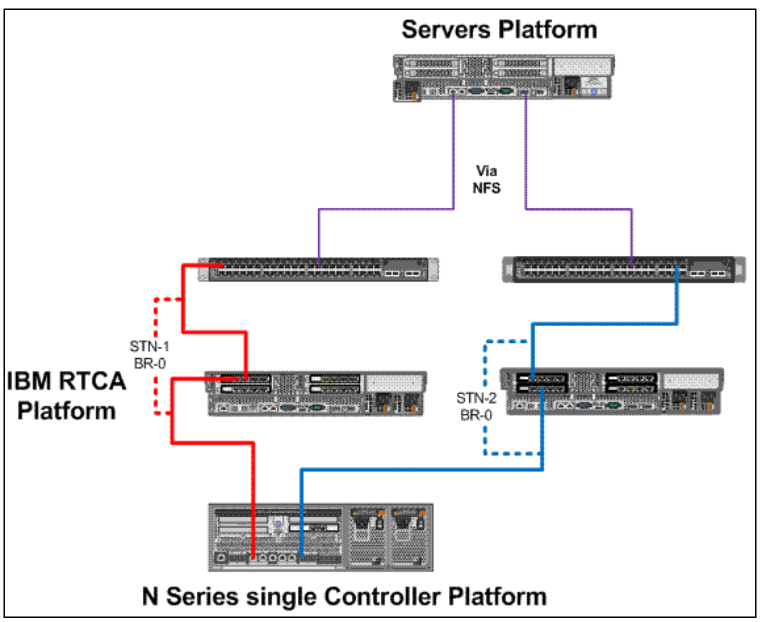


Figure 1-5 N series single-node, active/passive path failover to HA pair

The port connectivity that belongs to the solution in Figure 1-5 on page 7 is illustrated in Table 1-3.

Table 1-3 N series single-node, active/passive path failover to HA pair port connectivity

Switch/ port	RTCA/ port	Bond	Bridge	Bond	RTCA/ port	NAS port/ VIF
SW1/ 9	STN1/ eth1	n/a	STN-1 BR-0	n/a	STN1/ eth0	NAS1 e0a/ vif-s
SW2/ 9	STN2/ eth1	n/a	STN-2 BR-0	n/a	STN2/ eth0	NAS1 e0c/ vif-s

The N series virtual interface configuration is illustrated in Example 1-2.

Example 1-2 Active/passive path failover to RTCA product HA pair: N series VIF configuration

```
NAS1> vif status
default: transmit 'IP Load balancing', VIF Type 'multi_mode', fail 'log'
vif-s: 1 link, transmit 'none', VIF Type 'single_mode' fail 'default'
     VIF Status    Up     Addr_set
    up:
        e0a: state up, since 31Mar2011 18:43:52 (00:00:23)
                mediatype: auto-1000t-fd-up
                flags: enabled
                input packets 13, input bytes 1062
                output packets 4, output bytes 168
                output probe packets 4, input probe packets 0
                strike count: 4 of 10
                up indications 1, broken indications 0
                drops (if) 0, drops (link) 0
                indication: up at 31Mar2011 18:43:52
                        consecutive 22, transitions 1
    down:
        e0c: state down, since 31Mar2011 18:43:52 (00:00:23)
                mediatype: auto-1000t-fd-up
                flags: enabled
                input packets 0, input bytes 0
                output packets 13, output bytes 570
                output probe packets 4, input probe packets 0
                strike count: 4 of 10
                up indications 1, broken indications 0
                drops (if) 0, drops (link) 0
                indication: up at 31Mar2011 18:43:52
                        consecutive 22, transitions 1
```

1.2.4 Active/passive path, RTCA product HA, EtherChannel/LACP bonds

The most common use of active/passive bonds is when they are combined with EtherChannel or LACP bonds. This provides the best combination of performance and high availability. In Figure 1-6 on page 9, the N series single-node system is connected with two or more interfaces to the RTCA product, configured as a single port aggregate (virtual interface), either an EtherChannel or 802.3ad (LACP) bond type.

Tip: Make sure that all active links of the same link aggregation (LACP or EtherChannel) are connected to the same RTCA product.

Similarly, a corresponding bond in the RTCA STN-2 product is connected to the network switch. An additional bond is connected to a standby RTCA STN-1 product in the same manner. Its corresponding bond is connected to the network switch, be it the same one or another switch (Figure 1-6).

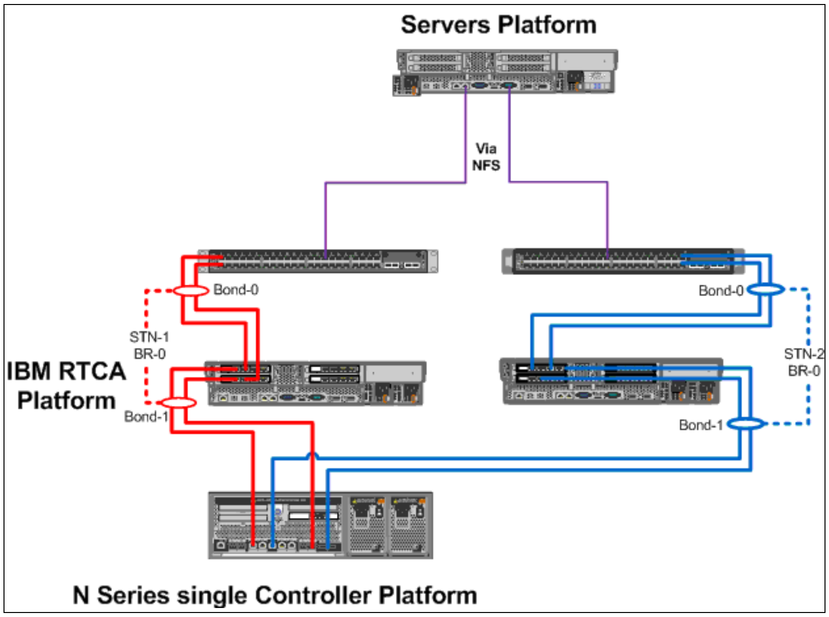


Figure 1-6 N series single-node, active/passive path failover to HA pair, EtherChannel or LACP bonds

The port connectivity that belongs to the solution in Figure 1-6 is illustrated in Table 1-4.

Table 1-4 N series single-node with EtherChannel or LACP bonds

Switch/ port	RTCA/ port	Bond	Bridge	Bond	RTCA/ port	NAS port/ vif
SW1/ 9	STN1/ eth1	bond0 LACP	STN-1 BR-0	bond1 LACP	STN1/ eth0	NAS1 e0a/ vifa
SW1/ 13	STN1/ eth5	bond0 LACP		bond1 LACP	STN1/ eth4	NAS1 e0c/ vifa
SW2/ 35	STN2/ eth9	bond0 LACP	STN-2 BR-0	bond1 LACP	STN2/ eth8	NAS1 e0b/ vifb
SW2/ 41	STN2/ eth13	bond0 LACP		bond1 LACP	STN2/ eth12	NAS1 e0d/ vifb

The virtual interfaces *vifa* and *vifb* are configured as dynamic multi-mode vif and are grouped by creating a second layer of vif called *vif1* as a single-mode vif. Either *vifa* or *vifb* is active.

In this way, data is transmitted through one RTCA product only and not at any time through both concurrently. The necessary configuration entries of the /etc/rc file are shown in Example 1-3 on page 10.

Example 1-3 N series or LACP port aggregation configuration entry in /etc/rc

```
vif create lacp vifa -b ip e0a e0c
vif create lacp vifb -b ip e0b e0d
vif create single vif1 vifb vifa
```

The N series virtual interface configuration is illustrated in Example 1-4.

Example 1-4 N series single-node LACP virtual interface configuration

```
NAS1> vif status
default: transmit 'IP Load balancing', VIF Type 'multi_mode', fail 'log'
vifa: 2 links, transmit 'IP Load balancing', VIF Type 'lacp' fail 'default'
     VIF Status    Up     Addr_set
    trunked: vif1
    up:
        e0c: state up, since 31Mar2011 18:50:30 (6+21:18:08)
                mediatype: auto-1000t-fd-up
                flags: enabled
                active aggr, aggr port: e0a
                input packets 675437, input bytes 43010404
                input lacp packets 19841, output lacp packets 19841
                output packets 122521, output bytes 7139083
                up indications 2, broken indications 0
                drops (if) 0, drops (link) 0
                indication: up at 31Mar2011 18:50:30
                        consecutive 0, transitions 2
        e0a: state up, since 31Mar2011 18:50:29 (6+21:18:09)
                mediatype: auto-1000t-fd-up
                flags: enabled
                active aggr, aggr port: e0a
                input packets 43968, input bytes 6839259
                input lacp packets 19840, output lacp packets 19841
                output packets 147663, output bytes 10142065
                up indications 2, broken indications 0
                drops (if) 0, drops (link) 0
                indication: up at 31Mar2011 18:50:29
                        consecutive 0, transitions 2
vifb: 2 links, transmit 'IP Load balancing', VIF Type 'lacp' fail 'default'
     VIF Status    Up     Addr_set
    trunked: vif1
    down:
        e0d: state up, since 31Mar2011 18:50:29 (6+21:18:09)
                mediatype: auto-1000t-fd-up
                flags: enabled
                active aggr, aggr port: e0b
                input packets 26427, input bytes 4709108
                input lacp packets 19841, output lacp packets 19841
                output packets 119029, output bytes 6626180
                up indications 2, broken indications 0
                drops (if) 0, drops (link) 0
                indication: up at 31Mar2011 18:50:29
                        consecutive 0, transitions 2
        e0b: state up, since 31Mar2011 18:50:29 (6+21:18:09)
                mediatype: auto-1000t-fd-up
                flags: enabled
```

```
                active aggr, aggr port: e0b
                input packets 659358, input bytes 41114314
                input lacp packets 19841, output lacp packets 19841
                output packets 119029, output bytes 6626180
                up indications 2, broken indications 0
                drops (if) 0, drops (link) 0
                indication: up at 31Mar2011 18:50:29
                        consecutive 0, transitions 2
vif1: 1 link, transmit 'none', VIF Type 'single_mode' fail 'default'
     VIF Status    Up     Addr_set
    up:
        vifa: state up, since 31Mar2011 18:50:29 (6+21:18:09)
                mediatype: Enabled virtual interface
                flags: enabled
                input packets 719405, input bytes 49849663
                output packets 270184, output bytes 17281148
                output probe packets 198378, input probe packets 198378
                strike count: 0 of 10
                up indications 1, broken indications 0
                drops (if) 0, drops (link) 0
                indication: up at 31Mar2011 18:50:29
                        consecutive 595085, transitions 1
    down:
        vifb: state down, since 31Mar2011 18:50:29 (6+21:18:09)
                mediatype: Enabled virtual interface
                flags: enabled
                input packets 685785, input bytes 45823422
                output packets 238058, output bytes 13252360
                output probe packets 198378, input probe packets 198378
                strike count: 0 of 10
                up indications 1, broken indications 0
                drops (if) 0, drops (link) 0
                indication: up at 31Mar2011 18:50:29
                        consecutive 595085, transitions 1
```

1.3 N series dual node solutions

The main idea for how to create a whole design solution and more ways for implementation are described in the companion IBM Redbooks publication *Introduction to IBM Real-time Compression Appliances*, SG24-7953, located at:

http://www.redbooks.ibm.com/abstracts/sg247953.html?Open

This section focuses on solutions using dual node N series controller.

1.3.1 Active/passive RTCA product on LACP bonds

The most common setup, illustrated in Figure 1-7 on page 12, is one that provides complete high availability for both the storage system and the RTCA product. This setup is an extension of the configuration that is described in the previous section 1.2.4, “Active/passive path, RTCA product HA, EtherChannel/LACP bonds” on page 8.

Here we have a combination of two N series single-node systems, each with its own active/passive links configured to two RTCA products. It is important to make sure that the basic configuration rules are applied—there can be only one specific RTCA product serving a specific NAS controller at a time.

> **Tip:** It is preferable to configure the links on the N series in such a way that a specific RTCA product is set as the primary and another RTCA product is set as the secondary. This configuration is done by using the `vif favor` command.

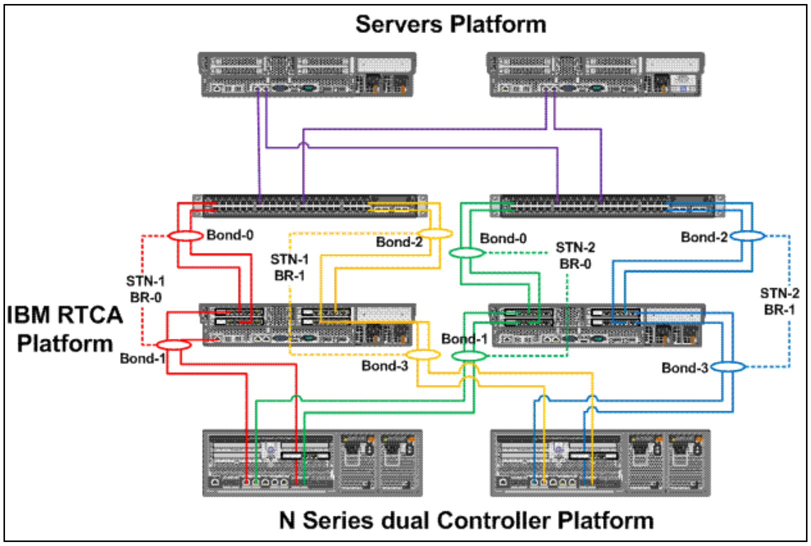


Figure 1-7 N series dual-node, active/passive RTCA product on LACP bonds

The port connectivity that belongs to the solution in Figure 1-7 is illustrated in Table 1-5.

Table 1-5 N series dual-node with EtherChannel or LACP bonds

Switch/ port	RTCA/ port	Bond	Bridge	Bond	RTCA/ port	NAS port/ vif
SW1/ 9	STN1/ eth1	bond0 LACP	STN-1 BR-0	bond1 LACP	STN1/ eth0	NAS1 e0a/ vifa
SW1/ 13	STN1/ eth5	bond0 LACP		bond1 LACP	STN1/ eth4	NAS1 e0c/ vifa
SW2/ 35	STN2/ eth9	bond0 LACP	STN-2 BR-0	bond1 LACP	STN2/ eth8	NAS1 e0b/ vifb
SW2/ 41	STN2/ eth13	bond0 LACP		bond1 LACP	STN2/ eth12	NAS1 e0d/ vifb
SW1/ 35	STN1/ eth9	bond2 LACP	STN-1 BR-1	bond3 LACP	STN1/ eth8	NAS2 e0b/ vifb
SW1/ 41	STN1/ eth13	bond2 LACP		bond3 LACP	STN1/ eth12	NAS2 e0d/ vifb
SW2/ 9	STN2/ eth1	bond2 LACP	STN-2 BR-1	bond3 LACP	STN2/ eth0	NAS2 e0a/ vifa
SW2/ 13	STN2/ eth5	bond2 LACP		bond3 LACP	STN2/ eth4	NAS2 e0c/ vifa

1.3.2 Controller configurations

Both N series controllers are configured in the exact same manner. Both vifa and vifb are configured as dynamic multi-mode vif and are grouped by creating a second layer of vif, called vif1, as a single-mode vif. Either vifa or vifb is active. In this way, data is transmitted through one RTCA product only and not at any time through both concurrently.

The configuration entries of the /etc/rc file of both controllers are identical. This is described in the previous section, 1.2.4, “Active/passive path, RTCA product HA, EtherChannel/LACP bonds” on page 8, and shown in Example 1-3 on page 10. Similarly, the vif configuration is described in the previous section, 1.2.4, “Active/passive path, RTCA product HA, EtherChannel/LACP bonds” on page 8 and shown in Example 1-4 on page 10.

1.4 N series MetroCluster solutions

IBM N series MetroCluster is a synchronous replication solution for combined high availability (HA) and disaster recovery (DR), protecting against site disasters within a campus or metropolitan area with distances up to 100 km. Usually, due to cabling and installation requirements, it is not possible to connect every N series controller of the HA pair with both RTCA products at each site to provide the best redundancy and availability.

Generally, you can configure the RTCA product in N series MetroCluster environments in a different manner:

- One RTCA product at each site to connect one RTCA product with only one N series controller (Figure 1-8 on page 14)
- Two RTCA products at each site to connect one N series controller redundant to two RTCA products (Figure 1-9 on page 14)

In the first option (Figure 1-8), it is absolute necessary to configure the N series to fail over to the partner site in case of an RTCA product failure. Additionally, it is preferable to connect each RTCA product to more than one switch; otherwise, a network switch failure will also require an N series controller failover.

Tip: To enable an automatic takeover, if the N series controller detects failures in the network interfaces, set the option “cf.takeover.on_network_interface_failure” to “on”.

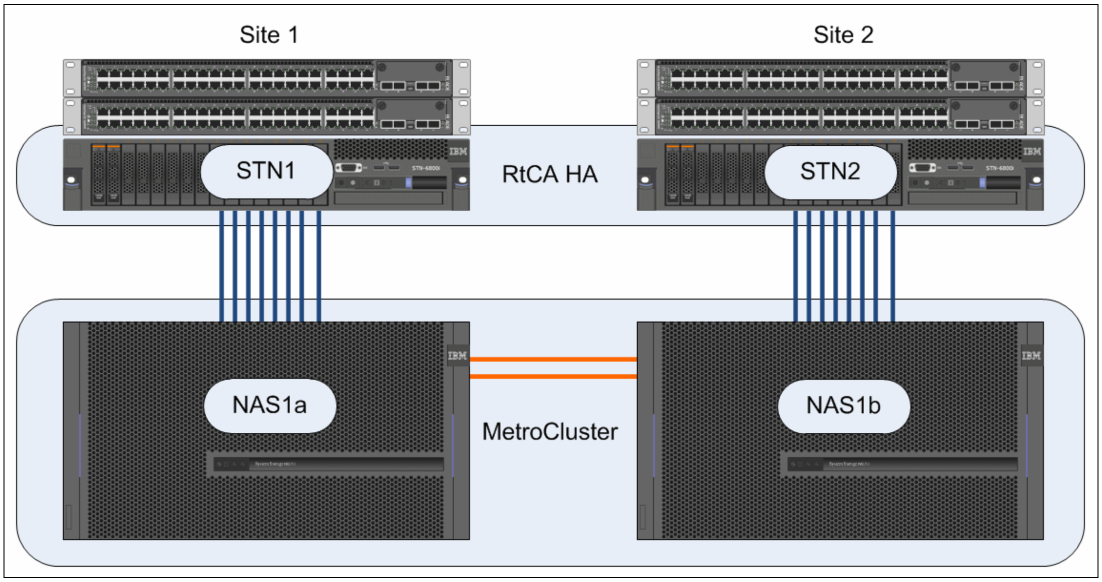


Figure 1-8 Single RTCA product design for MetroCluster

With the RTCA product unsynchronized, MetroCluster site failover is problematic. Therefore an HA RTCA product set up at each site with active/passive configuration is preferable, as illustrated in Figure 1-9.

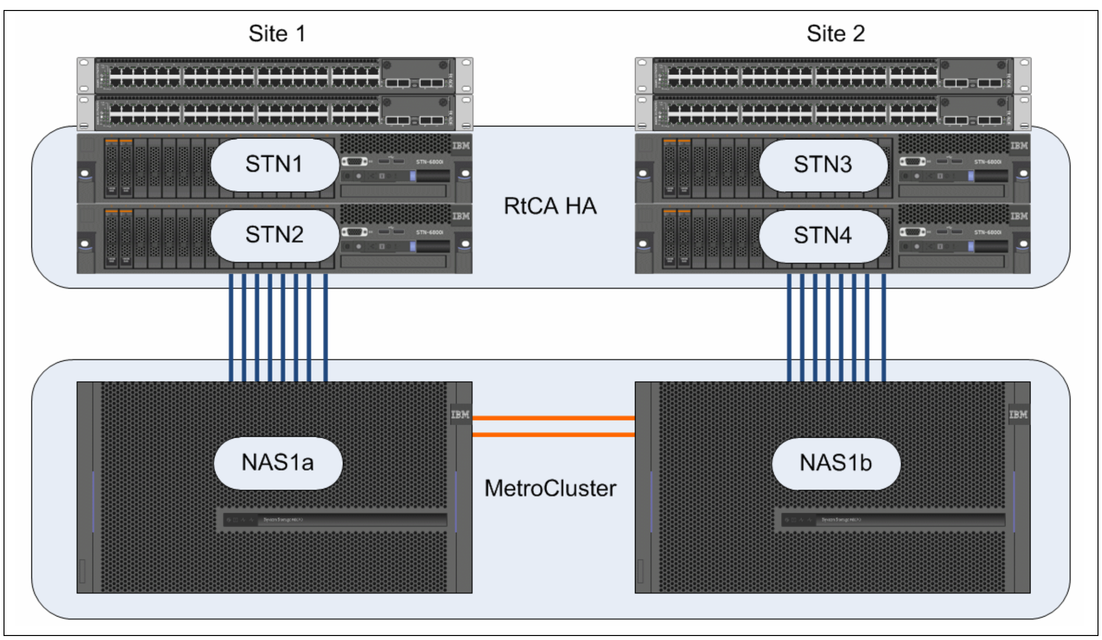


Figure 1-9 Dual RTCA product design for MetroCluster

In that architecture, STN1 and STN2 cannot see any shares of NAS1b, the remote cluster partner of the local controller NAS1a. However, it is necessary to create both storage server objects (for NAS1a and NAS1b) in all RTCA products at both sites. In such a configuration, it is best to build a star topology for the configuration of HA Auto Sync, as shown in Figure 1-10.

STN1 is configured as an HA pair with STN2, STN3, and STN4. The configuration of compression filters is performed only on one RTCA product, in this example STN1, to avoid synchronization issues. In other words: STN1 acts as a master and is paired with STN2, STN3, and STN4. Now, you can modify the compression filter configuration and your modification will be replicated to all other RTCA products immediately after you apply the changes.

If a compression filter is not defined as compressed in the compression filter list of all RTCA products in the same manner, data will not be accessible or corrupt after a cluster failover occurs and clients access the data through a logical bypass.

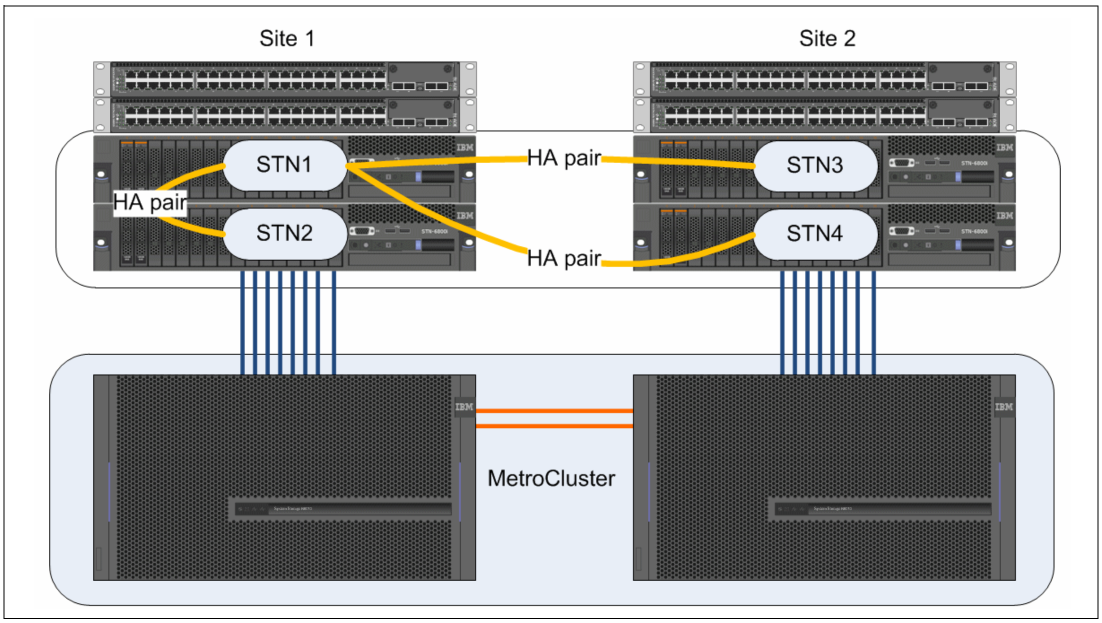


Figure 1-10 RTCA star topology

Depending on the architecture of a MetroCluster, a dual inter-site link failure can theoretically occur. Figure 1-11 shows the error message.

Important: It is not desirable to disable HA auto sync temporarily to modify a compression filter configuration after an inter-site link failure occurs. If you have a link problem, resolve it first, and then add or remove compression filters.

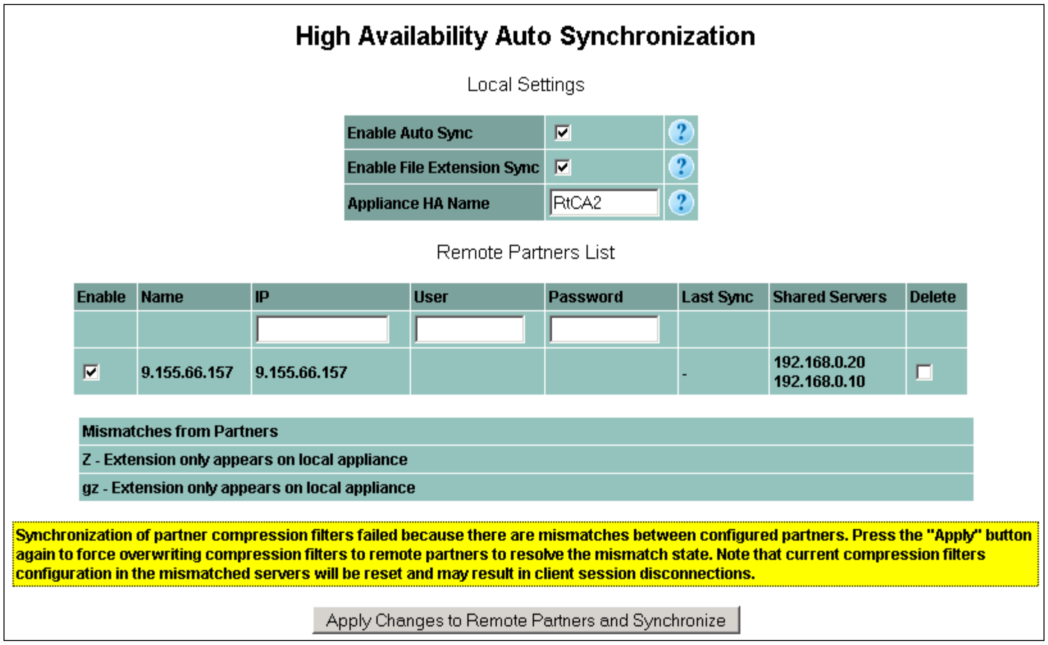


Figure 1-11 High Availability Auto Synchronization Error message

Attention: Synchronization of partner compression filters failed because there are mismatches between configured partners. Click the **Apply** button against it to force overwriting compression filters to remote partners to resolve the mismatch state. Note that the current compression filters configuration in the mismatched servers are reset and might result in client session disconnections.

In this case, the HA auto synchronization will not work and the RTCA product does not let you add or remove compression filters (Figure 1-12). However, it is possible to override it by disabling HA auto synchronization temporarily. It requires either using manual export/import, manual synchronization later, or performing the same task on all RTCA products.

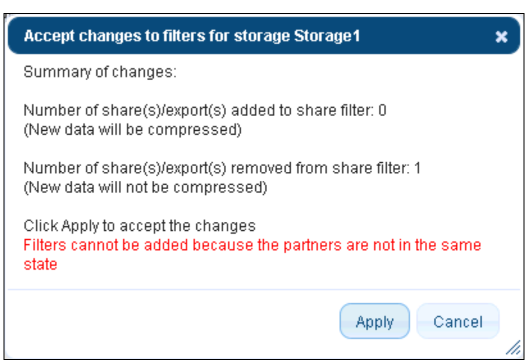


Figure 1-12 RTCA products do not let you add/remove compression filters

1.5 N series with MultiStore

The N series MultiStore feature provides the ability to configure virtual filers within a physical N series system. This feature is fully supported by the RTCA product, with each virtual filer (vfiler) configured as though it was a physical storage system. In other words, a vfiler is just another storage object in an RTCA configuration.

1.5.1 Considerations for vfilers

There are no special considerations for vfilers other than configuring their IP addresses and filters on all the RTCA products that might service them at any given time. This is mostly important if vfilers are replicated between an N series cluster and another one, such as with the DataMotion feature. In this environment, there typically are four N series controllers. NAS1a and NAS1b are the primary dual node system, and NAS2a and NAS2b are the secondary dual system, being the replica, as illustrated in Figure 1-13 on page 18.

1.5.2 Types of configurations

In the following sections, we describe the various configurations that are possible.

RTCA product configuration

STN1 is configured as an HA pair with STN2, STN3, and STN4. All four RTCA products are configured with all the vfiler IP addresses. The configuration of compression filters is performed only on one RTCA product, for example, STN1 to avoid synchronization issues. In such a configuration, it is best to build a star topology for the configuration of HA Auto Sync, in other words: STN1 is master and paired with STN2, STN3, and STN4. The HA configuration is similar to the RTCA product design in the MetroCluster example in Figure 1-10 on page 15.

Port connectivity

Any network configuration from the foregoing examples, typically a clustered NAS with active/passive failover on EtherChannel/LACP bonds, is usable for MultiStore configurations.

Network configuration

If all vfilers share the same IP space, all vfilers can be connected through the same bridges. If you have to create a vfiler that belongs to different IP spaces, it is necessary to create dedicated bridges, or in another way, use VLAN tagging (Figure 1-13).

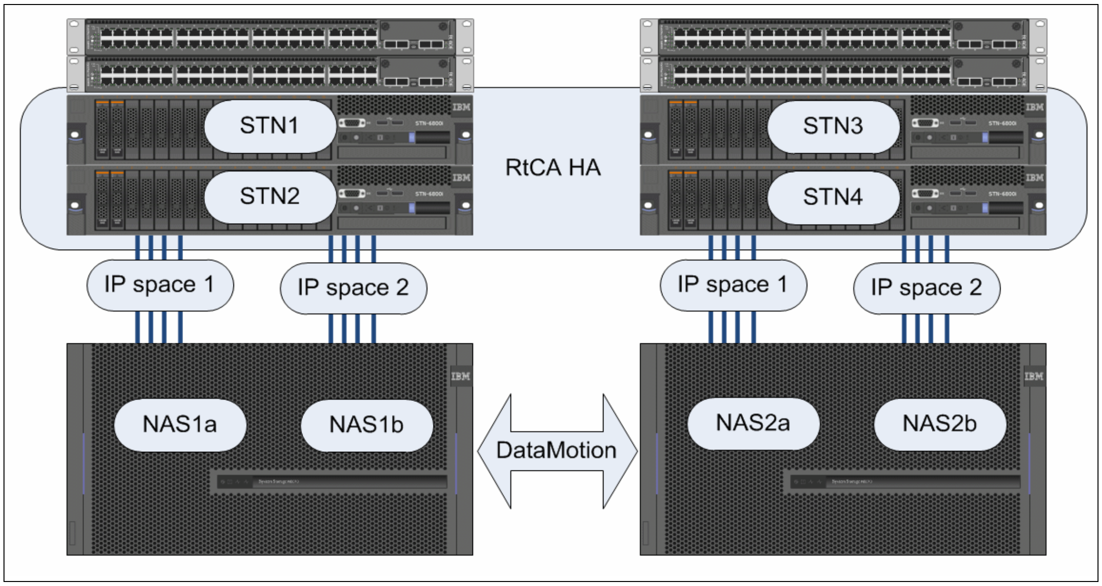


Figure 1-13 RTCA products in an N series MultiStore environment

2

EMC storage integration

In this chapter, we discuss IBM Real-time Compression Appliance solutions in a data center environment with EMC-supported NAS storage configurations.

You will understand the benefits and easy implementation of the IBM RTCA product in any of the supported NAS environments. These include IBM N series, Netapp, and also EMC Celerra, for example, if the EMC Celerra storage system is already installed in a data center environment, we can also think about a compression engine solution. Suppose that the customer uses EMC Celerra storage and they have performance and capacity improvement issues. The correct approach is to offer the benefit of the IBM Real-time Compression Appliance solution and explain how it can meet their expectations.

This chapter provides:

- Planning information
- IBM RTCA value added benefits
- IBM RTCA implementation in EMC NAS environment

We also discuss the basic steps before and during the installation process and validation of the system.

2.1 Planning information

This section demonstrates the proper installation and implementation procedure for the IBM Real Time Compression Appliance solution with EMC storage. The most important step for finding a proper solution is pre-installing the IBM Technical and Delivery Assessment (TDA) process. The TDA process is part of the IBM Solutions Assurance procedure and it is important in finding a proper solution. Pre-installation of TDA must be done before installation of the IBM RTCA solution. You must meet all of the IBM requirements for pre-installing TDA. Contact your local IBM or Business Partner contact to request this service offering.

Physical configuration and installation planning for the IBM Real-time Compression Appliance is the customer's responsibility. The customer must install the machine:

http://www-01.ibm.com/common/ssi/rep_ca/6/877/ENUSZG10-0406/ENUSZG10-0406.PDF

> **Important:** Only a person with knowledge and experience must install this product, because only in that case can customers expect a safe RTCA implementation, following good safety practices. We strongly advise using IBM Implementation Services. For more information, contact your local IBM sales representative.

2.2 IBM RTCA value added benefits

Comparing the IBM Real-time Compression Appliance with other builds in firmware NAS storage compression engines, the main performance benefits are as follows:

- All compression processes are segregated from the main NAS storage systems and processed in the external appliance module or modules (if RTCA HA is included).
- The NAS storage CPUs are only dedicated to input/output operations and handle advanced storage features, such as point-in-time copy, local or external data replication, data mirroring, RAID features, and so on.
- When using internal NAS compression features, there might be increased CPU utilization visible because compression-related tasks utilize the NAS appliance CPU directly. In contrast, when using an external appliance, there is CPU performance off-load in place for the NAS appliance.
- There is easy management regulation.
- Fast segregation of problems expedites troubleshooting procedures.

These are the main improvements that we can accomplish using the IBM RTCA solution in any supported NAS environments:

- The RTCA compression dramatically decreases storage capacity, depending on file types or the application environment. The compression ratio can go from 30% up to 80%.
- Supporting online and active primary applications.
- Optimizing and leveraging investments in existing storage and new storage solutions.
- Analyzing compression status and Random Access Compression Engine (RACE) engagement during the data store process. We can easily determine which files are used most and which files are ready for the next step in data store process: archiving.
- Managing more data per defined time.
- Enhancing energy efficiency.
- Reducing capital and operating costs.

2.3 IBM RTCA implementation in EMC NAS environments

In the IBM RTCA interoperability matrix, the only supported NAS vendors are IBM N series, Netapp, and EMC Celerra. We already described the IBM N series and Netapp in the previous chapters. We can now explain in a simple way whether the IBM RTCA product can be implemented in a customer EMC Celerra environment.

The EMC Celerra is EMC's NAS product, and it too has high-availability network features like other NAS storage systems, such as IBM N series and Netapp. Here we describe two possible ways of using the IBM RTCA product with the EMC NAS storage system:

- The first option is applied when EMC Celerra is already installed and is working in a current data environment, and the IBM RTCA product must be installed within it.
- The second option is applied when a customer wants to use EMC Celerra system along with the compression engine feature. In that case, we install IBM RTCA and EMC Celerra from the beginning.

2.3.1 The first option

Here, the EMC Celerra storage is already installed and it is running. The customer wants to implement a compression engine in the current solution without decreasing the storage performance. In that case, we need to offer an external appliance that resolves the compression engine without the issue of decreasing storage performance. In addition, the other benefits possible are described in 2.2, "IBM RTCA value added benefits" on page 20.

Suppose the current EMC Celerra storage system was properly installed in the network with a high availability (HA) solution. In this case, implementing the IBM RTCA product can be easily done without destroying the current production system, as shown in Figure 2-1 on page 22.

We must check whether the HA network feature on EMC Celerra is properly installed and configured. If it is not, the first step in implementation must be installing network HA on the storage system.

EMC Celerra has *Filesafe networking* interfaces for failover of physical network ports. Filesafe networking allows network ports to have a dedicated passive (backup) port on a separate network card in case of failure. It is not important whether the reason for the failure was a physical network card in the storage system, an external network switch, or a router. Filesafe networking handles the failover on the physical network card on the EMC Celerra. Every physical port on EMC Celerra supports two standards for aggregation. One is the industry standard Link Aggregation IEEE802.3ad and the other is Ethernet Port Trunking.

Next we describe the procedure for implementing the IBM RTCA in an EMC Celerra environment.

Step 1. We advise that you first check the NAS Storage (EMC Celerra) failover data path configuration. If it is good and the configuration is similar, as shown in Figure 2-1, we can go to the next step. If the failover data path is not configured, as we discussed, the first step in the IBM RTCA implementation must be to reconfigure EMC Celerra data paths.

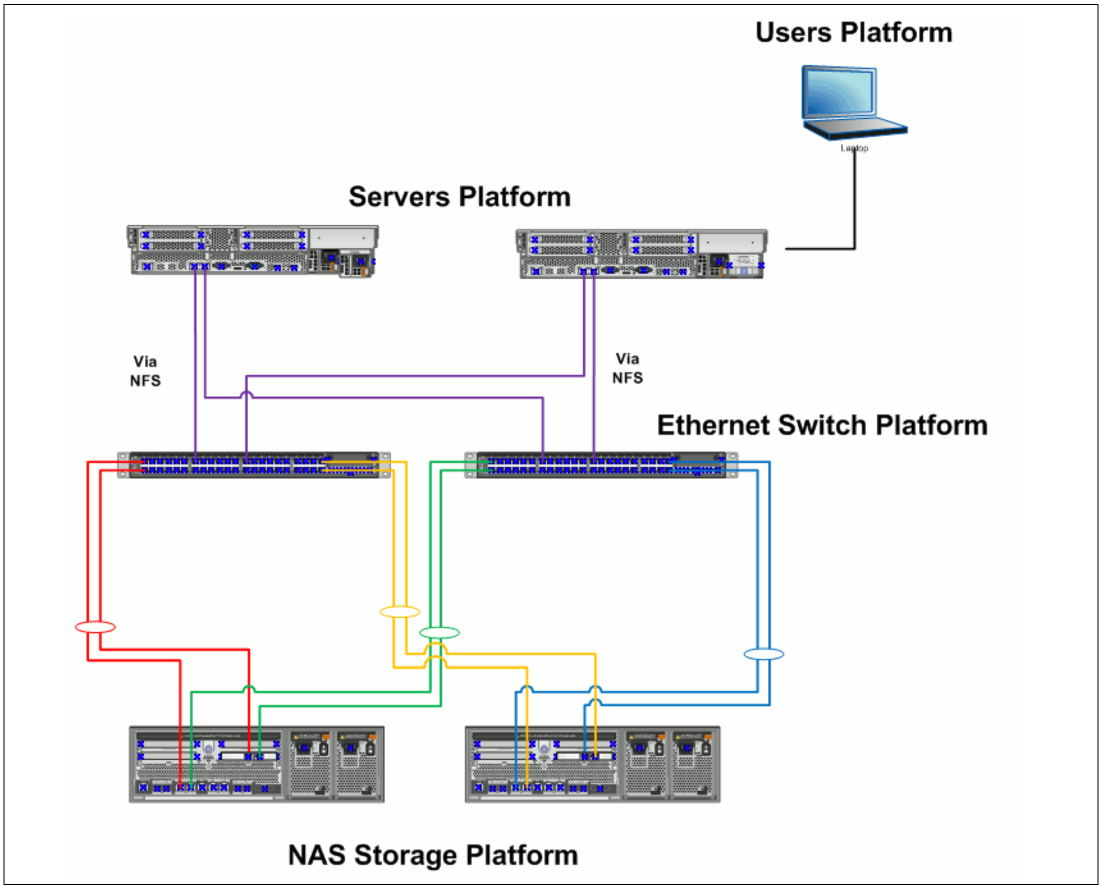


Figure 2-1 RTCA and NAS integration without single point of path failure

Step 2. All data must be transferred through predefined active storage paths (Figure 2-2). Backup data paths are used for IBM RTCA implementation. When we implement the IBM RTCA product using backup data paths, we will create and configure IBM RTCA bonds and bridges. This is done as described in the companion IBM Redbooks publication *Introduction to IBM Real-time Compression Appliances*, SG24-7953, located at:

http://www.redbooks.ibm.com/abstracts/sg247953.html?Open

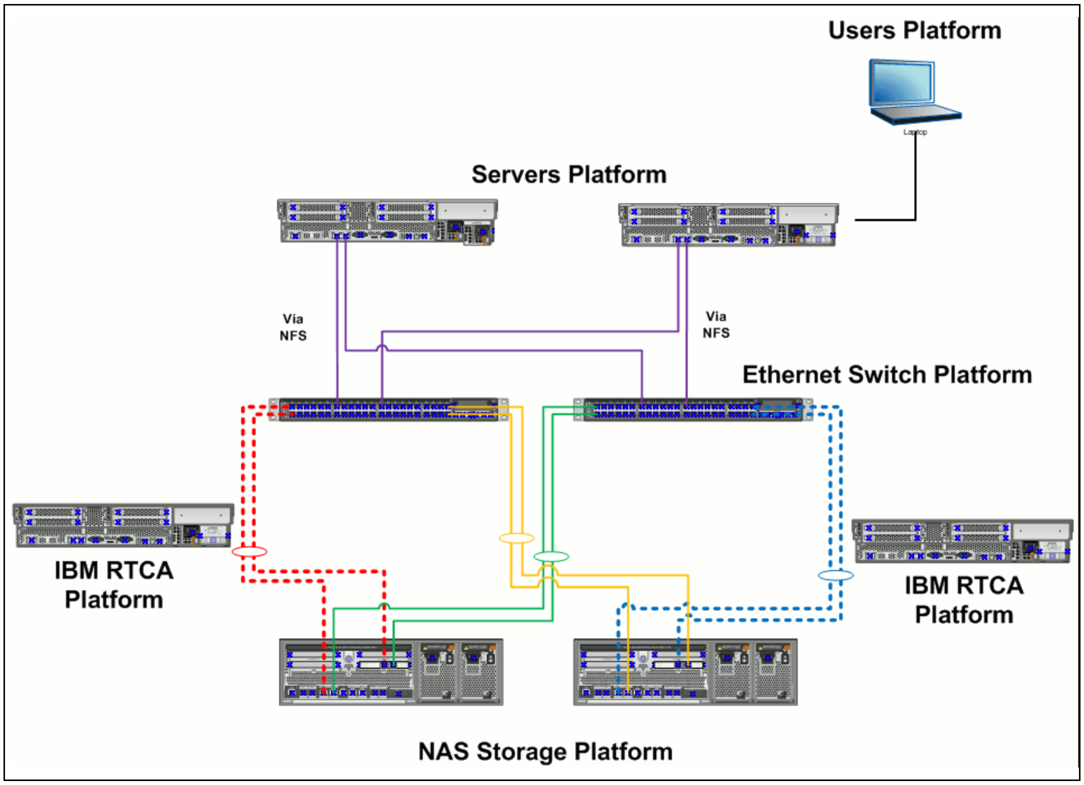


Figure 2-2 All data transferred through the active data path

IBM RTCA implementation using backup data path and bonds and bridges configuration is shown in Figure 2-3.

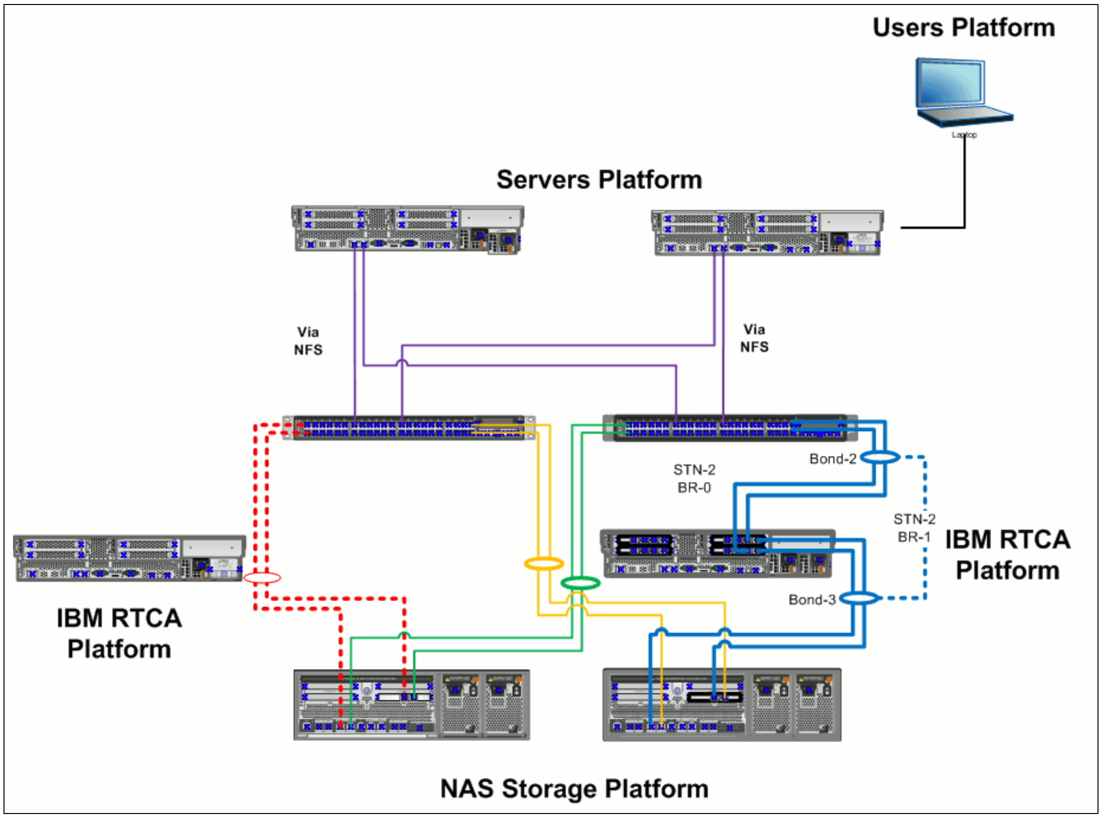


Figure 2-3 Creating IBM RTCA bonds and bridges on the backup data path

Step 3. When we are finished with the IBM RTCA installation, as shown in Figure 2-3, the next step is to reroute the active paths. Active data paths now are working in transparent IBM RTCA mode, as described in the companion IBM Redbooks publication *Introduction to IBM Real-time Compression Appliances*, SG24-7953.

All data now must be rerouted from the current active storage data paths through current backup data paths. After this rerouting scenario, we must integrate all remaining backup data paths with the IBM RTCA product. New active data paths are still working in IBM RTCA transparent mode, as illustrated in Figure 2-4 on page 25.

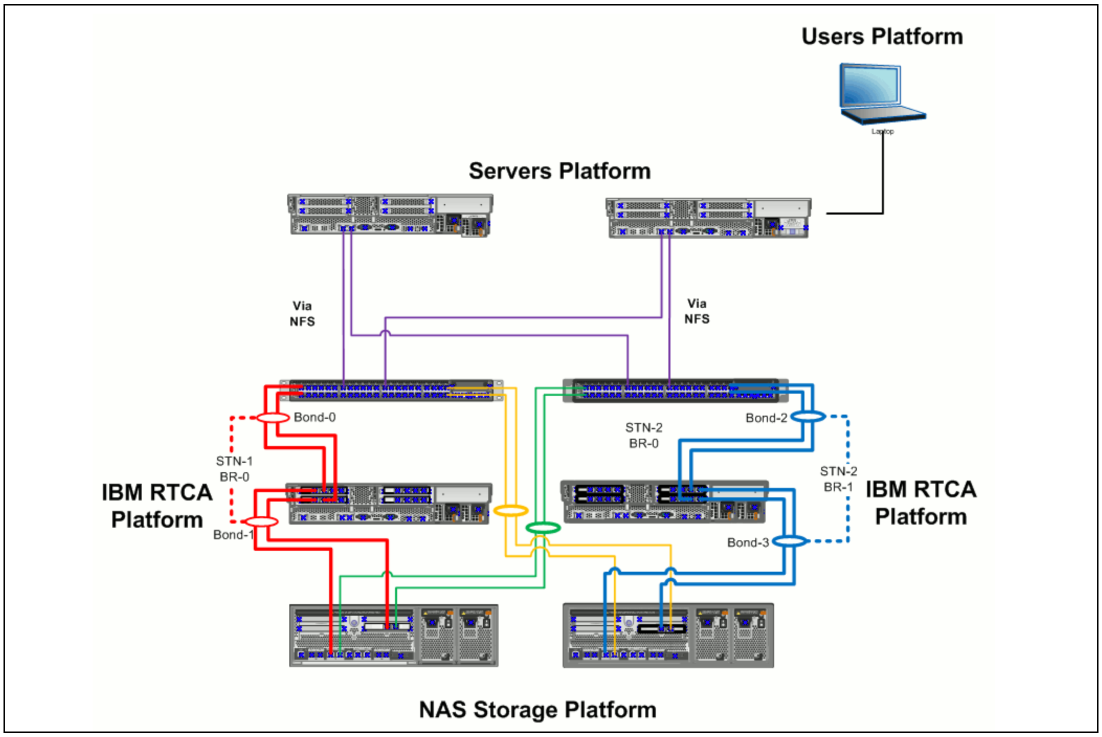


Figure 2-4 IBM RTCA bonds and bridges on the active and backup data path

Step 4. After the IBM RTCA integration is completed within the NAS storage environment, we can start working with the compression engine.

The final configuration with the IBM RTCA product implemented in the environment is shown in Figure 2-5.

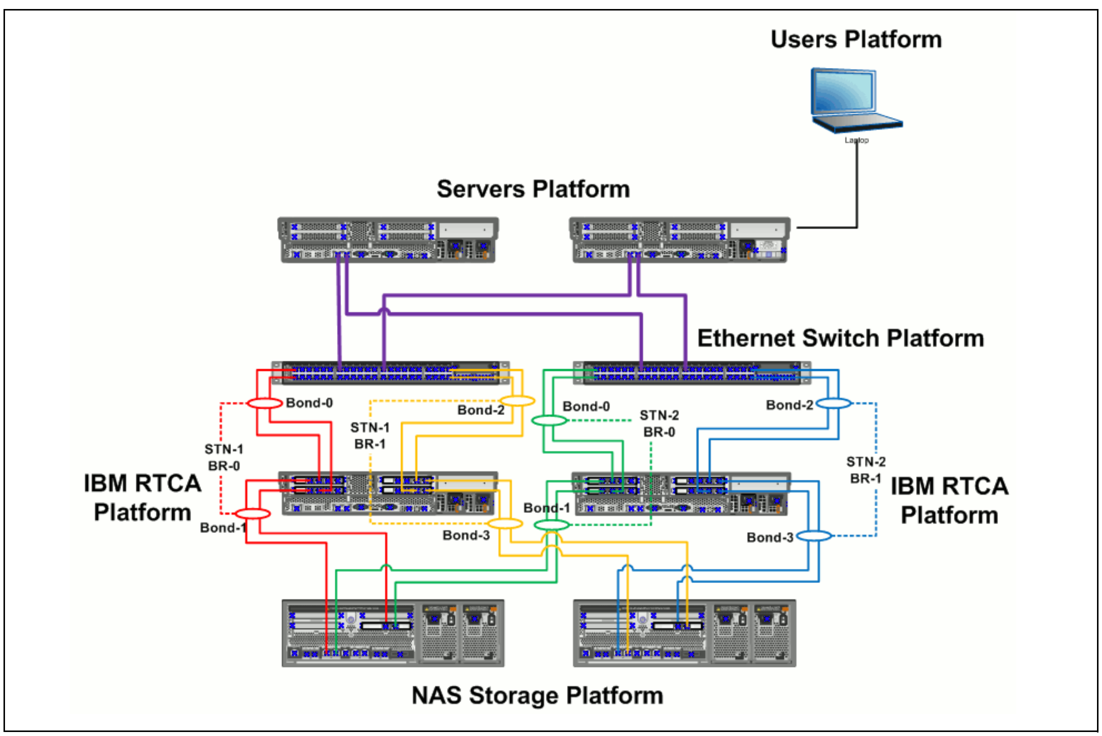


Figure 2-5 IBM RTCA completed integration in NAS Storage environment

2.3.2 The second option

Suppose that a customer wants to use the EMC Celerra storage system but needs a data compression engine without decreasing the storage performance. In this case, we can implement IBM RTCA and EMC Celerra storage from the beginning. We install all storage components, create logical structure of storage, and define and implement network solution. On the storage side, properly install network failover and both active and passive physical ports on the storage machine. The IBM RTCA appliance is included in the solution before the time of production. The whole solution made of EMC Celerra storage system, IBM RTCA appliance, and network switches will be implemented, tested, and verified and only then can we confirm that the solution is ready for production.

During these IBM RTCA implementation steps, we use planning and deploying information, as described in the companion IBM Redbooks publication, *Introduction to IBM Real-time Compression Appliances*, SG24-7953, located at:

http://www.redbooks.ibm.com/abstracts/sg247953.html?Open

The most important thing during the storage configuration is to prepare the storage system for primary and redundancy data paths through Ethernet switches (primary and redundancy switches for network high availability).

Also important is the fact that the IBM RTCA product can be installed in a one-node configuration, but this is not desirable. In that case, we do not have an HA appliance solution; instead, we have a single point of failure for data paths and for the appliance. Having only one compression appliance is a single point of failure. Because of that, we strongly advise using a two-way RTCA configuration. An HA solution is reached with a minimum of primary and redundancy data path, network, and appliance high availability.

3

Nearline and offline backup solutions

In this chapter, we explain how to combine the IBM Real-time Compression Appliance with several nearline and offline backup solutions. This enables you to understand the benefits of real-time compression in the complete data life-cycle.

3.1 RTCA and SnapVault

SnapVault is a replication-based disk-to-disk backup for IBM System Storage N series or Network Appliance filers. It creates full point-in-time copies on disk. It uses a block incremental methodology.

3.1.1 Overview of SnapVault

SnapVault is a separately licensed feature in Data ONTAP that provides disk-based data protection for storage systems. The SnapVault server runs on the IBM System Storage N series platform. However, you can use an IBM System Storage N series storage system as a SnapVault client as well.

SnapVault replicates selected Snapshots from multiple client storage systems to a common Snapshot on the SnapVault server, which can store many Snapshots. These Snapshots on the server have the same function as regular tape backups (Figure 3-1). Periodically, data from the SnapVault server can be dumped to tape for extra security.

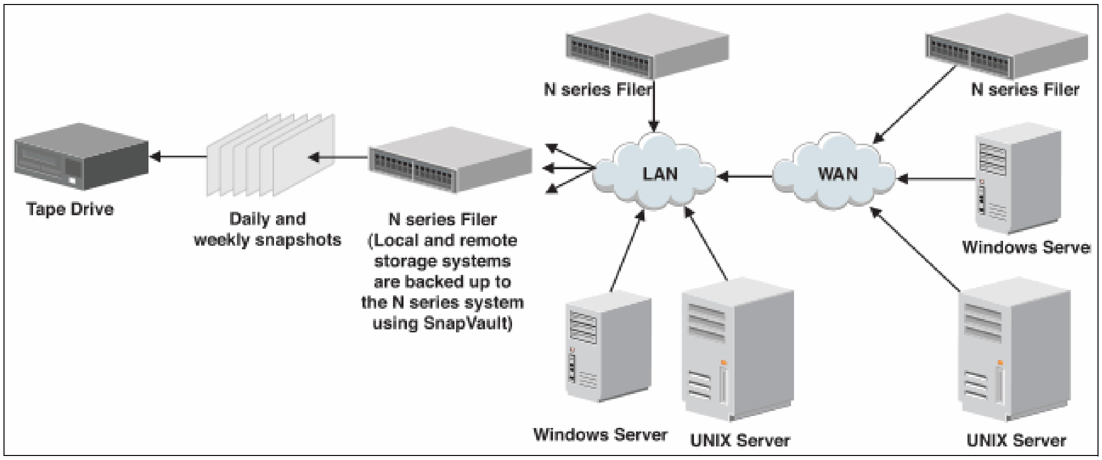


Figure 3-1 IBM System Storage N series with SnapVault configuration

SnapVault is a heterogeneous disk-to-disk data protection solution that is ideal for use with IBM System Storage N series storage systems. A SnapVault primary system corresponds to a backup client in the traditional backup architecture. The SnapVault secondary is always an IBM System Storage N series storage system running Data ONTAP. SnapVault software protects data residing on a SnapVault primary.

All of this heterogeneous data is protected by maintaining online backup copies (Snapshot) on a SnapVault secondary system. The replicated data on the secondary system can be accessed through Network File System (NFS) or Common Internet File System (CIFS), just as regular data can be. The primary systems can restore entire directories or single files directly from the secondary system. There is no corresponding equivalent to the SnapVault secondary in the traditional tape-based backup architecture.

Reference: For more information about SnapVault, see the Redbooks publication, *IBM System Storage N series Software Guide*, SG24-7129, available at this website:

http://www.redbooks.ibm.com/redpieces/abstracts/sg247129.html

3.1.2 Benefits of using SnapVault with IBM RTCA

When using SnapVault in combination with the IBM Real-time Compression Appliance the benefits are significant on different levels, which we describe next.

Bandwidth

Because the RTCA product compresses the client data before writing to the source NAS storage controller, the SnapVault feature will read, send, and store the data in compressed form.The result is that when using the IBM Real-time Compression Appliance, the bandwidth needed to copy is far less than when you copy uncompressed data.

Backup window

Because only the compressed blocks are copied, the data that needs to be copied will be smaller and thus the backup window will be smaller.

Processing power

The IBM Real-time Compression Appliance adds processing power to do the job of compressing the data, when using the native compression inside of the filers this can add up to 30% of extra load to your storage processor.

Both the bandwidth and backup window benefits cannot be claimed when using the native compression in the box as data will be uncompressed to be transported to the other filer.

3.2 Combining RTCA and deduplication

To understand the benefits of combining real-time compression and deduplication, it is important first to understand the difference in the technologies and where they fit in the overall data optimization landscape.

Real-time compression is designed to work transparently in front of your primary storage and reduce the size of every file you create up to five times depending upon file type. Applications have random, read-write access to compressed data while the physical capacity required to store a file, or copies and permutations of a file are significantly reduced throughout the entire life-cycle including backup. Because less data is written to disk, overall network and storage performance and utilization can also be significantly enhanced.

Deduplication is designed to reduce the physical storage required to store redundant data. In the deduplication process, duplicate data is removed and replaced with a pointer to the main copy, leaving only one copy of the data to be stored which is why it is well suited for backup data where you typically have multiple data sets (daily/weekly) of mostly redundant data. The more copies of redundant data you have, the higher your effective deduplication rate. An example of this situation is illustrated in Figure 3-2 on page 30. Hence real-time compression and deduplication address different problems and are applicable at different points within the data life-cycle. But, most importantly the two technologies are complementary. In particular, they deploy real-time compression with a random access compression engine, significantly enhancing the value and performance of deduplication.

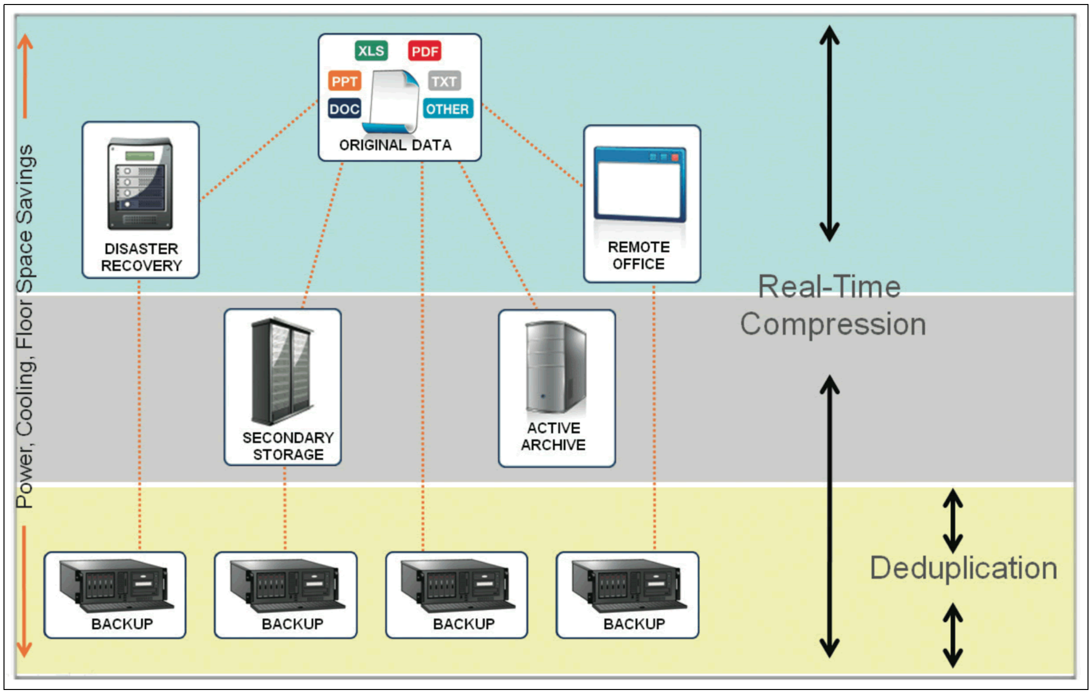


Figure 3-2 The Data Optimization Landscape

3.2.1 Combining RTCA and IBM ProtecTIER deduplication

Our test environment, shown in Figure 3-3 on page 31, provides a combination of IBM Real-time Compression and IBM ProtecTIER® deduplication in an Oracle database backup environment. This resulted in more than 96% savings on disk space and additional significant data protection benefits for organizations that perform regular backups of Oracle databases. Each technology offers compelling benefits when implemented independently. Also, the combination of both technologies results in far greater space savings, significantly improved backup time, and resource utilization.

In the case of Oracle database backups, real-time compression provided these benefits:

- An 82% immediate savings on initial write to disk.
- A 96% overall data reduction when combined with deduplication
- Up to 71% reduction in backup time
- Less deduplication for CPU utilization, less deduplication for disk activity, and less deduplication for network traffic

The significant overall reduction in primary and backup data storage requirements coupled with the dramatic improvement in deduplication CPU, Network, and disk utilization illustrate the significant synergies between real-time compression and deduplication.

Running with RTCA compression and ProtecTier deduplication and compression reduced the used size by an additional 10%, and this is the preferable configuration.

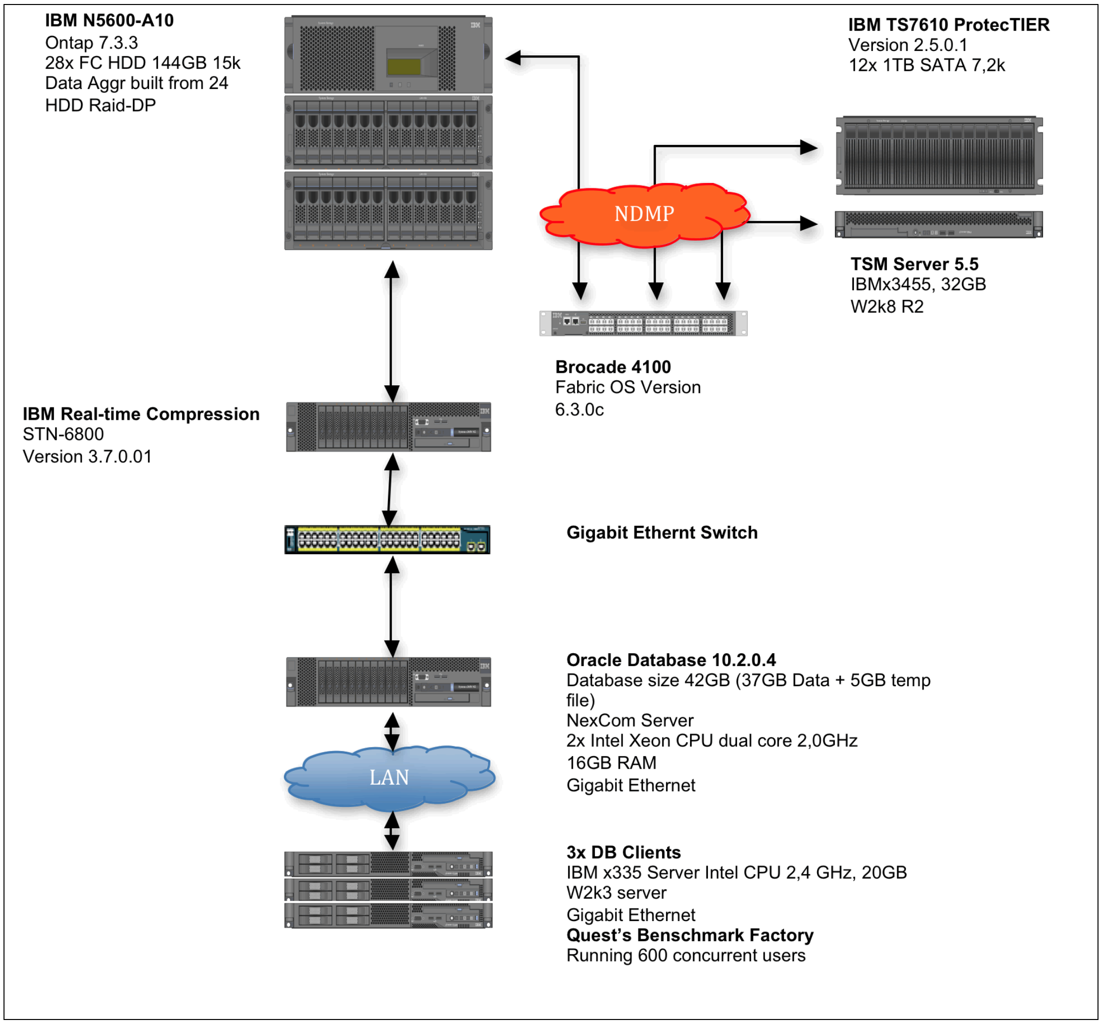


Figure 3-3 RTCA and ProtecTIER test setup for Oracle environment

Based on the ProtecTIER logs review, the RTCA absorbs all of the small changes and prepares the data for the ProtecTIER in a more linear way, allowing it to store the deduplicated data much more efficiently.

3.2.2 RTCA and EMC Data Domain

As for the previous test in 3.2.1, “Combining RTCA and IBM ProtecTIER deduplication” on page 30, IBM used a similar test setup for the IBM Real-time Compression Appliance in combination with EMC’s Data Domain Appliance.

The combination of IBM Real-time Compression and EMC Data Domain deduplication can result in enhanced overall storage efficiency, as well as significant reductions in deduplication requirements for system CPU and disk utilization and network bandwidth. In the case of Oracle database backups, IBM Real-time Compression provided these benefits:

- More than 80 percent immediate savings on initial write to disk
- More than 93 percent overall reduction in storage space when combined with deduplication
- A 72% reduction in backup time
- A 74% reduction in deduplication CPU utilization
- A 67% reduction in deduplication disk activity
- A 77% reduction in deduplication network traffic

The significant overall reduction in primary and backup data storage requirements, coupled with the dramatic reduction in requirements for deduplication system and network resources, illustrates the significant synergies between real-time compression and deduplication.

The combination of IBM Real-time Compression and the deduplication capabilities of EMC Data Domain products in an Oracle database backup environment resulted in more than 93 percent savings on disk space and additional significant data protection benefits. Whereas each technology offers compelling benefits when implemented independently, the combination of the two technologies results in far greater space savings, significantly improved backup time, and better resource utilization.

For more information, see *IBM Real-time Compression and EMC Data Domain deduplication: Working together*, at the following website:

http://public.dhe.ibm.com/common/ssi/ecm/en/tsw03082usen/TSW03082USEN.PDF

3.3 IBM RTCA and NDMP backup to tape

The NDMP is a standardized protocol for controlling backup, recovery, and other transfers of data between primary and secondary storage devices, such as storage systems and tape libraries.

By enabling NDMP protocol support on a storage system, you enable that storage system to carry out communications with various components. These include NDMP-enabled commercial network-attached backup applications (also called data management applications, or DMAs), data servers, and tape servers participating in backup or recovery operations.

NDMP also provides low-level control of tape devices and medium changers.

Data ONTAP NDMP backup and recovery operations use the same dump and restore services that you might use for data backup to tape and data restoration from tape. However, accessing these data protection services through backup applications that support NDMP offers a number of advantages.

These are some of the advantages:

- NDMP backup applications provide sophisticated scheduling of data protection operations across multiple storage systems. They also provide media management and tape inventory management services to eliminate or minimize manual tape handling during data protection operations.
- NDMP backup applications support data-cataloging services that simplify the process of locating specific recovery data. Direct Access Recovery (DAR) optimizes the access of specific data from large backup tape sets.
- NDMP supports multiple topology configurations, allowing efficient sharing of secondary storage (tape library) resources through the use of three-way network data connections. NDMP backup applications typically provide user-friendly interfaces that simplify the management of data protection services. Figure 3-4 shows an NDMP backup example with IBM Tivoli® Storage Manager.

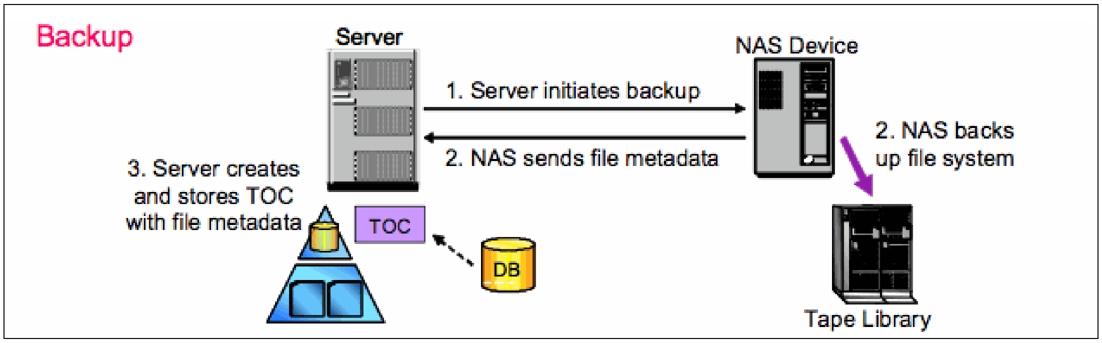


Figure 3-4 NDMP backup example with Tivoli Storage Manager

The NAS NDMP copy feature is unaffected by the RTCA product. Because the NDMP is initiated at the logical (file system) level the data is copied to the new volume in compressed form. The metadata sent to the backup server is untouched so the backup server knows exactly what is written to the tapes. The space used on the backup medium is far less than without the IBM Real-time Compression Appliance.

In contrast, because NDMP occurs at the logical (file system) level, any NAS deduplicated, or NAS compressed data must be uncompressed in memory on the NAS controller before being sent across the network and stored on the target in uncompressed form.

When the data is restored, be sure that this happens to a volume that can be decompressed either through an IBM Real-time Compression Appliance or a volume that is accessed first by the IBM RTCA recovery utility.

> **NDMP:** For more information about NDMP backups with IBM N series, see *Using the IBM System Storage N series with IBM Tivoli Storage Manager*, SG24-7243, at this website:
>
> `http://www.redbooks.ibm.com/redpieces/abstracts/sg247243.html`

3.4 Conclusion

When using the IBM Real-time Compression Appliance, you not only save space on your primary data but also on your secondary and backup data. The benefits do not stop there. You also benefit on bandwidth, which in turn saves you on networking and ISP costs.

4

VMware vSphere integration

This chapter contains the best practices we advise for use with virtualization.

More specifically, for virtualization with VMware, we discuss best practices regarding the following topics:

- Setting up the IBM N series for use with VMware
- VMware vSphere and use with NFS datastores
- The network environment for use with VMware and NFS datastores
- The use of VMware with the IBM Real-time Compression Appliance

4.1 IBM N series and VMware vSphere

This section briefly covers the use of VMware vSphere with an IBM N series system.

> **Reference:** This section is an abstract of the Redbooks publication, *IBM System Storage N series and VMware vSphere Storage Best Practices*, SG24-7871. For current information, visit the following website:
>
> http://www.redbooks.ibm.com/abstracts/sg247871.html

4.1.1 vSphere in an N series MetroCluster environment

N series MetroCluster configurations consist of a pair of active-active storage controllers configured with mirrored aggregates and extended distance capabilities to create a high-availability solution. The primary benefits are:

- Higher availability with geographic protection
- Minimal risk of lost data, easier management and recovery, and reduced system downtime
- Quicker recovery when a disaster occurs
- Minimal disruption to users and client applications

A MetroCluster (either Stretch or Fabric) behaves in most ways similar to an active-active configuration. All of the protection provided by core N series technology (RAID-DP, snapshot copies, automatic controller failover) also exists in a MetroCluster configuration. MetroCluster adds a complete synchronous mirroring and the ability to perform a complete site failover, from a storage perspective, with a single command.

The following N series MetroCluster type exists and works seamlessly with the complete VMware vSphere and ESX server portfolio:

- Stretch MetroCluster (sometimes referred to as nonswitched):

  This type is an active-active configuration that can extend up to 500 m, depending on speed and cable type. It also includes synchronous mirroring (SyncMirror) and the ability to do a site failover with a single command.

- Fabric MetroCluster (also referred to as switched):

  This type uses four Fibre Channel switches in a dual-fabric configuration and a separate cluster interconnect card to achieve an even greater distance (up to 100 km depending on speed and cable type) between primary and secondary locations.

4.1.2 Integration and redundancy

The integration of the MetroCluster and VMware vSphere provides seamless integration and storage and application redundancy. In addition, to connecting to the vSphere environment using FCP, iSCSI, or NFS, the solution can serve other network clients with CIFS, HTTP, and FTP simultaneously. The solution shown in Figure 4-1 on page 37 provides redundant VMware vSphere server, redundant N series heads, and redundant storage.

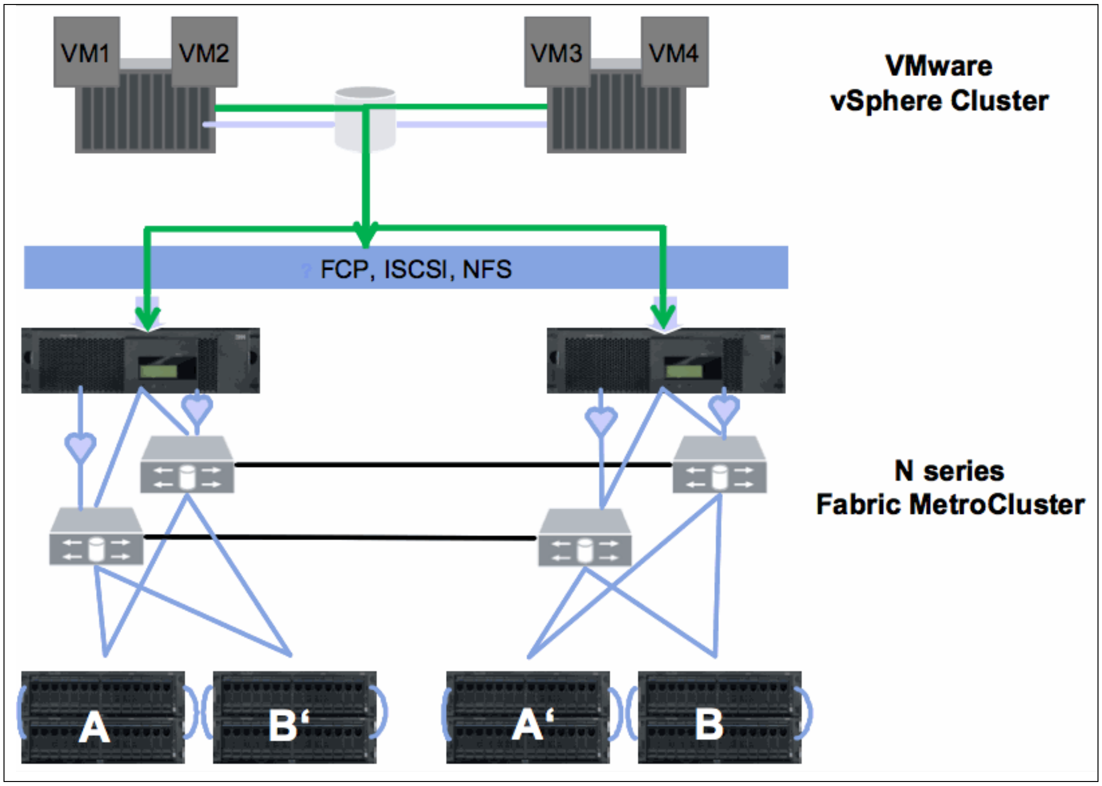


Figure 4-1 MetroCluster and VMware vSphere integrated solution

For more information about N series MetroCluster, see the “MetroCluster” chapter in *IBM System Storage N series*, SG24-7129.

4.2 VMware vSphere and NFS datastores

This section briefly covers the use of NFS datastores on an IBM N series with a VMware vSphere environment.

> **Reference:** This section is an abstract of the Redbooks publication, *IBM System Storage N series and VMware vSphere Storage Best Practices*, SG24-7871. For current information, see the following website:
>
> http://www.redbooks.ibm.com/abstracts/sg247871.html

4.2.1 Increasing the number of NFS datastores

By default, VMware ESX is configured with eight NFS datastores. However, this limit can be increased to 64 to meet the needs as the virtual infrastructure grows. Although the maximum number of NFS datastores (64) is less than what is available with VMFS datastores (256), this difference is offset by the density available to N series NFS datastores.

To be sure of availability, increase the maximum number of datastores available when deploying a VMware ESX Server. This is because pre configuring this setting ensures that NFS datastores can be dynamically added at any time without disruption or effort.

To make this change, perform the following steps from within the virtual infrastructure client:

1. Open vCenter Server.
2. Select a VMware ESX host.
3. Select the Configuration tab.
4. In the Software box, select **Advanced Configuration**.
5. In the dialog box (left pane), select **NFS**.
6. Change the value of NFS.MaxVolumes to 64 (see Figure 4-2).
7. In the dialog box (left pane), select **Net**.
8. Change the value of Net.TcpIpHeapSize to 30.
9. Change the value of Net.TcpIpHeapMax to 120.

Repeat the steps for each VMware ESX Server.

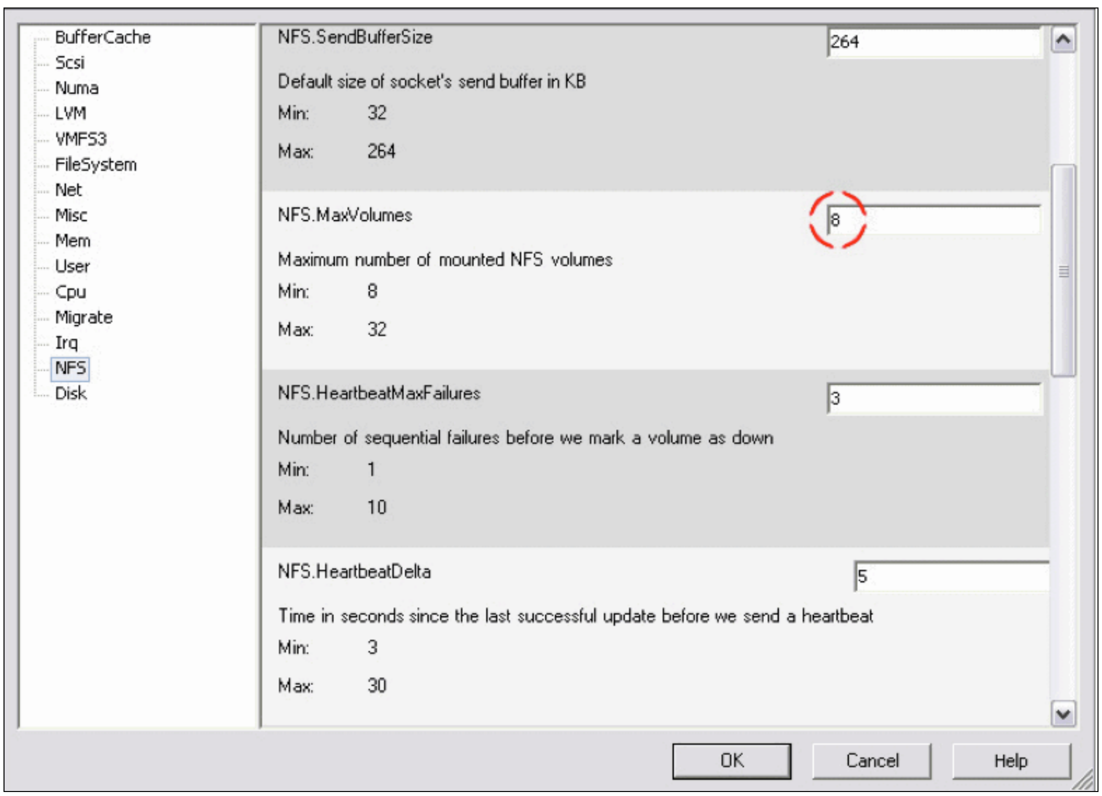


Figure 4-2 Increasing the maximum number of NFS datastores

4.2.2 File system security

The N series storage systems permit you to set the security style of each Flexible Volume (or file system) to use UNIX permissions or NTFS permissions. File system security can be mixed and matched with share or export security. As an example, a UNIX share (or export) can allow access to a file system with NTFS permissions and vice versa. In addition, security style can also be made on a file-by-file basis using the MIXED permissions setting.

For VMware ESX deployments, be sure to set the security style of all datastores to UNIX. The security setting of the root volume will be the security setting when a new volume is created.

Typically, customers who run VMware ESX on NFS want to access their datastores from Windows systems, to be able to complete administrative functions. Considering this situation, set the volume security style to UNIX, and make sure that the N series user mapping is set up correctly to enable Windows users access to this data.

If you must change the file system security type:

1. Log in to the N series console.
2. From the storage appliance console, run the following command:

   ```
   qtree security <volume path> UNIX
   ```

Repeat these steps for each NFS accessed volume.

4.2.3 VMware ESX NFS time-out settings

When connecting to NFS datastores, we advise adjusting a few NFS options around connection monitoring and resiliency. These settings can be automatically set for you if you decide to install the N series VMware ESX Host Utilities. The VMware ESX Host Utilities are supported only with VMware ESX. Therefore, if you are running VMware ESXi, do not install the VMware ESX Host Utilities, but update these settings by performing the following steps on each VMware ESX 4.0 host:

> **Attention:** For optimal availability with NFS datastores, make the following changes on each VMware ESX 4.0 host.

1. Open vCenter Server.
2. Select a VMware ESX host.
3. Select the Configuration tab.
4. In the Software box, select **Advanced Configuration**.
5. In the dialog box (left pane) select **NFS**.
6. Change the value of NFS.HeartbeatFrequency to 12.
7. Change the value of NFS.HeartbeatMaxFailures to 10.

Repeat these steps for each VMware ESX Server.

4.2.4 NFS storage network best practices

As a best practice, be sure to separate IP-based storage traffic from public IP network traffic by implementing separate physical network segments or VLAN segments. This design follows the architecture of SCSI and FC connectivity.

Creating a second network in VMware ESX requires the creation of a second vSwitch to separate the traffic on to other physical NICs. The VMware ESX Server requires a VMkernel port to be defined on the new vSwitch.

Each VMware ESX Server must have a service console port defined on the vSwitch that transmits public virtual machine traffic and on the vSwitch that is configured for IP storage traffic. This second service console port adds the redundancy in VMware ESX HA architectures and follows VMware ESX HA best practices.

With this design, do not allow routing of data between these networks (do not define a default gateway for the NFS storage network). With this model, NFS deployments require a second service console port be defined on the VMkernel storage virtual switch within each VMware ESX server.

The IP storage network, or VMkernel, connectivity can be verified using the **vmkping** command. With NFS-connected datastores, the following syntax tests connectivity:

```
vmkping <NFS IP address>
```

Tip: You can find general best practices for using NFS in a VMware vSphere environment in the *VMware Best Practices Guide* for use with NFS datastores. See “Best Practices for running VMware vSphere on Network Attached Storage”, available at this website:

http://www.vmware.com/files/pdf/techpaper/VMware-NFS-BestPractices-WP-EN.pdf

4.3 VMware vSphere and storage networking

This section briefly covers the use of VMware vSphere with an IBM N series system.

Reference: This section is an abstract of the Redbooks publication, *IBM System Storage N series and VMware vSphere Storage Best Practices*, SG24-7871. For current information, see the following website:

http://www.redbooks.ibm.com/abstracts/sg247871.html

4.3.1 Ethernet storage networking best practices

Use dedicated resources for storage traffic when possible. With Ethernet storage networks, this task can be achieved with separate physical switches or logically by implementing VLAN segments for storage I/O on a shared, switched IP infrastructure.

One of the challenges of configuring VMware ESX networking for IP storage is that the network configuration must meet all of the following goals:

- Be redundant across switches in a multiswitch environment.
- Use as many available physical paths as possible.
- Be scalable across multiple physical interfaces.

4.3.2 Virtual LANs (VLANs)

When segmenting network traffic with VLANs, interfaces either can be dedicated to a single VLAN or can support multiple VLANs with VLAN tagging.

For systems that have fewer NICs, such as blade servers, VLANs can be useful. Channeling two NICs together provides a VMware ESX server with physical link redundancy. By adding multiple VLANs, you can group common IP traffic on separate VLANs for optimal performance, as follows, for example:

- Group the Service console access with the virtual machine network on one VLAN.
- The VMkernel activities of IP Storage and vMotion can reside on a second VLAN.

VLANs and VLAN tagging also play a simple but important role in securing an IP storage network. NFS exports can be restricted to a range of IP addresses that are available only on the IP storage VLAN. N series storage systems also allow the restriction of the iSCSI protocol to specific interfaces, VLAN tags, or both.

These simple configuration settings have an enormous effect on the security and availability of IP-based datastores. If you are using multiple VLANs over the same interface, be sure that sufficient throughput can be provided for all traffic.

4.3.3 Flow control

Flow control is the process of managing the rate of data transmission between two nodes to prevent a fast sender from over running a slow receiver. Flow control can be configured for VMware ESX servers, N series storage systems, and network switches. Be sure to configure the end points: VMware ESX servers and N series storage systems with flow control set to Send ON and Receive OFF.

For network switches, set the switch ports connecting to VMware ESX hosts and N series storage systems to *Desired*, or if this mode is not available, set these ports to *Send OFF* and *Receive ON* (see Figure 4-3).

Tip: The switch ports are configured with the opposite settings of the end points, the VMware ESX and N series systems.

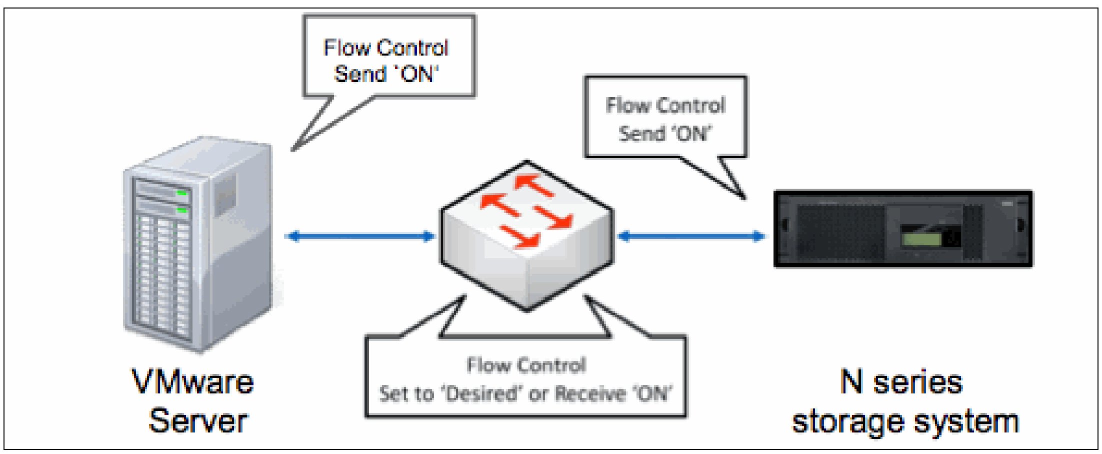


Figure 4-3 Configuring flow control setting

4.3.4 Spanning tree protocol

The *Spanning Tree Protocol (STP)* is a network protocol that makes sure of a loop-free topology for any bridged LAN. In the OSI model for computer networking, STP falls under the OSI layer-2. STP allows a network design to include spare (redundant) links to provide automatic backup paths if an active link fails, without the danger of bridge loops, or the need for manual enabling/disabling of these backup links. Bridge loops must be avoided because they result in flooding the network.

When connecting VMware ESX and N series storage systems to Ethernet storage networks, be sure that the Ethernet ports, which these systems connect to, are configured with either *Rapid Spanning Tree Protocol (RSTP)* or are portfast-disabled.

4.3.5 Bridge Protocol Data Unit (BPDU)

The *Bridge Protocol Data Unit (BPDU)* exchanges information about bridge IDs and root path costs within STP. When connecting VMware ESX and N series storage systems to

Ethernet storage networks, be sure that the Ethernet ports, which these systems connect to, are configured with BPDU disabled.

4.3.6 Virtual Interfaces

A Virtual Network Interface (VIF) is a mechanism that supports aggregation of network interfaces into one logical interface unit. After the VIF is created, it is indistinguishable from a physical network interface. VIFs provide fault tolerance of the network connection and, in certain cases, higher throughput to the storage device.

The N series enables the use of two types of load-balancing VIFs:

- Multimode VIF: This type is a static-configured Ethernet trunk. In a multimode VIF, all of the physical connections in the VIF are simultaneously active and can carry traffic. This mode requires that all of the interfaces be connected to a switch that supports trunking or aggregation over multiple port connections. The switch must be configured to understand that all the port connections share a common MAC address and are part of a single logical interface. In the event of a physical interface failure resulting in the loss of link, the VIF will automatically transmit traffic on the surviving links in the VIF without loss of connectivity.
- Dynamic multimode VIF: This type is an LACP-compliant (IEEE 802.3ad) VIF. In a dynamic multimode VIF, all of the physical connections are simultaneously active and carry traffic as with multimode VIFs, described previously. Dynamic multimode VIFs introduce the use of LACP signaling transmissions between the N series storage system and the remote switch. This signaling informs the remote channeling partner of link status. If a failure or inability to transmit data on a link is observed, the device identifying this problem informs the remote channeling partner of the failure, causing the removal of the interface from the VIF. This feature differs from standard multimode VIFs in that there is no signaling between channel partners to inform the remote partner of link failure. The only means for an interface to be removed from a standard multimode VIF is loss of link.

Multimode and dynamic multimode VIFs each use the same algorithm for determining load-balancing. This algorithm is based on source and destination IP or MAC address. Use IP-based source and destination load-balancing especially when the network is designed to route storage traffic. The reason is because during a transmission of a routed packet, a host transmits the packet to the default router IP address.

When arriving at the router, the router changes the MAC address of the routed packet to the MAC address of the local router interface on which the packet is transmitted out. The changing of the source MAC address can produce situations where traffic arriving from other subnets is always load-balanced to the same physical interfaces in the VIF. IP addresses are not changed unless Network Address Translation (NAT) is used. NAT is rarely used within the data center, where communications between VMware ESX hosts and N series systems occur.

In a single-mode VIF, only one of the physical connections is active at a time. If the storage controller detects a fault in the active connection, a standby connection is activated. No configuration is necessary on the switch to use a single-mode VIF, and the physical interfaces that make up the VIF do not have to connect to the same switch. Note that IP load balancing is not supported on single-mode VIFs.

Another possibility is to create second-level single or multimode VIFs. By using second-level VIFs, you can take advantage of both the link aggregation features of a multimode VIF and the failover capability of a single-mode VIF. In this configuration, two multimode VIFs are created, each one to a separate switch. A single-mode VIF is then created composed of the two multimode VIFs. In normal operation, traffic flows over only one of the multimode VIFs;

but if an interface or switch fails, the storage controller moves the network traffic to the other multimode VIF.

4.3.7 Ethernet switch connectivity

An IP storage infrastructure provides the flexibility to connect to storage in separate ways, depending on the needs of the environment. A basic architecture can provide a single non-redundant link to a datastore, suitable for storing ISO images, various backups, or VM templates. A redundant architecture, suitable for most production environments, has multiple links, providing failover for switches and network interfaces. Link-aggregated and load-balanced environments make use of multiple switches and interfaces simultaneously to provide failover and additional overall throughput for the environment.

More modern Ethernet switch models support *cross-stack EtherChannel trunks* or *virtual port channel trunks*, where interfaces on separate physical switches are combined into an IEEE 802.3ad EtherChannel trunk. The advantage of *multiswitch EtherChannel trunks* is that they can eliminate the need for additional passive links that are accessed only during failure scenarios in certain configurations.

All IP storage networking configuration options covered here use multiple switches and interfaces to provide redundancy and throughput for production VMware environments.

4.4 Best practices for using vSphere and RTCA

This section covers the best practices for using the IBM Real-time Compression Appliance (IBM RTCA product) with VMware vSphere environments. More specifically, we mean the use of the IBM RTCA product with NFS Datastores.

4.4.1 RTCA hardware setup best practices

To ensure high availability in a production environment, a single point of failure is unacceptable. Therefore we advise you to install a redundant high available pair of IBM Real-time Compression Appliances when you decide to implement the IBM RTCA product.

A high available pair of IBM Real-time Compression Appliances is never a true cluster. Both machines simply share information about which shares they compress. Thanks to the transparency and the way the IBM RTCA product compresses the data, this is enough for one IBM RTCA product to take over from another in case of a failure.

However, we do advise you to have a backup of each IBM RTCA configuration file stored in a safe place in case of disaster. This will simplify the restore of a failed IBM RTCA product.

Reference: For more information about how to implement the IBM Real-time Compression Appliance in your network environment, see 4.4.3, “Storage networking best practices” on page 45.

4.4.2 Storage controller best practices setup

To ensure high availability in a production environment, a single point of failure is unacceptable. Therefore we advise you to install a highly available controller pair.

An HA pair is two storage systems (nodes) whose controllers are connected to each other directly. You can configure the HA pair so that each node in the pair shares access to a common set of storage, subnets, and tape drives, or each node can own its own distinct set of storage. The nodes are connected to each other through an NVRAM adapter, or, in the case of systems with two controllers in a single chassis, through an internal interconnect. This allows one node to serve data that resides on the disks of its failed partner node. Each node continually monitors its partner, mirroring the data for each other's nonvolatile memory (NVRAM or NVMEM). See Figure 4-4.

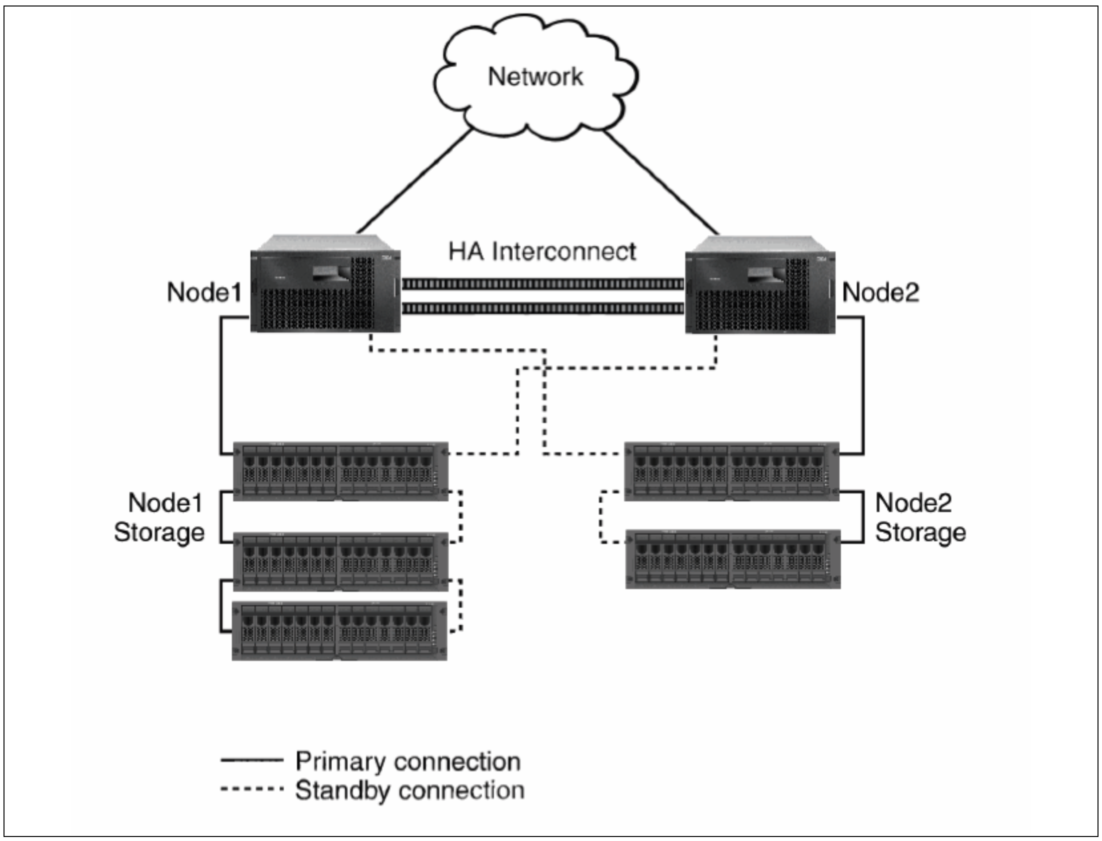


Figure 4-4 Standard HA pair configuration

In a standard HA pair, Data ONTAP functions so that each node monitors the functioning of its partner through a heartbeat signal sent between the nodes. Data from the NVRAM of one node is mirrored by its partner, and each node can take over the partner's disks or array LUNs if the partner fails. Also, the nodes synchronize each other's time.

Reference: For more information about HA cluster pairs with IBM N series systems, see Chapter 3, "Highly Available controller pairs," in the Redbooks publication, *IBM System Storage N series Hardware Guide*, SG24-7840. It is available at the following website:

http://www.redbooks.ibm.com/redbooks/pdfs/sg247840.pdf

4.4.3 Storage networking best practices

As a best practice, be sure to separate IP-based storage traffic from public IP network traffic by implementing separate physical network segments or VLAN segments.

Creating a second network in VMware ESX requires the creation of a second vSwitch to separate the traffic on to other physical NICs. The VMware ESX Server requires a VMkernel port to be defined on the new vSwitch.

Each VMware ESX Server must have a service console port defined on the vSwitch that transmits public virtual machine traffic and on the vSwitch that is configured for IP storage traffic. This second service console port adds the redundancy in VMware ESX HA architectures and follows VMware ESX HA best practices.

With this design, do not allow routing of data between these networks (do not define a default gateway for the NFS storage network). With this model, NFS deployments require a second service console port be defined on the VMkernel storage virtual switch within each VMware ESX server.

IP storage network, or VMkernel, connectivity can be verified using the **vmkping** command. With NFS-connected datastores, the following syntax tests connectivity:

```
vmkping <NFS IP address>
```

Because the IBM Real-time Compression Appliance does not support multiswitch EtherChannel trunking, and to ensure high availability, the network can be configured in the two ways (represented in Figure 4-6 on page 46 and Figure 4-7 on page 47). Each design requires the storage controller to have four physical network connections. Both designs are similar. They both provide multiple active links to each storage controller, provide a means to scale throughput by simply adding more links, and require multiple IP addresses per controller. Each design uses two physical links for each active network connection in order to achieve path high availability.

The multi layer VIF design

The multi-mode design (see Figure 4-5 on page 46) requires each storage controller to have at least four physical network connections. The connections are divided into two multimode (active-active) VIFs with IP load balancing enabled, one VIF connected to each of the two switches. These two VIFs are then combined into one single mode (active-passive) VIF. We refer to this configuration as a second-level VIF. This option also requires multiple IP addresses on the storage appliance. Multiple IP addresses can be assigned to the single-mode VIF by using IP address aliases or by using VLAN tagging.

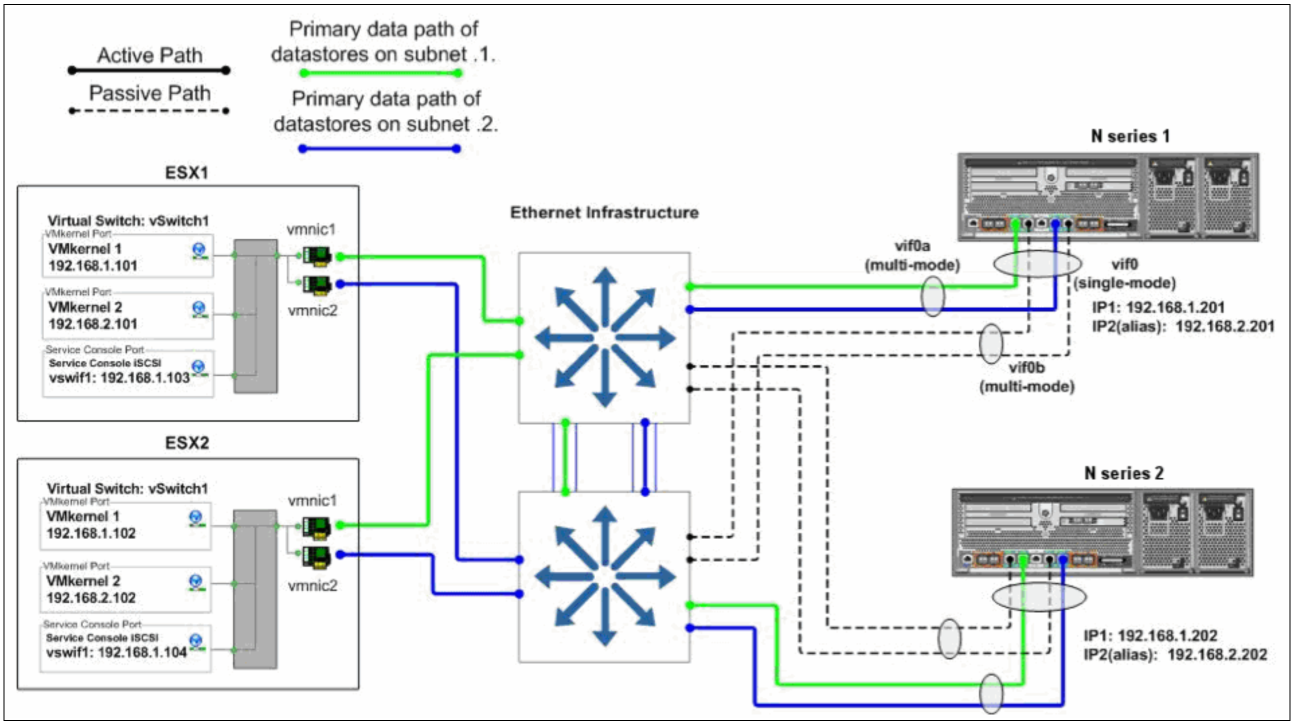


Figure 4-5 Storage-side Multimode VIFs

When using the multi-mode design together with the IBM Real-time Compression Appliance, your network will look like Figure 4-6.

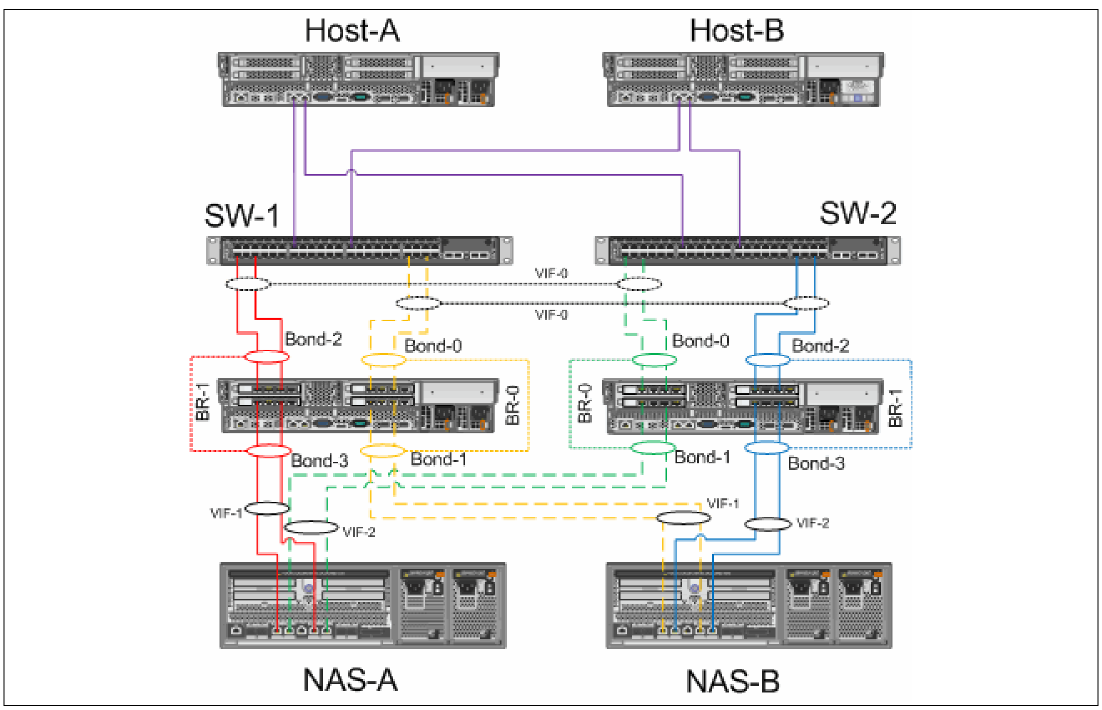


Figure 4-6 IBM RTCA implemented with multi-mode VIFS

The single layer VIF design

The single-mode design (Figure 4-7) requires each pair of network links to be configured as a single mode (active passive) VIF. Each VIF has a connection to both switches and has a single IP address assigned to it, providing two IP addresses on each controller. The **`vif favor`** command is used to force each VIF to use the appropriate switch for its active interface. This option is preferable because of its simplicity and the lack of any special configuration in the network switches.

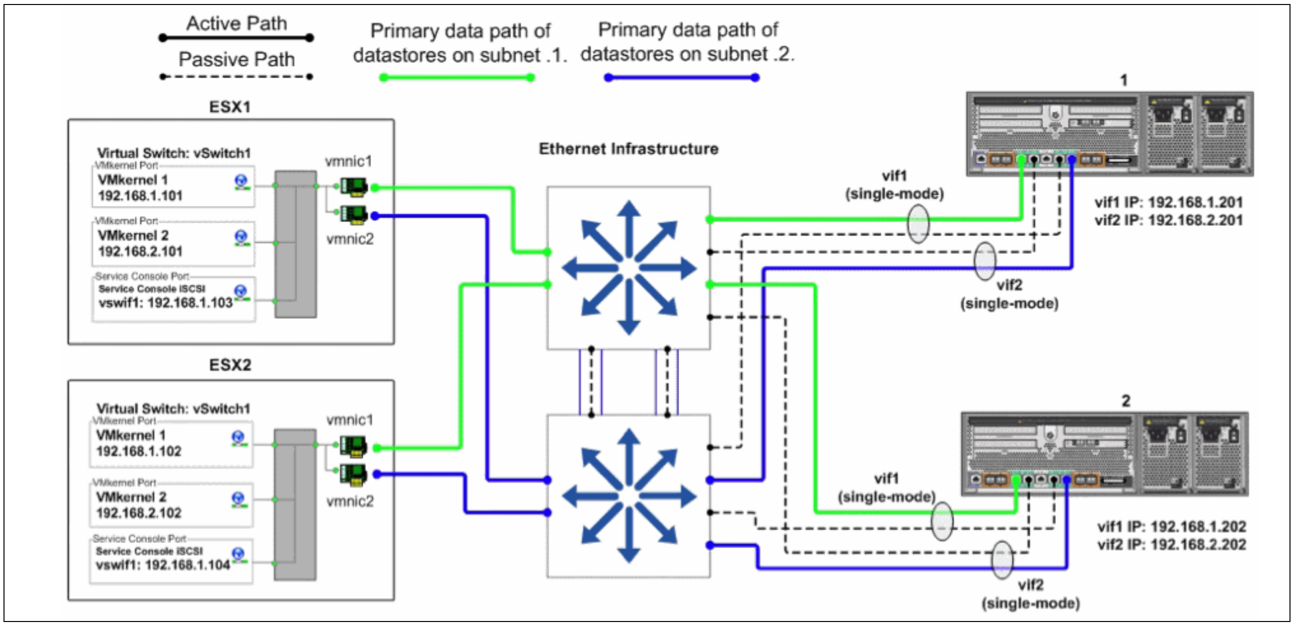


Figure 4-7 Single mode VIFs

> **Reference:** For more information about VMware best practices on IBM N series storage, see the Redbooks publication, *IBM System Storage N series and VMware vSphere Storage Best Practices,* SG24-7871. It is available at the following website:
>
> `http://www.redbooks.ibm.com/abstracts/sg247871.html`

4.4.4 Virtual machine guest file placement

Virtual machines (VMs) consist of multiple files on datastores. To ensure the best compression and deduplication ratios, we have some file placement advice.

Virtual machine files include these:

- .vmx file
- .vmdk file
- .vswp file
- .log file

Our advice is to keep all files together except for the .vswp file. This file is the swap file for the virtual machine. This machine must be placed on a separate datastore on which you do not use snapshots and deduplication. The reason is that this file will never be the same for any of the virtual machines and will change all the time if used. However, the datastore on which the .vswp file resides can be compressed to save space.

To change the .vswp file location:

1. Connect to the ESX(i) server or to the vCenter server using the VMware vSphere Client.
2. Select the server and go to the **Configuration** tab.
3. Select **Virtual Machine Swapfile Location** and click **Edit**.
4. Specify the datastore that you want to use for the .vswp files.
5. The new location will be used when the virtual machine restarts.

4.4.5 Block alignment with NFS

When not using the IBM Real-time Compression Appliance, block alignment is a serious consideration. Following is a short description of block alignment, which involves alignment of VM partitions and VMFS to storage arrays.

Virtual machines store their data on virtual disks. As with physical disks, these virtual disks contain storage partitions and file systems, which are created by the VM's guest operating system. To make sure of optimal disk I/O within the VM, align the partitions of the virtual disks to the block boundaries of VMFS and the block boundaries of the storage array. Failure to align all three items can result in a dramatic increase of I/O load on a storage array and negatively affects the performance of all virtual machines being served on the array.

Be sure that the partitions of VMs and the partitions of VMFS datastores are to be aligned to the blocks of the underlying storage array. For more information about VMFS and GOS file system alignment, see the following documents:

- IBM: *Storage Block Alignment with VMware Virtual Infrastructure*, NS3593
- VMware: *Recommendations for Aligning VMFS Partitions*

The IBM Real-time Compression Appliance tries to compress all blocks, and the compression rate of all blocks will be different for every block. This implies that block alignment is a setting that does not need to be considered anymore when using the IBM RTCA.

4.4.6 Considerations regarding deduplication and snapshots

When using deduplication and snapshots, there are considerations to take into account before implementing the IBM Real-time Compression Appliance.

Considerations for deduplication

When implementing compression on an NFS datastore that is using deduplication, we have certain considerations. When we run the Compression Accelerator daemon, all data will be rewritten on to the volume. This implies that all existing data will disappear and new compressed data will be written. Because all data disappears, the deduplication ratio will disappear as well. In Example 4-1 on page 49 we show the `df -Sk` command before and after the compression accelerator has run.

Example 4-1 Command df -Sk before and after compression accelerator

```
Before CA
N5600-A> df -Sk
Filesystem               used      compressed          a-sis     %saved
/vol/vmware1/      19047572KB             0KB     21415064KB        53%

After CA
N5600-A> df -Sk
Filesystem               used      compressed          a-sis     %saved
/vol/vmware1/      11300908KB             0KB           12KB         0%
```

As you can see, the percentage saved is reduced to 0% but the used space has lowered by about 40%.

Because the data is all new, you must rerun the command **`sis start -s <path to the volume>.`** This command will force the deduplication process to rescan all the data on the volume. After you have started the command, you can monitor the progress of the deduplication process by running the command **`sis status`** (Example 4-2).

Example 4-2 Restarting the deduplication process and monitoring it

```
N5600-A> sis start -s /vol/vmware1
The file system will be scanned to process existing data in /vol/vmware1.
This operation may initialize related existing metafiles.
Are you sure you want to proceed (y/n)? y
The SIS operation for "/vol/vmware1" is started.
[N5600-A: wafl.scan.start:info]: Starting SIS volume scan on volume vmware1.

N5600-A> sis status
Path                           State      Status     Progress
/vol/vmware1                   Enabled    Active     11 GB Scanned

N5600-A> sis status
Path                           State      Status     Progress
/vol/vmware1                   Enabled    Active     10 GB Searched

N5600-A> sis status
Path                           State      Status     Progress
/vol/vmware1                   Enabled    Active     235 MB (6%) Done

N5600-A> sis status
Path                           State      Status     Progress
/vol/vmware1                   Enabled    Idle       Idle for 00:01:21
```

Attention: Make sure you have enough free space in your volume when turning on compression and when running the Compression Accelerator daemon on volumes that have deduplication turned on.

Considerations for snapshots

When using the IBM Real-time Compression Appliance on volumes with snapshots, one consideration is that all data written through the IBM Real-time Compression Appliance and all data compressed by the Compression Accelerator daemon will be seen as new data.

When new data is written on volumes with snapshots, the old data will reside in the snapshot space, which can fill up quite quickly. This must be considered when turning on compression on volumes with snapshots. Make sure you have enough free space in your volume when turning on compression and when running the Compression Accelerator daemon on volumes that have snapshots.

4.5 Related documents

To complete the installation of VMware for use with NFS datastores, and more specifically with IBM System Storage N series, you can consult the following documents.

- *IBM System Storage N series and VMware vSphere Storage Best Practices*, SG24-7871

  `http://www.redbooks.ibm.com/abstracts/sg247871.html`
- *IBM System Storage N series Hardware Guide*, SG24-7840

  `http://www.redbooks.ibm.com/abstracts/sg247840.html`
- *IBM System Storage N series with VMware ESX Server*, SG24-7636

  `http://www.redbooks.ibm.com/abstracts/sg247636.html`

5

Oracle database application integration

In this chapter, we describe Oracle database integration using RTCA. IBM and Oracle have worked over the past several years to validate Oracle products on N series storage devices and a range of server platforms.

5.1 Foundational best practices

It is important to realize that an RTCA implementation builds on the best practices that relate to three aspects: the NAS storage controller, NAS client platforms, and Oracle database application.

Consider the following related best practices in the deployment of the NAS environment prior to the implementation of the RTCA product:

- Best practices guidelines - *IBM System Storage N series and Oracle Database 11g, Technical report NS3633-1*:

  https://www-304.ibm.com/partnerworld/wps/servlet/ContentHandler/whitepaper/systems/oracle/best_practice

  This document describes best-practice guidelines for running Oracle Database 11g on IBM System Storage N series storage systems with platforms, such as Solaris, Hewlett Packard HP/UX, IBM AIX, Linux and Microsoft Windows. It represents the best-practice minimum requirements for deployment of Oracle on IBM System Storage N series. Intended to be a guideline of proven methods and techniques.

- Red Hat Enterprise Linux Protocol Performance Comparison with Oracle Database 11g Release 2:

  http://media.netapp.com/documents/tr-3932.pdf

  This technical report compares the performance of different network protocols and file systems and how they interact with Oracle Database 11g Release 2 workloads running on Red Hat Enterprise Linux 5 U4 using Gigabit Ethernet (GbE), 10 GbE, 10 Gb FCOE, and 4 Gb/8 Gb Fibre Channel (FC).

This section describes the best practices for deploying the RTCA product in Linux RHEL using an Oracle Database environment on NFS exported from N series.

5.2 Oracle database on Network Attached Storage

You must choose carefully when deciding on your infrastructure because you will use it for some time. Making changes might require changing the disk systems, which in turn requires data migration.

The following topics provide information that will be useful initially when installing a new Oracle database or migrating an Oracle database from direct-attached storage (DAS), either to Network Attached Storage (NAS) or Storage Attached Network (SAN).

In the following section, we only focus on Network Attached Storage.

5.2.1 Network Attached Storage

NAS uses two kind of network protocol types: NFS and CIFS.

NFS consolidation

One common consolidation technique uses network attached storage (NAS) and the network file system (NFS). NFS is becoming a common and critical component in successful enterprise database deployments. NFS provides extremely effective storage virtualization while utilizing existing infrastructure and management facilities.

NFS has a long and successful history in non-database environments. However, it is not traditionally viewed as a strong candidate for database deployments. This view is a result of perceived performance problems in database deployments that use NAS.

There are some technical documents that counter the traditional view by providing an investigation of database performance with NAS and a roadmap to successfully deploy a relational database management system (RDBMS) with NFS. Specific guidelines are provided for deploying Oracle on NFS along with tips and techniques for analyzing performance. For details, see "*Red Hat Enterprise Linux Protocol Performance Comparison with Oracle Database 11g Release 2.*"

Storage solution requirements

A successful solution for database storage has several important properties:

- Storage virtualization: Storage configuration, management, and access must be handled almost completely by the storage system.
- Complexity and cost reduction: Controlling the cost and complexity of deploying and managing a storage solution is a primary concern in most IT environments.
- Rapid application deployment: Rapid deployment and simplified support for a wide variety of applications are essential.
- Grid support: There is a clear trend toward grid-aware applications that can take advantage of emerging "server blade" environments, such as Oracle RAC. A successful storage solution must accommodate current and future grid deployments.

NFS fulfills these requirements and provides a number of additional benefits over traditional storage access models. NFS not only simplifies data transport but also reduces the management complexity of large storage installations. NFS also offloads the physical I/O processing from the host system to NAS. This can be a real benefit over systems that require the host system to consume CPU cycles to handle all I/O functionality. This offloading can translate into more efficient resource usage.

5.3 Oracle, N series, and RTCA compression comparison

Several compression technologies exist for NAS storage. and Oracle itself provides a compression feature. Native compression must be evaluated before applying an additional compression technology. Each of these features has its strengths, and the best compression solution for each environment might differ.

5.3.1 Oracle Advanced Compression

In recent years, databases have experienced explosive growth in size. Oracle Database 11g addresses this issue with a new Oracle Advanced Compression feature that works with all data types (regularly structured [numbers, characters], unstructured [documents, spreadsheets, XML and other files] and backup data).

Advanced Compression in Oracle Database 11g not only reduces disk space requirements for all types of data, it also improves application performance and enhances memory and network efficiency. In addition, it can be used for any type of application without application changes.

Advanced Compression in Oracle Database 11g has the following new features:

- OLTP table compression: This feature allows structured or relational data to be compressed during all types of data manipulation operations, including regular INSERT, UPDATE, or DELETEs. This new feature leverages a sophisticated and intelligent algorithm that minimizes the compression overhead during write operations, thereby making it viable for all application workloads. Additionally, it significantly improves performance of queries by reducing disk I/O and improving memory efficiency. Previous Oracle Database releases supported compression for bulk data-loading operations commonly used for data warehousing applications. Oracle Database 11g OLTP table compression improves database performance with more effective use of memory for caching data and reduced I/O for table scans. With OLTP table compression, you can achieve two-fold to three-fold compression ratios with minimal processing overhead.
- Fast files deduplication: This feature is intelligent technology that eliminates duplicate copies of files stored in Oracle Database 11g. Besides reducing storage footprint, this feature also dramatically improves the performance of write and copy operations involving duplicate content.
- Fast files compression: This feature compresses the unstructured or file data stored within the database. Two levels of compression are available so you have a choice of higher compression by using additional system (CPU) resources.
- Backup data compression: The storage requirements for maintaining database backups and backup performance are directly affected by database size. To that end, advanced compression includes compression for backup data when you employ Recovery Manager (RMAN) or Oracle Data Pump for database backups.
- Network traffic compression: Advanced compression offers the capability to compress Oracle Data Guard (standby databases) redo data as Data Guard resolves redo gaps. This improves the efficiency of network utilization and speeds up gap resolution.

5.3.2 N series / Netapp compression

Netapp data compression is a software-based solution that provides transparent data compression. No application changes are required to use Netapp data compression.

How data compression works

Netapp data compression does not compress the entire file as a single contiguous stream of bytes. This is prohibitively expensive when it comes to servicing small reads or overwrites from part of a file because it requires the entire file to be read from disk and uncompressed before the request can be served. This will be especially difficult on large files. To avoid this, Netapp data compression works by compressing a small group of consecutive blocks, known as a compression group. In this way, when a read or overwrite request comes in, we only need to read a small group of blocks, not the entire file. This optimizes read and overwrite performance and allows greater scalability in the size of the files being compressed.

When data compression runs

Netapp data compression can be run in-line or post-process and also includes the ability to perform compression of existing data.

In-line operations

Netapp data compression can be configured as an in-line operation. In this way, as data is sent to the storage system, it is compressed in memory before being written to the disk. The advantage of this implementation is that it can reduce the amount of write I/O. This implementation option can affect your write performance and thus must not be used for performance-sensitive environments that without proper testing to understand the impact.

To provide the fastest throughput, in-line compression will compress most new writes but will defer some more performance-intensive compression operations to compress when the next post-process compression process is run. An example of a performance-intensive compression operation includes partial compression group writes and overwrites.

Post-process operations

Netapp data compression includes the ability to run post-process compression. Post-process compression uses the same schedule as deduplication utilizes. If compression is enabled when the **`sis schedule`** initiates a post-process operation, it runs compression first, followed by deduplication. It includes the ability to compress data that existed on disk prior to enabling compression.

If both in-line and post-process compression are enabled, then post-process compression will try to compress only blocks that are not already compressed. This includes blocks that were bypassed by in-line compression, such as small partial compression group overwrites.

5.3.3 RTCA compression

RTCA implementation has already been described in detail in the previous chapters.

Intuitively, inserting a compression appliance in the data path might seem to introduce risk of negatively impacting performance. In reality, customer experience and performance benchmarks illustrate that storage system efficiency is increased, and performance is either maintained, or even increased, when the IBM Real-time Compression Appliance is utilized.

5.3.4 Comparison between compression features

Each compression feature has pluses and minuses that you must take into consideration before deciding which one is the more suitable to implement. Table 5-1 on page 56 shows a brief comparison.

Table 5-1 Comparison between each compression feature

Oracle compression	Netapp compression	RTCA
Consideration is more on the overhead of the server's CPU limit reached.	Consideration is more on the overhead of the Netapp CPU limit reached.	Appliances act as "smart cables" to perform compression processing.
Included as a new feature since Oracle Database 11g.	Must be on 64-bit aggregate in Data ONTAP 8. Free compression license.	Charge for new appliances and more devices to be maintained. No CPU overhead on the server or storage side. RTCA also speeds up the I/O performance (depends on the configuration).

> **Tip:** For every implementation, always consider the system as a whole (end to end), for CPU overhead, reduced I/O, compression ratio, network traffic, and so on.

Oracle compression

As your database grows in size, consider using table compression. Compression saves disk space, reduces memory use in the database buffer cache, and can significantly speed query execution during reads. Compression has a cost in CPU overhead for data loading and DML. However, this cost can be offset by reduced I/O requirements.

N series / Netapp compression feature

Choosing when to enable compression involves balancing the benefits of space savings against the potential overhead. Your savings and acceptable overhead will vary depending on the use case, and, as such, you might find some solutions suited for primary tier and others better suited for backup/archive tier only.

The amount of system resources they consume and the possible savings are highly dependent upon the type of data. Performance impact will vary in each environment, and Netapp highly recommends that the performance impact be fully tested and understood before implementing in production.

Table 5-2 shows some examples of where to consider adding compression. These are strictly examples, not rules. Your environment might have different performance requirements for specific use cases. Netapp highly recommends that the performance impact be fully tested and understood before you decide to implement in production.

Table 5-2 When to consider adding compression

Type of application	Storage efficiency to consider
Backup/Archive	In-line Compression + Post-Process Compression
Test/Development	In-line Compression + Post-Process Compression
File Services/IT Infrastructure, Engineering Data	In-line Compression + Post-Process Compression
Geosiesmic	In-line Compression Only (set post-process schedule to none)
Oracle OLTP	Maybe[a]
Oracle Data Warehouse	Maybe[a]
Exchange 2010	Maybe[a] Post-Process Compression

a. Maybe: These are use cases that can be considered but only if there is sufficient time to run compression processes and there are sufficient savings. Testing must be done before implementing in production. Netapp recommends a Flash Cache card.

Tip: These are guidelines, not rules. They assume that the savings are high enough, your system has sufficient system resources, and any potential effect on performance is fully understood and acceptable. Especially, this is the case when enabling in-line compression including during peak hours.

In-line compression provides immediate space savings; post-process compression first writes the blocks to disk as uncompressed and then at a scheduled time compresses the data. Post-process compression is useful for environments that want compression savings but don‘t want to incur any performance penalty associated with new writes. In-line compression is useful for customers who are not as performance sensitive and can handle some impact on new write performance and CPU during peak hours.

5.4 New features on Oracle 11g relevant to RTCA

Oracle Database 11g is a new major release of the Oracle Database with many new features. Some of these features are relevant for RTCA implementations.

5.4.1 Data segment compression

Data compression, Figure 5-1, increases the number of rows stored per physical data block. Enable data segment compression to reduce disk and cache usage.

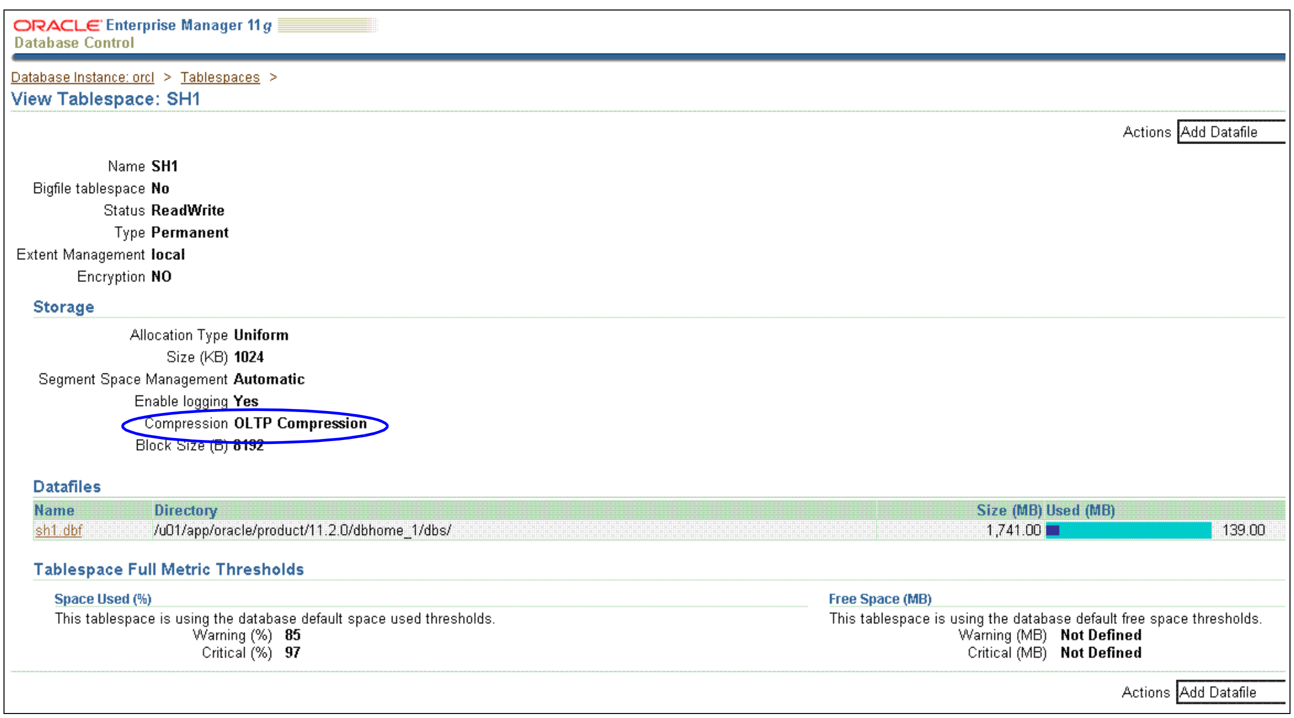


Figure 5-1 Compression on Oracle table space

You can specify compression for a table space, a table, or a partition. If specified at the table space level, then all tables created in that table space are compressed by default. See Figure 5-1 on page 57 for an example of Oracle Table space Compression.

5.4.2 Oracle compression options

The Oracle compression options are:

- No Compression
- Basic Compression
- OLTP Compression
- Data Warehouse Compression
- Online Archival Compression

See Figure 5-2 for a display of compression options.

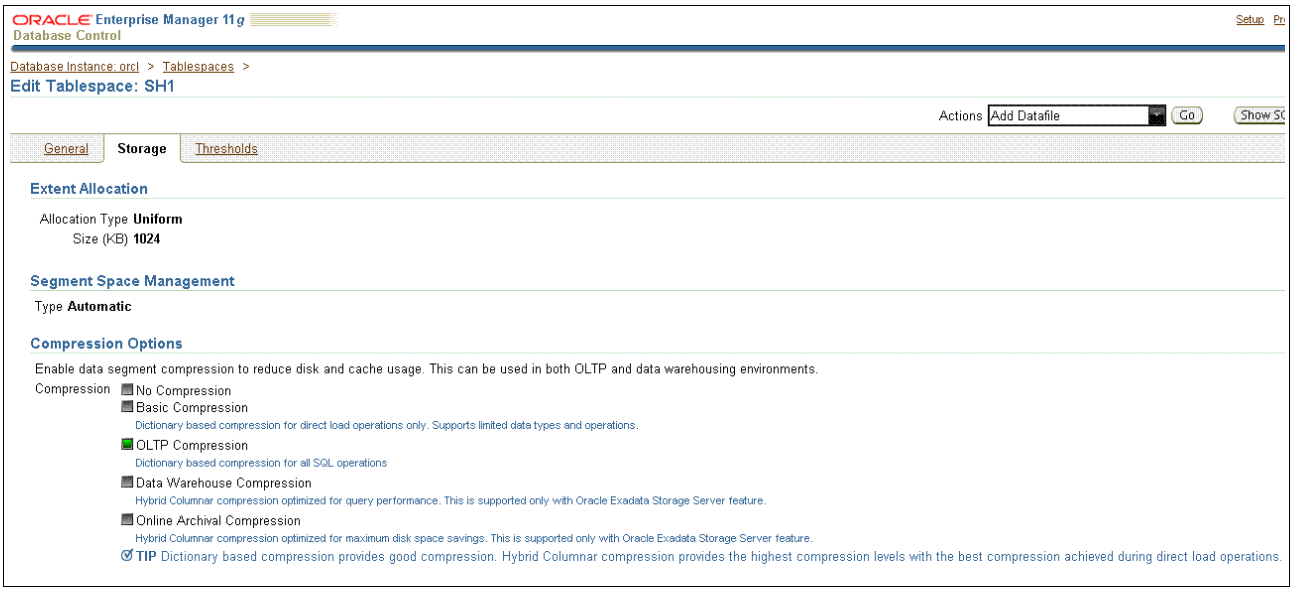


Figure 5-2 Oracle compression options for table space

5.5 Best practices

This section provides information for Linux on Oracle database using NFS on N series storage.

5.5.1 Storage network configuration

Using jumbo frames can improve performance in environments where Linux NFS clients and NAS systems are together on an unrouted network. Be sure to consult the command reference for each switch to make sure it can handle jumbo frames. If possible, set the Maximum Transmission Unit (MTU) size of 9,000 for all storage interfaces on the host, for all interfaces on the N series controllers, and for the ports involved on the switch.

If possible, to take advantage of all 1 GbE interfaces, the best practice is to create two separate LACP bonds. Each bond contains two individual interfaces. Connect these interfaces to a single ifgrp (vif), which bonds four 1 GbE interfaces on the storage side.

5.5.2 NFS mount options for Oracle databases

Table 5-3 and Table 5-4 show the mount options for Oracle databases on N series using NFS.

Table 5-3 Oracle 10g (R1,R2), 11g RAC with Oracle CRS clusterware

OS	Mount options for binaries	Mount options for Oracle data files	Mount options for OCR and CRS voting files	Mount options for ADR HOME (11g only)	init.ora parameters
Solaris	[common], noac,nointr, proto=tcp, suid	[common], forcedirectio, noac,nointr, proto=tcp,suid	[common], forcedirectio, noac,nointr, proto=tcp	[common], actimeo=0, nointr, proto=tcp, suid	filesystemio _options= setall
AIX	[common], nointr,timeo=600 , proto=tcp	[common], cio,noac, nointr,timeo=600, proto=tcp	[common], cio,intr,noac, timeo=600, proto=tcp	[common], nointr, timeo=600, proto=tcp	filesystemio _options= setall
HPU	[common], noac,nointr, timeo=600, proto=tcp,suid	[common], forcedirectio, noac,nointr, timeo=600, proto=tcp, suid	[common], forcedirectio, noac,nointr, timeo=600, proto=tcp,suid	[common], actimeo=0, nointr, timeo=600, proto=tcp, suid	filesystemio _options= setall
Linux	[common], actimeo=0, nointr, suid, timeo=600, tcp	[common], actimeo=0, nointr, suid, timeo=600, tcp	[common], actimeo=0, nointr, suid, timeo=600, tcp	[common], actimeo=0, nointr, suid, timeo=600, tcp	filesystemio _options= setall or directio

Table 5-4 Oracle 10g (R1,R2) non-RAC, Single Instance (SI)

OS	Mount options for binaries	Mount options for Oracle data files	Mount options for OCR and CRS voting files	Mount options for ADR HOME (11g only)	init.ora parameters
Solaris	[common], nointr, proto=tcp, suid	[common], [forcedirectio or llock], nointr, proto=tcp, suid	N/A	[common], nointr, proto=tcp suid	filesystemio _options= setall
AIX	[common], intr,timeo=600, proto=tcp	[common], cio,intr, timeo=600, proto=tcp	N/A	[common], nointr, timeo=600, proto=tcp	filesystemio _options= setall
HPU	[common], nointr,timeo=600 , proto=tcp	[common], nointr, timeo=600, proto=tcp, suid	N/A	[common], nointr, timeo=600, proto=tcp, suid	filesystemio _options= setall
Linux	[common], nointr,timeo=600 , tcp	[common], nointr, timeo=600, tcp	N/A	[common], nointr,suid, timeo=600, tcp	filesystemio _options= setall or directio

```
[common] = rw,bg,hard,rsize=32768,wsize=32768,vers=3
[common] = These mount options must be used in addition to the ones in the
matrix above.
```

For read/write size for all UNIX-based operating systems, rsize=65536 and wsize=65536 can also be used. Using these settings enabled us to take better advantage of larger TCP window sizes, which are now available on the storage controllers.

5.5.3 Data ONTAP tuning options

There are several tuning options that must be implemented if using N series or Netapp NAS.

This increases the size of the TCP receive window for NFS. The TCP receive window on the client must be adjusted to match the size of the NetApp controller because TCP reduces the size of the larger TCP receive window to match the size of the smaller TCP window:

`nfs.tcp.recvwindowsize 262144`

We found that it was necessary to increase the default value of 26,280 to realize the performance increase of jumbo frames.

`nfs.tcp.xfersize 65536`

This is the default setting for this option. Make sure that it is not reduced. This option controls the TCP transmit window size for NFS.

`setflag wafl_max_write_alloc_blocks 256`

We set one additional flag in the /etc/rc file of each storage system. This flag optimizes the WAFL (Write Anywhere File Layout) on-disk data layout.

5.5.4 LINUX tuning options

Here are some Linux tuning parameters that must be implemented:

`sunrpc.tcp_slot_table_entries = 128`

Increasing this parameter from the default of 16 to the maximum of 128 increases the number of in-flight Remote Procedure Calls (I/Os).

Be sure to edit /etc/init.d/netfs to call /sbin/sysctl –p in the first line of the script so that sunrpc.tcp_slot_table_entries is set before NFS mounts any file systems. If NFS mounts the file systems before this parameter is set, the default value of 16 is implemented.

5.5.5 Storage layout configuration

The Oracle data files must be distributed evenly across both N series or NetApp storage controllers in the oradata_nfs and oradata2_nfs volumes. The Oracle online redo logs must be balanced across the two controllers. The primary redo log members can be stored on the first controller, and all multiplexed redo log members are stored on the second controller. See the illustration in Figure 5-3.

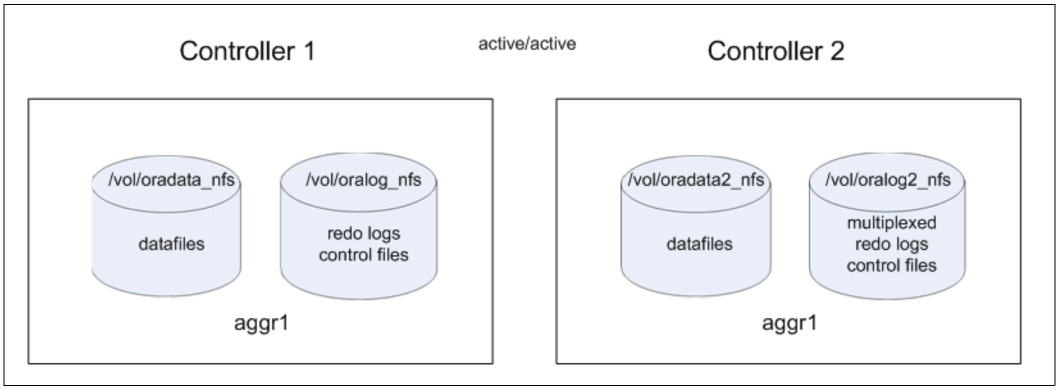


Figure 5-3 Illustration for N series storage layout

6

File serving solutions

This chapter describes the best practices for deploying the RTCA product into a CIFS or NFS file service environment.

Tip: This chapter uses IBM System Storage N series as the NAS system. The different solution and design approaches can be easily applied to other NAS solutions that are available in the marketplace.

6.1 Foundations

It is important that an RTCA implementation builds on the best practices of both the NAS storage controller and NAS client platforms.

Consider the following related best practices in the deployment of the NAS environment prior to the implementation of the RTCA product.

6.1.1 NAS storage controllers

See the best practices guides for your NAS storage controller:

- IBM N series:
  - *IBM System Storage N series Hardware Guide*, SG24-7840
  - *IBM System Storage N series Software Guide*, SG24-7129
  - *Setting up CIFS and Joining the Active Directory*, REDP-4074
  - *Using the Linux NFS Client with IBM System Storage N series*, SG24-7462
  - *IBM System Storage N series Multiprotocol Use Guide*, SG24-7469
  - *IBM Storage System N series Antivirus Scanning Best Practices Guide*, REDP-4084
- NetApp:
  - Enterprise File Services Best Practices Index:

    http://now.netapp.com/NOW/knowledge/docs/bpg/efs/
  - Data ONTAP 7.3 File Access and Protocols Management Guide:

    http://now.netapp.com/NOW/knowledge/docs/ontap/rel733/pdfs/ontap/filesag.pdf
- EMC Celerra:
  - See the following product documentation:

    https://powerlink.emc.com/

6.1.2 NAS clients

See the best practices guides for your NAS client platform:

- Windows (CIFS):
  - Windows File Services Best Practices with NetApp Storage Systems:

    http://www.netapp.com/us/library/technical-reports/tr-3771.html
- UNIX (NFS):
  - AIX:

    http://publib.boulder.ibm.com/infocenter/aix/v6r1/index.jsp?topic=%2Fcom.ibm.aix.commadmn%2Fdoc%2Fcommadmndita%2Fnfs_intro.htm
  - Linux:

    http://nfs.sourceforge.net/nfs-howto/
  - Solaris:

    http://download.oracle.com/docs/cd/E19455-01/806-0916/6ja8539fd/

6.2 Common file service requirements

This section includes information about possible file service requirements.

> **WARNING:** Never add the NAS root file system (for example, /vol/vol0 or \C$ in the IBM N series product) to the compression accelerator task list. This volume contains the NAS system files, which are read directly by the NAS system.
>
> **Reason:** The NAS controller is rendered inoperable if its system files are compressed by the RTCA product.

6.2.1 Workload considerations

This section discusses important practical implications when using compression. The behaviors listed here are common for compression algorithms and are not specific to the RTCA product,

Very small files

Overwriting small files (NFS files smaller than 64K or CIFS files smaller than 122 KB) might increase the compressed file size up to 121 KB.

Very large files

When a compressed file is opened, its metadata is loaded into memory. In larger files this might take multiple seconds before the appliance will reply to the open operation.

Compressed files greater than 50 GB in size might take multiple seconds to load.

Incompressible file types

When compressing files that contain incompressible data, this can be example encrypted data or already compressed data, such as mp3 or mpeg files, the compressed file size might exceed the original file's size in some cases.

> **Attention:** Files with an uncompressible first block (32 KB in NFS, 61 KB in CIFS) cannot be compressed by the Compression Accelerator. Such files will be skipped by the Compression Accelerator.

6.2.2 Compression filters

As previously discussed, to improve the handling of impressible data, you can use compression filters to improve file handling.

File name extension filters

File name extension filters only pertain to the file name's last two extensions. Multiple extensions separated by a period are not supported, for example, if “gz” is configured as a filter for not compressing, files with tar.gz and gz.tar extensions are filtered too.

Directories to ignore

This setting does not support directory names containing a space character. It is impossible to exclude directories that include a space in the path name.

This setting is also case-sensitive. The list of directories to ignore in Compression Accelerator is case-sensitive, even when running a task to compress a CIFS share. For example, even if the directories to ignore list will include “old”, a directory named “Old” is compressed.

Be patient

Newly added compression filters cannot be compressed immediately. After adding a new compression filter, wait 120 seconds before adding this filter in the Compression Accelerator full scan configuration page.

6.3 Planning a CIFS file service

Before implementing RTCA, you must gain a good understanding of your current IT infrastructure. Because CIFS is mostly linked to Microsoft Windows environments, it is important to understand your Active Directory, NT4 Domain, or Workgroup structure.

The RTCA product is fully compatible with clients that conform to the Common Internet File System (CIFS) access protocol, which is also known as Server Message Block (SMB).

> **Attention:** Support for the SMB2 protocol was introduced in Version 3.8 of the RTCA product. Prior to that, the SMB2 protocol was unsupported and was failed back to SMB1 by the compression appliance.

6.3.1 Authentication and authorization

The RTCA product does not participate in the security access negotiation between the CIFS client and the NAS file server. All user access is configured on the underlying NAS storage controller, exactly as though the RTCA product was not in the data path

Although the RTCA product does support connection to an LDAP server, that is for administrative authentication only, and it plays no part in client access to data.

CIFS signing

SMB Signing is a feature that is designed to prevent the man-in-the middle attack. Because the compression appliance is an in-line product, the SMB Signing mechanism perceives the appliance as a man-in-the-middle. Thus the SMB Signing feature for the domain controller must be turned off when using the compression appliance.

Listing the available shares

The RTCA product needs to be able to list the CIFS shares that are being presented by the NAS storage controller.

Access based enumeration (ABE)

This option, on the NAS controller, limits the displayed contents of a share to only those files and directories that the user has permission to access. This is different than the default case, where the full content of a share is visible to the user. For example, if a share contains 100 subdirectories, but the user only has file system permission for five of them, then only those five directories will be visible. This helps to reduce confusion when multiple users access a common CIFS share.

In normal operations, both transparent and compressed mode, the RTCA does not present a user identity of its own; instead, the user that generated the request is being used, so the file system access, including ABE filtering, will perform as expected.

However, when processing a CIFS share, the Compression Accelerator must be given a username and password to compress the existing data. This must be an administrative account so that any ABE filtering does not interfere with the compression process.

Access based share enumeration (ABSE)

This option, available on some NAS controllers, is used to limit the display of shares to only those that the user has permission to access. This is different than the default case, where the full list of shares is visible to all users. For example, if the NAS controller has 100 shares, but the user only has share permission for five of them, only those five shares will be visible. This helps to reduce confusion when a user only has access to a subset of the NAS shares.

In normal operations, both transparent and compressed mode, the RTCA uses its predefined CIFS credentials to list the available shares. This must be an administrative account so that any ABSE filtering does not interfere with the share enumeration.

However, the compression accelerator can only process a share if it is enumerated and is visible in the GUI interface.

Note that even if a share is not visible to the RTCA product, the share will still be visible to the clients, and the share contents can still be compressed.

On the IBM N series platform, an example of ABSE is the *auto home share* feature. In this mode, the user's home share are dynamically created when accessed by the users, removed when idle, and are invisible to other users, including the NAS administrator account. This can present a challenge when using the RTCA to compress their existing data.

6.3.2 Auto home shares

The IBM N series (and NetApp FAS) storage controllers can dynamically create a user home share when accessed. When the user is not connected to the NAS, that home share will not exist, though of course the data still exists in the NAS storage volume. This both simplifies the configuration and reduces the load on the NAS controller.

```
cifs.home_dir_namestyle
cifs.home_dirs_public_for_admin on
```

The following steps will help you to perform the task:

1. Define the parent location of the user's home directories.
2. Specify the naming style of the home directories.
3. Create the user's individual home directories.
4. The home shares will now be dynamically created when accessed.

> **Tip:** The CIFS shares command does not display the list of auto home shares.

6.4 Planning an NFS file service

Most UNIX clients use NFS for remote file access. Sun Microsystems introduced NFS in 1985. Since then, it has become a de facto standard protocol, used by 10 million systems worldwide. NFS is particularly common on UNIX-based systems, but NFS implementations are available for virtually every modern computing platform in current use, from desktops to supercomputers. Only when used by UNIX-based systems, however, does NFS closely resemble the behavior of a client's local file system. NFSv4 addresses some weaknesses in earlier NFS versions such as ACL, security, and file system namespace, and so on. However, this is beyond the scope of this book.

The compression appliance is fully compatible with clients that conform to the NFSv3 protocol standards. NFSv3 is supported over TCP and UDP.

> **Support:** The NFSv2 and NFSv4 protocols are *unsupported* and pass through the compression appliance transparently. After which they can be used to access data on the underlying NAS storage controller, but they cannot correctly read any data that has been compressed by the RTCA product.

6.5 Block-mode pass-through

Although the RTCA product is designed to provide compression to an NAS environment (NFS and CIFS), it is possible that the underlying NAS controller will also present storage by one of the block-mode protocols.

6.5.1 FCP protocol access

Because the FC protocol is transported over a completely separate physical media (fibre optic cable) there is no interaction with the RTCA product.

Because the RTCA product compresses the NAS data, reducing the related workload on the NAS controller, there is the potential that this will make more NAS resources available to process the FC traffic. This might have the indirect effect of improving FC performance, but will depend on the specifics of your storage environment.

6.5.2 iSCSI protocol access

Although the iSCSI protocol is transported over the same physical media (Ethernet cable) there is no interaction with the RTCA product.

Because the RTCA product compresses the NAS data, reducing the related workload on the NAS controller, there is the potential that this will make more NAS resources available to process the iSCSI traffic. This might have the indirect effect of improving FC performance, but will depend on the specifics of your storage environment.

Under normal conditions it is not necessary to bypass the RTCA product for any Ethernet traffic.

Tips:

- To avoid direct access to compressed data by clients without passing the RTCA product, it is best to connect all network interfaces of the NAS storage server with the RTCA product.
- If you have free Ethernet ports on your NAS storage server and want to use these ports for iSCSI or replication purposes, at these Ethernet ports all NAS protocols must be deactivated.

Part 2

Application integration use cases

In this part of the book, we demonstrate several application environments in which we implemented the IBM Real-time Compression Appliance. We cover the steps to take to implement the IBM RTCA with minimum impact to the application and to the environment.

7

VMware vSphere use cases

This chapter provides a practical approach about how to implement the IBM Real-time Compression Appliance when using VMware with NFS datastores. We start by describing the setup of our test environment and then continue with the implementation steps.

7.1 Test environment setup

To show you how to implement the IBM Real-time Compression Appliance we used a simple VMware implementation. As shown in Figure 7-1, we use one host connected to an NFS volume that it uses as a datastore. We created six virtual machines, all running Windows 2008 Server. The NFS datastores are presented to the host using an IBM N series cluster. The IBM N series uses multi-layered VIFs to be highly available at all time.

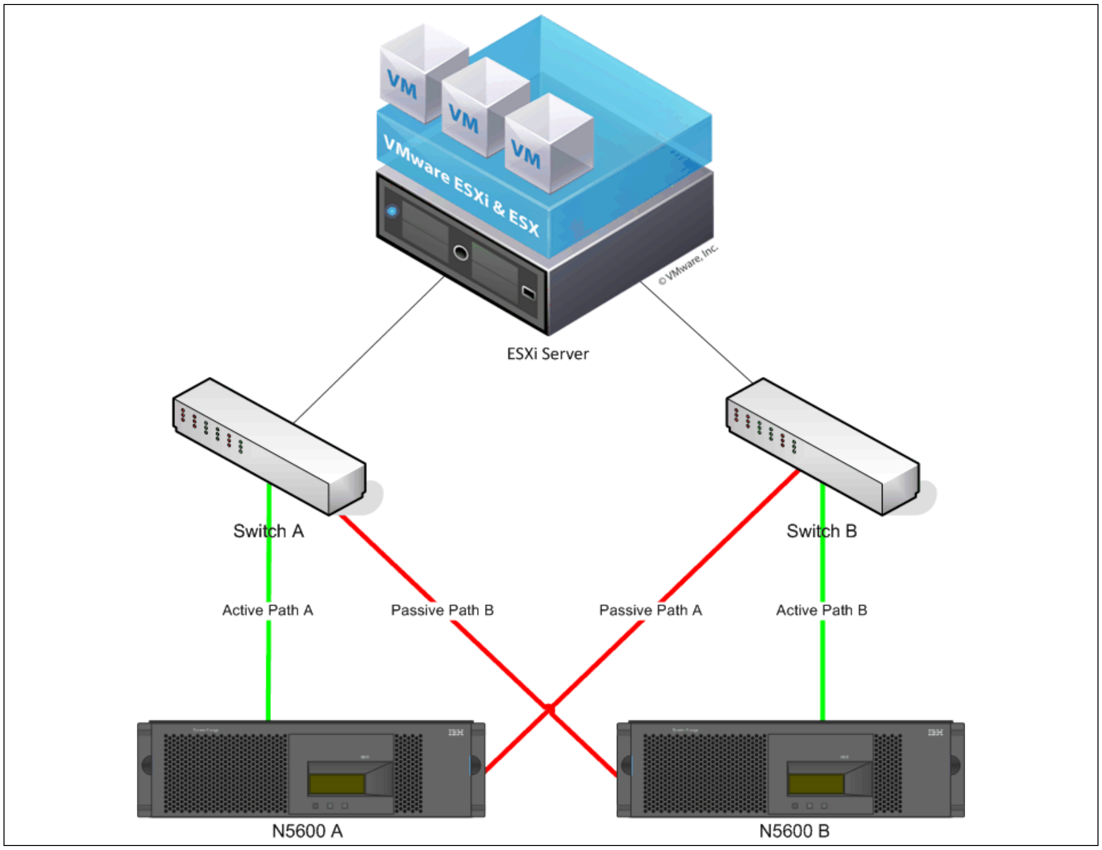


Figure 7-1 VMware setup without the IBM Real-time Compression Appliance

7.2 Planning for implementation

Before starting to implement the IBM Real-time Compression Appliance, you must do some planning. The planning is limited to providing IP addresses for the management port of each appliance and an IP address for each bridge you want to create.

> **Attention:** You can expect to implement the IBM Real-time Compression Appliance non disruptively. However, always prepare the proper change and backout planning.

7.3 Preparing the IBM Real-time Compression Appliance

Before implementing the IBM RTCA, you must prepare the bridges and bonds for your setup. In our case, we need, per RTCA, four bonds and two bridges.

7.4 Installing the IBM Real-time Compression Appliance

In this part, we show you how we installed the IBM Real-time Compression Appliance in front of our IBM N series N5600 cluster.

7.4.1 Installation without IBM RTCA

For this part, we refer you to Figure 7-1 on page 74, which shows the actual setup from which we started.

7.4.2 Connecting the passive path to the IBM RTCA

We start by disconnecting the passive path of the multi layered VIF. We will reconnect it through the IBM RTCA.

The first step is to check the passive path. We connect to the IBM N series and use the command **vif status vif0**, as shown in Example 7-1.

Example 7-1 Issuing the command 'vif status vif0'

```
N5600-A> vif status vif0
default: transmit 'IP Load balancing', VIF Type 'multi_mode', fail 'log'
vif0: 1 link, transmit 'none', VIF Type 'single_mode' fail 'default'
     VIF Status    Up     Addr_set
    up:
   vif1: state up, since 20Oct2011 09:13:33 (00:03:24)
      mediatype: Enabled virtual interface
      flags: enabled favored
      input packets 613761708, input bytes 334762812541
      output packets 801813939, output bytes 980301047646
      output probe packets 175176, input probe packets 175151
      strike count: 0 of 10
      up indications 8, broken indications 1
      drops (if) 0, drops (link) 0
      indication: up at 20Oct2011 09:13:33
         consecutive 526118, transitions 9
    down:
   vif2: state down, since 20Oct2011 09:14:05 (00:02:52)
      mediatype: Enabled virtual interface
      flags: enabled
      input packets 790190606, input bytes 658415884281
      output packets 831068905, output bytes 819332121433
      output probe packets 175171, input probe packets 175154
      strike count: 0 of 10
      up indications 12, broken indications 4
      drops (if) 0, drops (link) 0
      indication: up at 19Oct2011 13:35:11
```

```
      consecutive 525933, transitions 16
```

Example 7-1 on page 75 shows us that vif1 is active and vif2 is passive. To check which physical ports are connected to vif2, we issue the command **vif status vif2** as shown in Example 7-2.

Example 7-2 Issuing the command 'vif status vif2'

```
N5600-A> vif status vif2
default: transmit 'IP Load balancing', VIF Type 'multi_mode', fail 'log'
vif2: 2 links, transmit 'IP Load balancing', VIF Type 'lacp' fail 'default'
     VIF Status    Up     Addr_set
    trunked: vif0
    down:
   e0d: state up, since 19Oct2011 13:35:11 (19:45:59)
      mediatype: auto-1000t-fd-up
      flags: enabled
      active aggr, aggr port: e0d
      input packets 310599273, input bytes 250955714839
      input lacp packets 17585, output lacp packets 17610
      output packets 533670511, output bytes 569553754316
      up indications 32, broken indications 10
      drops (if) 0, drops (link) 0
      indication: up at 19Oct2011 13:35:11
         consecutive 0, transitions 42
   e0b: state up, since 19Oct2011 13:35:11 (19:45:59)
      mediatype: auto-1000t-fd-up
      flags: enabled
      active aggr, aggr port: e0d
      input packets 479591995, input bytes 407460215291
      input lacp packets 17582, output lacp packets 17606
      output packets 297398498, output bytes 249778372961
      up indications 32, broken indications 10
      drops (if) 0, drops (link) 0
      indication: up at 19Oct2011 13:35:11
         consecutive 0, transitions 42
```

Example 7-2 shows us that vif2 consists of port e0b and port e0d. We will disconnect those two ports and connect them to a bond on RTCA2 and reconnect the bond, that is connect to the first bond with a bridge, back to the switch ports. We do this also for the other IBM N series and connect it to RTCA1. This step is shown in Figure 7-2 on page 77.

When we disconnect the cables, the storage box will log some errors similar to those in Example 7-3.

Example 7-3 Error log when disconnecting the passive path

```
[N5600-A: netif.linkDown:info]: Ethernet e0d: Link down, check cable.
[N5600-A: netif.linkDown:info]: Ethernet e0b: Link down, check cable.
[N5600-A: pvif.allLinksDown:CRITICAL]: vif2: all links down
[N5600-A: netif.linkUp:info]: Ethernet e0b: Link up.
[N5600-A: netif.linkUp:info]: Ethernet e0d: Link up.
[N5600-A: pvif.switchLink:warning]: vif2: switching to e0b
```

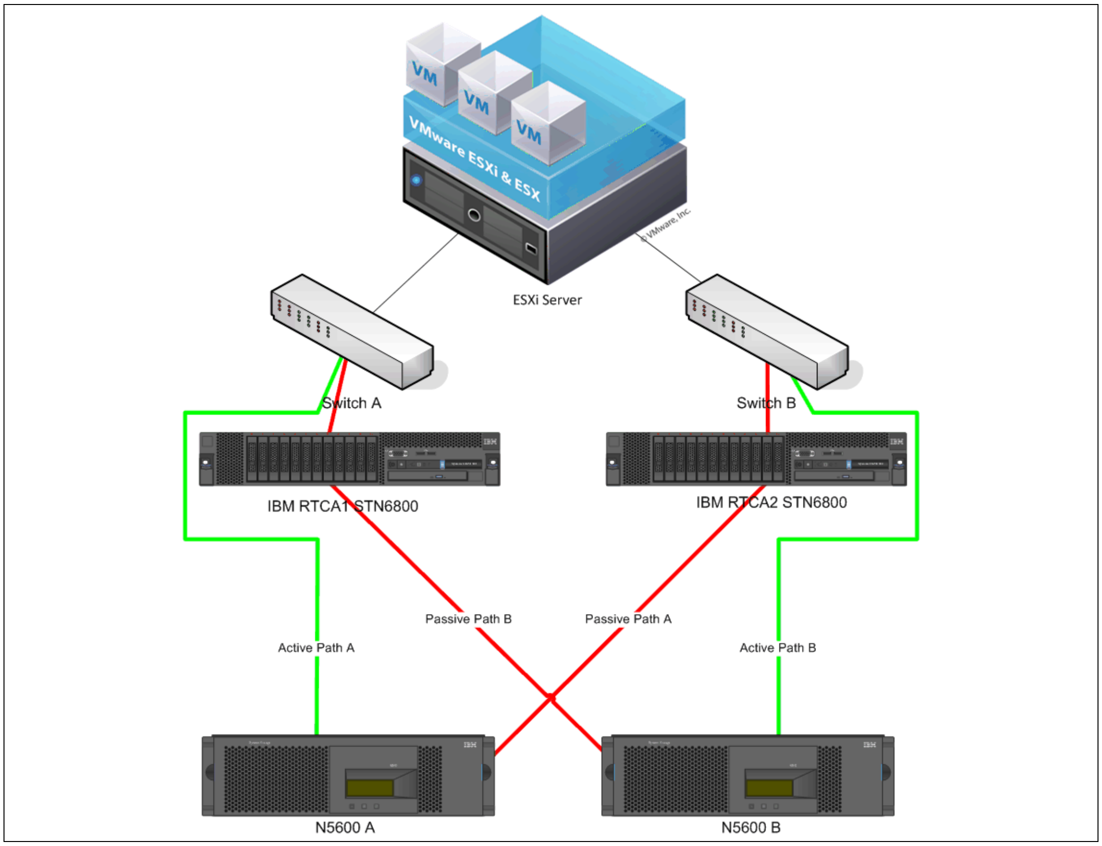


Figure 7-2 VMware setup with the passive path through the IBM RTCA

7.4.3 Switching active and passive path

The next step is to switch the active and passive paths, but before doing that, we will do some tests to see if the IBM RTCA was connected correctly. We do this using the ping command on both the IBM N series and the IBM Real-time Compression Appliance.

From the IBM N series, we ping all IP addresses of the IBM RTCA bridges (Example 7-4).

Example 7-4 Ping tests on IBM N series

```
N5600-A> ping 9.155.90.170
9.155.90.170 is alive
N5600-A> ping 9.155.90.171
N5600-A> ping 9.155.90.172
N5600-A> ping 9.155.90.173
9.155.90.173 is alive
```

We can only ping the IP addresses of the bridges that are already connected. The IBM N series will ping the bridge IP addresses using its active path, go through the switch, and back down to the passive path.

> **Tip:** To check if all cabling is done correctly for the passive path, you can check the status panel of the IBM Real-time Compression Appliance. The connected ports will light up green if everything is OK. This is shown in Figure 7-3 on page 78.

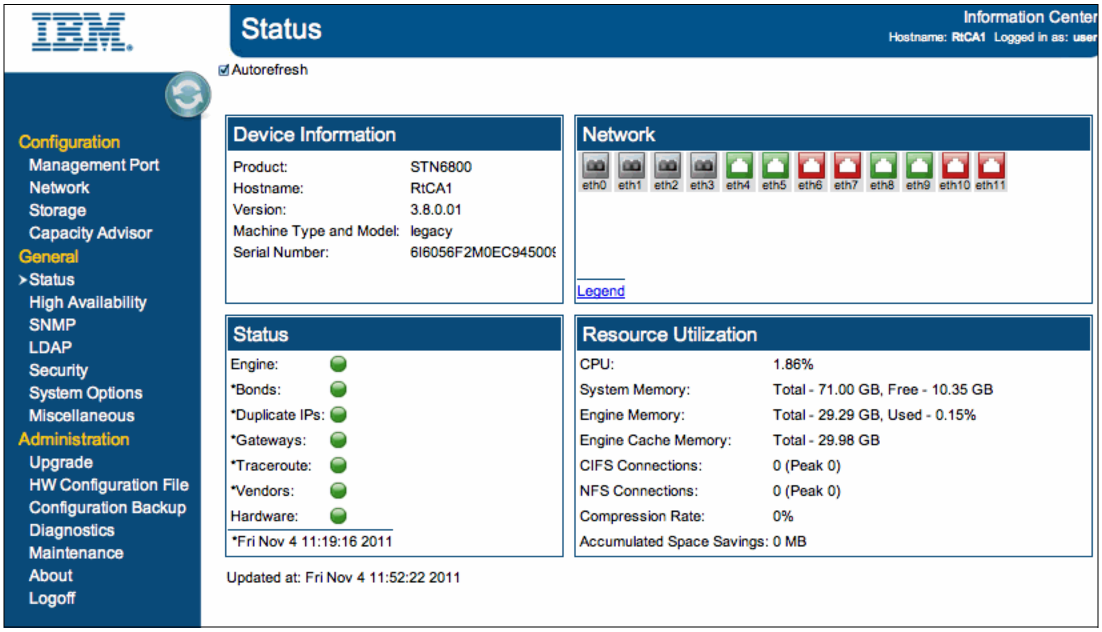


Figure 7-3 RTCA Status panel with the passive path connected

> **Tip:** The lit ports do not take in consideration that bonding is done on the ports, they only check the physical connectivity.

Tests that you can do using the IBM Real-time Compression Appliance are located in the “Diagnostics” page, which you can find in the “Administration” part of the menu.

In the “Diagnostic Tools” you can run the **`ping`** command from any interface you want. Select the bridge you already connected to and try to ping the IP addresses of the connected bridges and storage servers.

> **Tip:** Figure 7-4 on page 79 shows you how to select an interface from which you can issue the **`ping`** command. Be sure to select the already connected bridge to test these commands.

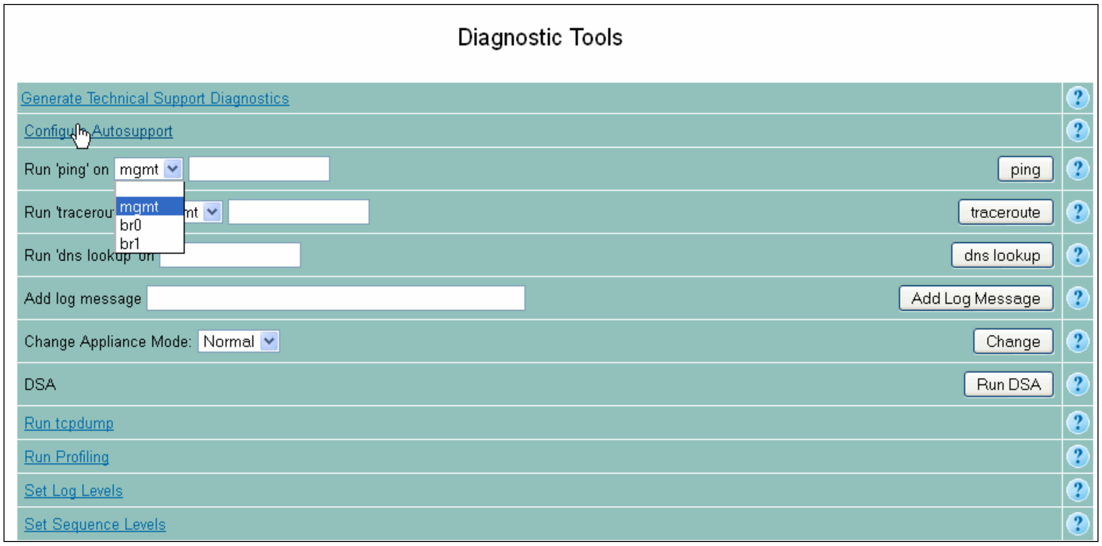


Figure 7-4 Select the interface for the 'ping' command

To send the command, type in the IP address after selecting the interface, and click **ping**. The result will be shown in the web interface (see Figure 7-5).

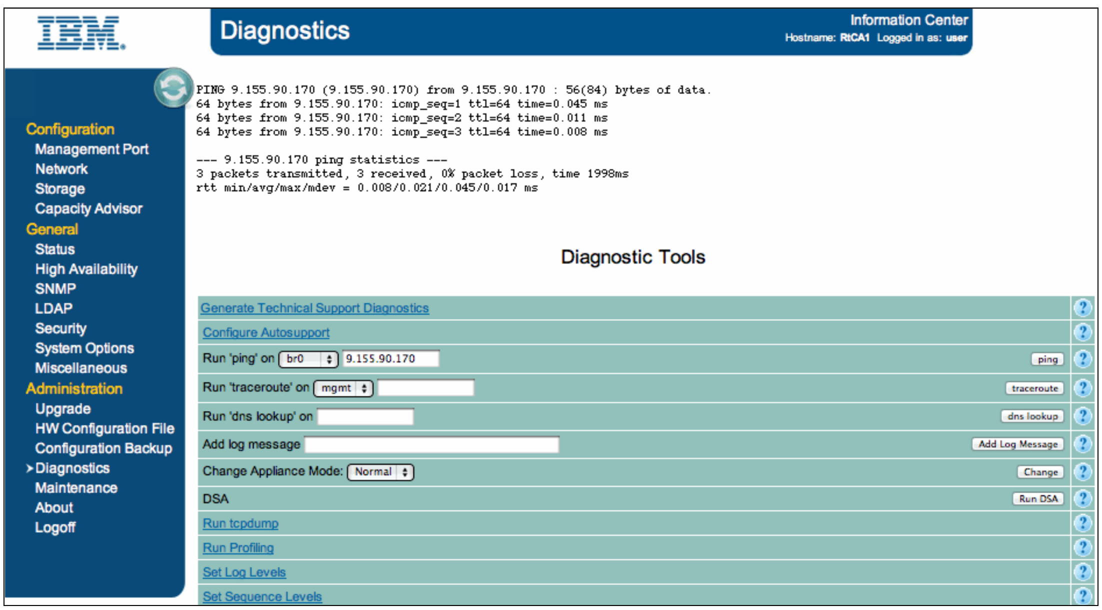


Figure 7-5 Ping result

When all tests seem alright, you can switch the active and the passive path by issuing the command **`vif favor <name of vif>`** (see Example 7-5).

Example 7-5 Issuing the command 'vif favor <name of vif>'

```
N5600-A> vif favor vif2
[N5600-A: pvif.allLinksDown:CRITICAL]: vif2: all links down
[N5600-A: pvif.switchLink:warning]: vif2: switching to e0b
[N5600-A: pvif.switchLink:warning]: vif0: switching to vif2
[N5600-A: netinet.ethr.duplct.ipAddr:error]: Duplicate IP address 9.155.90.14!!
sent from Ethernet address:00:24:3f:00:22:f6.
```

The last line in Example 7-5 on page 79 shows the error of a duplicate IP address. This is due to the fact that when a failover occurs, a gratuitous ARP is sent from the storage to the switch through the appliance. This gratuitous ARP is sent back to the storage server with the appliance MAC address, which will cause the storage server to log a duplicate IP address message.

Figure 7-6 shows a view of how the data flows when the active and passive paths are switched.

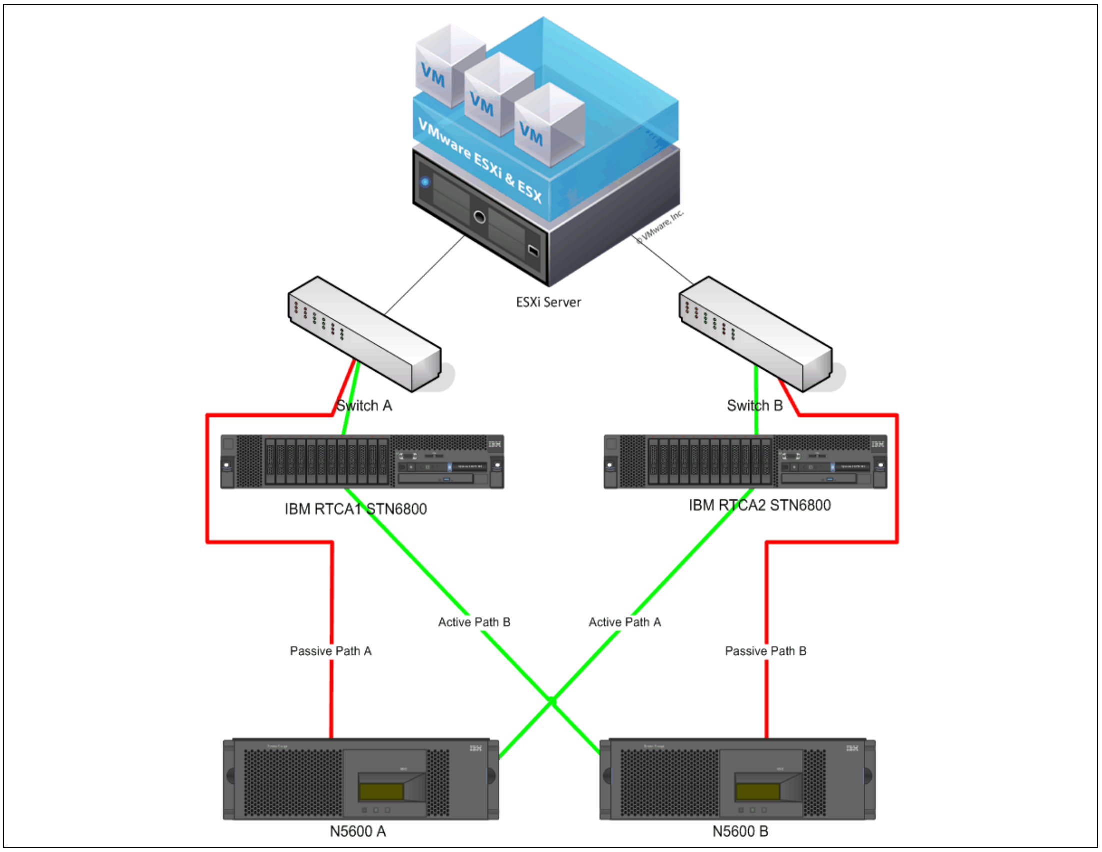


Figure 7-6 Active path through the IBM Real-time Compression Appliance

7.4.4 Connecting the active path to the RTCA

Now we can connect the active path, which was made passive now, to the RTCA. Start by checking if the correct VIF is active. Again, use the **vif status** command (Example 7-6).

Example 7-6 Checking the active VIF

```
N5600-A> vif status vif0
default: transmit 'IP Load balancing', VIF Type 'multi_mode', fail 'log'
vif0: 1 link, transmit 'none', VIF Type 'single_mode' fail 'default'
     VIF Status    Up     Addr_set
    up:
   vif2: state up, since 20Oct2011 13:14:50 (20:16:55)
      mediatype: Enabled virtual interface
      flags: enabled favored
      input packets 877095722, input bytes 705765665353
      output packets 952718212, output bytes 976857724758
```

```
      output probe packets 204137, input probe packets 204111
      strike count: 0 of 10
      up indications 13, broken indications 4
      drops (if) 0, drops (link) 0
      indication: up at 20Oct2011 13:14:50
         consecutive 613218, transitions 17
    down:
   vif1: state down, since 20Oct2011 13:46:43 (19:45:02)
      mediatype: Enabled virtual interface
      flags: enabled
      input packets 693423969, input bytes 393548956950
      output packets 894087114, output bytes 1082351380001
      output probe packets 204142, input probe packets 204107
      strike count: 0 of 10
      up indications 12, broken indications 5
      drops (if) 0, drops (link) 0
      indication: up at 20Oct2011 13:46:43
         consecutive 613051, transitions 17
```

To check which physical port is associated with the VIF, issue the command **`vif status <name of vif>`** as shown in Example 7-7.

Example 7-7 Issuing the command 'vif status vif1'

```
N5600-A> vif status vif1
default: transmit 'IP Load balancing', VIF Type 'multi_mode', fail 'log'
vif1: 2 links, transmit 'IP Load balancing', VIF Type 'lacp' fail 'default'
     VIF Status    Up     Addr_set
    trunked: vif0
    down:
   e0c: state up, since 20Oct2011 13:46:43 (19:50:09)
      mediatype: auto-1000t-fd-up
      flags: enabled
      active aggr, aggr port: e0a
      input packets 422670408, input bytes 215767150480
      input lacp packets 20486, output lacp packets 20790
      output packets 850730315, output bytes 1038814078296
      up indications 33, broken indications 14
      drops (if) 0, drops (link) 0
      indication: up at 20Oct2011 13:46:43
         consecutive 0, transitions 47
   e0a: state up, since 20Oct2011 13:46:43 (19:50:09)
      mediatype: auto-1000t-fd-up
      flags: enabled
      active aggr, aggr port: e0a
      input packets 270754287, input bytes 177781856544
      input lacp packets 20483, output lacp packets 20791
      output packets 43356921, output bytes 43537308469
      up indications 36, broken indications 16
      drops (if) 0, drops (link) 0
      indication: up at 20Oct2011 13:46:43
         consecutive 0, transitions 52
```

Example 7-7 on page 81 shows us that vif1 consists of port e0a and e0c. We will disconnect those two ports and connect them to a bond on RTCA1 and reconnect the bond. That is, we will connect to the first bond with a bridge, back to the switch ports.

We will do this also for the other IBM N series and connect it to RTCA2. While disconnecting, we see similar entries in the log files, as shown in Example 7-8.

Example 7-8 Log entries when disconnecting and reconnecting the active path

```
[N5600-A: netif.linkDown:info]: Ethernet e0a: Link down, check cable.
[N5600-A: netif.linkDown:info]: Ethernet e0c: Link down, check cable.
[N5600-A: pvif.allLinksDown:CRITICAL]: vif1: all links down
[N5600-A: netif.linkUp:info]: Ethernet e0a: Link up.
[N5600-A: pvif.switchLink:warning]: vif1: switching to e0a
[N5600-A: netif.linkUp:info]: Ethernet e0c: Link up.
```

When the cables are connected, the logical view of the environment will look like Figure 7-7.

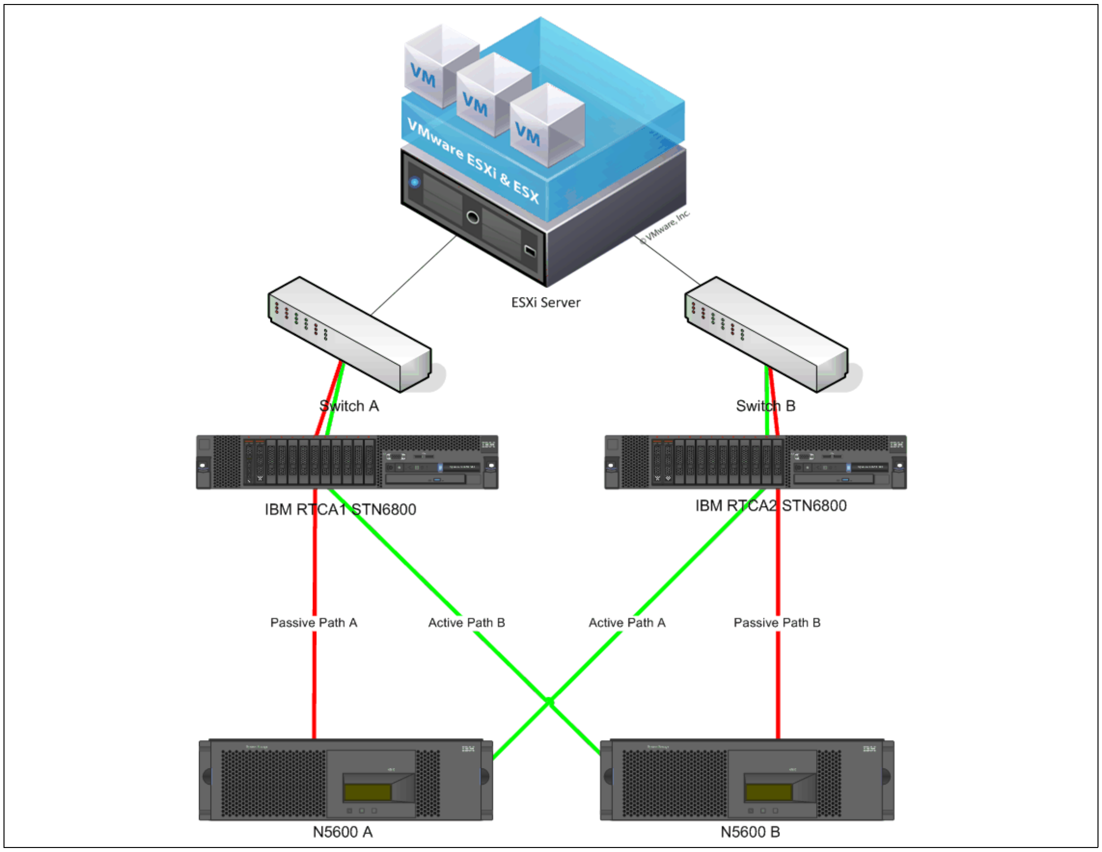


Figure 7-7 Logical view with all paths going through the RTCA

In the web interface of the IBM Real-time Compression Appliance on the Status panel, you will see something such as all connected ports in green, as shown in Figure 7-8 on page 83.

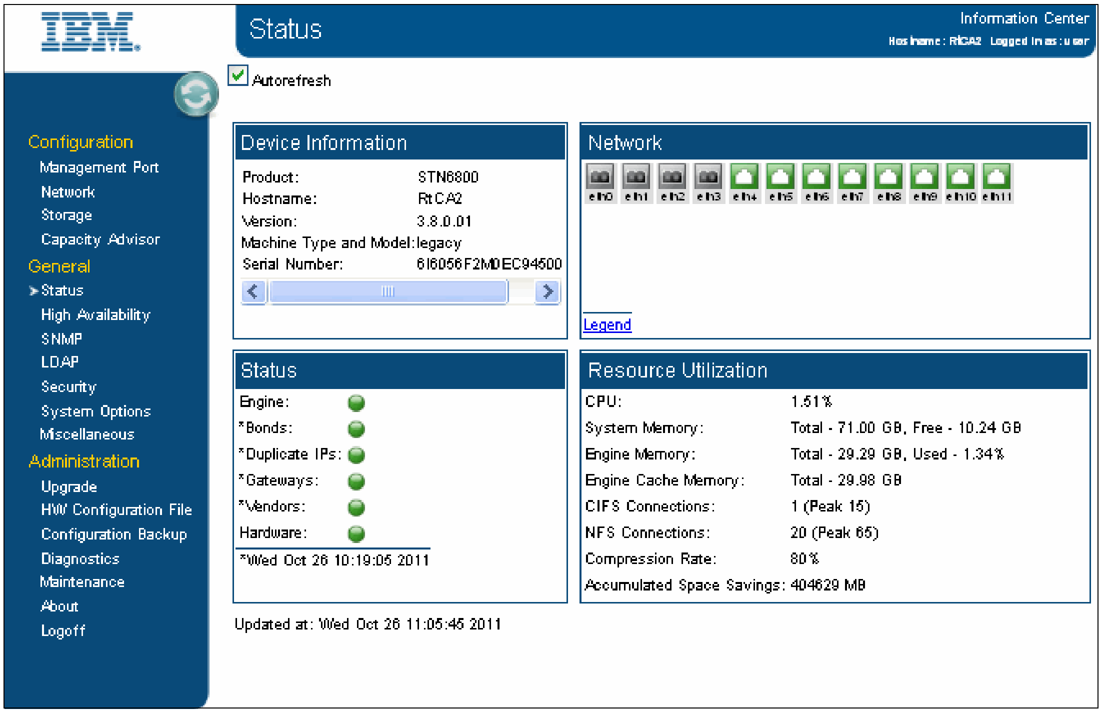


Figure 7-8 RTCA status panel after all ports are connected

As you can see in Figure 7-8, there are already some NFS connections. So even when compression was not yet configured, the appliance already detects the NFS and CIFS connections to the storage connected to the appliance.

7.4.5 Resetting active paths in the initial state

The last step is to reset the multi layered VIF in its initial active-passive state. We do this by issuing the command **`vif favor <name of vif>`**. Example 7-9 provides details of the command.

Example 7-9 Issuing the command 'vif favor vif1'

```
N5600-A> vif favor vif1
[N5600-A: pvif.allLinksDown:CRITICAL]: vif1: all links down
[N5600-A: pvif.switchLink:warning]: vif1: switching to e0a
[N5600-A: pvif.switchLink:warning]: vif0: switching to vif1
```

The IBM Real-time Compression Appliance is now physically installed. The logical view of the current environment is shown in Figure 7-9.

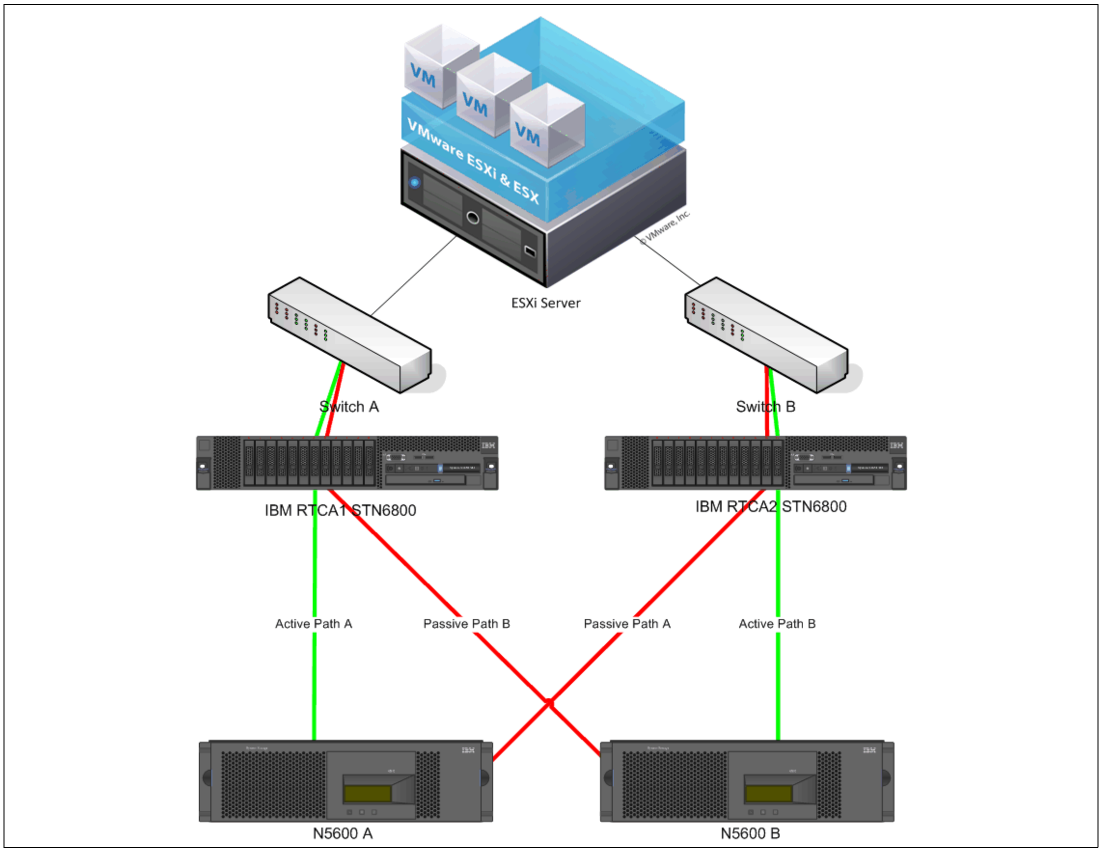


Figure 7-9 Logical view of the environment

7.5 Turning on compression

Now that the IBM Real-time Compression Appliance is placed in between the storage and the switches, we can start the compression for certain on all shares or exports.

Figure 7-10 shows the status of the IBM Real-time Compression Appliance. All configured Ethernet ports have a green color. The Real-time Compression Engine is running. The bonds are good. There are no duplicate IPs in the network and the hardware is healthy.
We can start off.

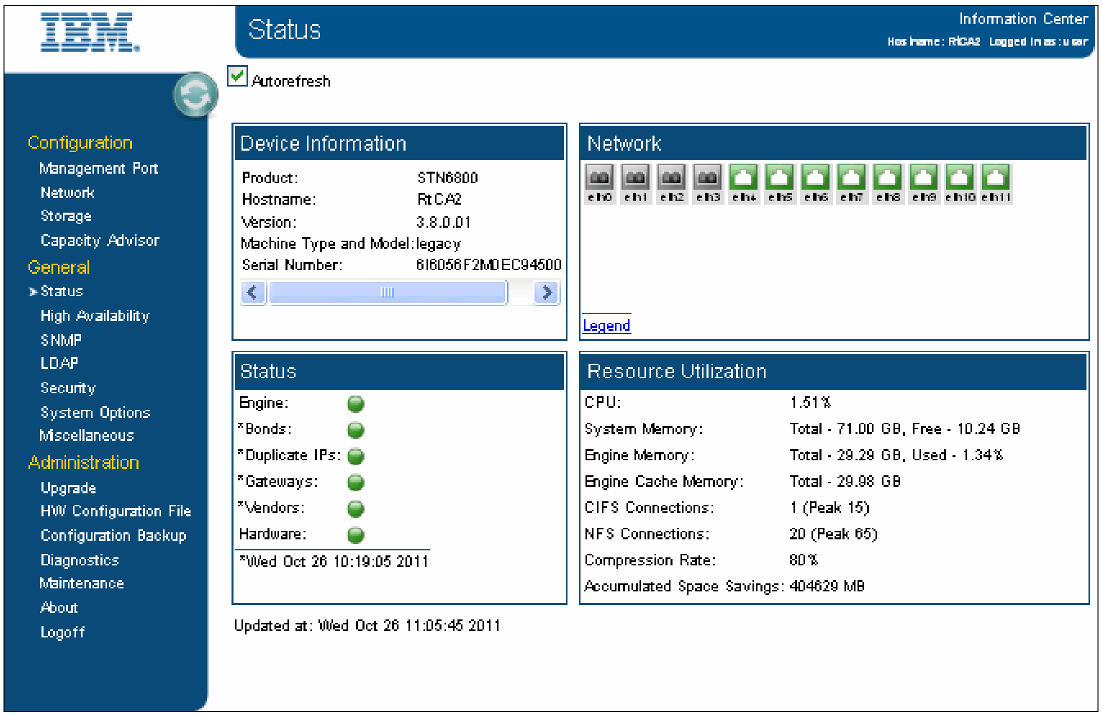


Figure 7-10 IBM RTCA Status panel

In the previous caption, we installed the IBM RTCA, in transparent mode, into the data flow. Transparent mode implies that all data will pass through the RTCA without being compressed unless the RTCA is told otherwise.

We will now tell the Real-time Compression Appliance to compress the data for the VMware volume we are using. We do that by clicking the **Storage** link under the Configuration menu (see Figure 7-11).

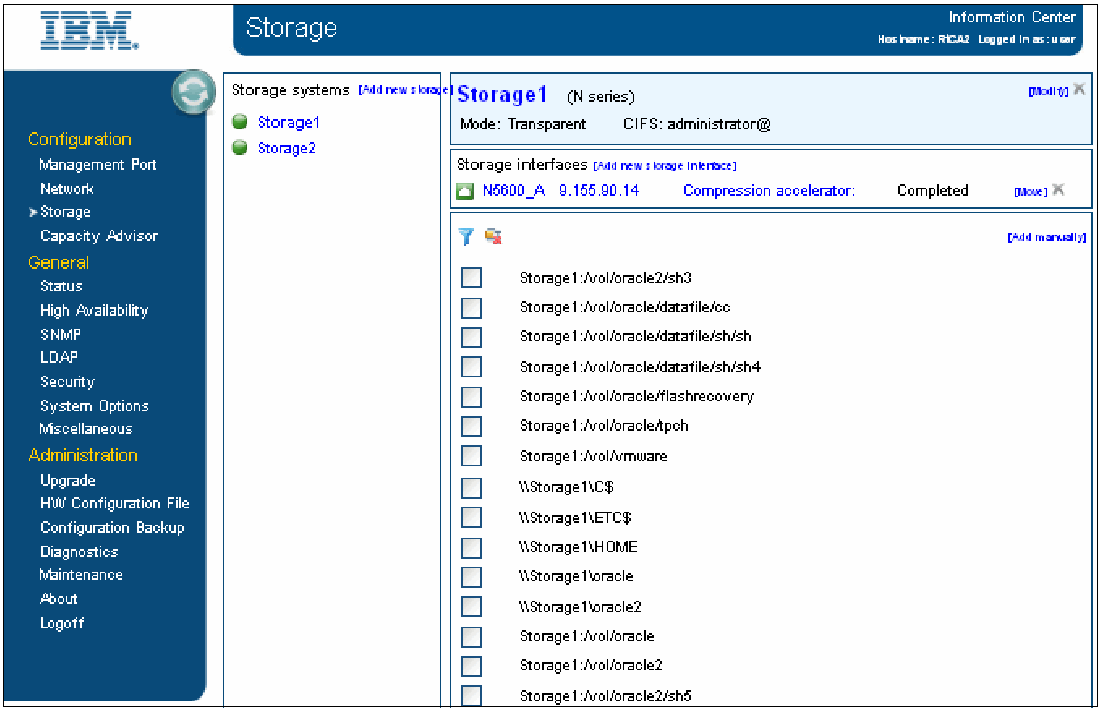


Figure 7-11 Storage page of the RTCA

We select the option next to the share we want to compress, which makes a second option appear. The new option is the “no new compression” option. When we select the no new compression option, the RTCA will not compress newly written data. We only select the compression option and then click **Apply** on the top of the page (see Figure 7-12).

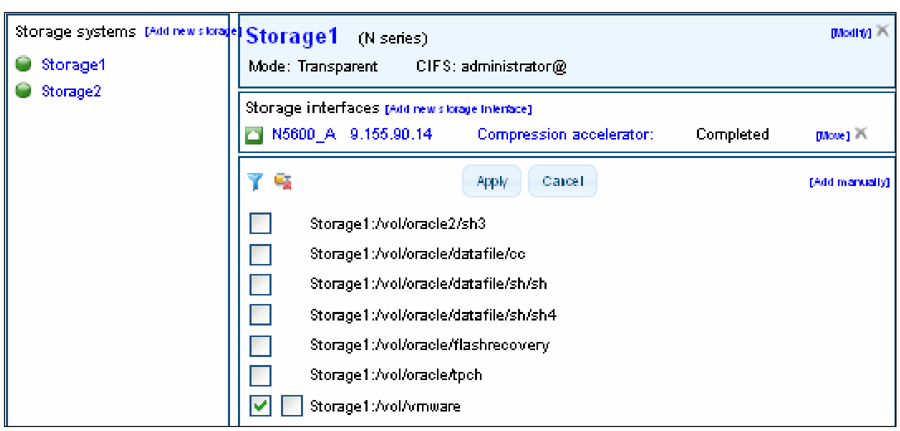


Figure 7-12 Select shares to compress

These actions complete the compression setup. All data that will be written to the share from now on will be compressed. The files with extensions that are listed to be excluded will not be compressed. To compress the data that is already written on the share, we have to run the Compression Accelerator. Details on the Compression Accelerator are given in the next part, 7.6, “Setting up the Compression Accelerator” on page 87.

7.6 Setting up the Compression Accelerator

Setting up the Compression Accelerator consists of two steps:

- Providing access to the IBM Real-time Compression Appliance on the shares
- Setting up and starting the Compression Accelerator

7.6.1 Providing root access to the IBM RTCA

To be able to run the Compression Accelerator on NFS shares, the IBM Real-time Compression Appliance needs root access to read and write the files. The root access is also necessary to implement the security of all files correctly. We start with Figure 7-13. Here you can see the actual status of the storage, in our case, an IBM N series.

Filer	N5600-A
Model	N5600
System ID	0118054991
Version	7.3.6P1
Volumes	5 Volumes
Aggregates	1 Aggregates
Disks	7 Disks (1 spare, 0 failed)
Status	The system's global status is normal.

Figure 7-13 IBM N series status

To run the Compression Accelerator on NFS shares:

1. Go to NFS, and select **Manage Exports**, as shown in Figure 7-14.

Figure 7-14 Manage exports

2. Select the export you want to handle by clicking the '**`/vol/<vol name>`**' hot link (Figure 7-14 on page 87).

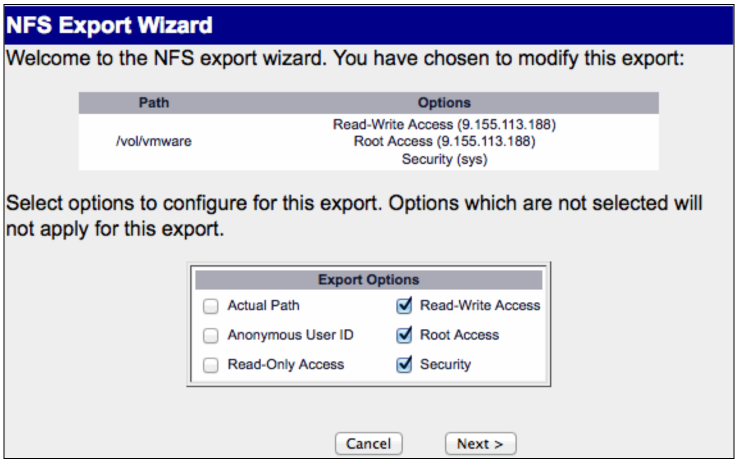


Figure 7-15 Manage /vol/vmware Export

3. Follow the menu steps until you reach the Read-Write Hosts page (Figure 7-16).

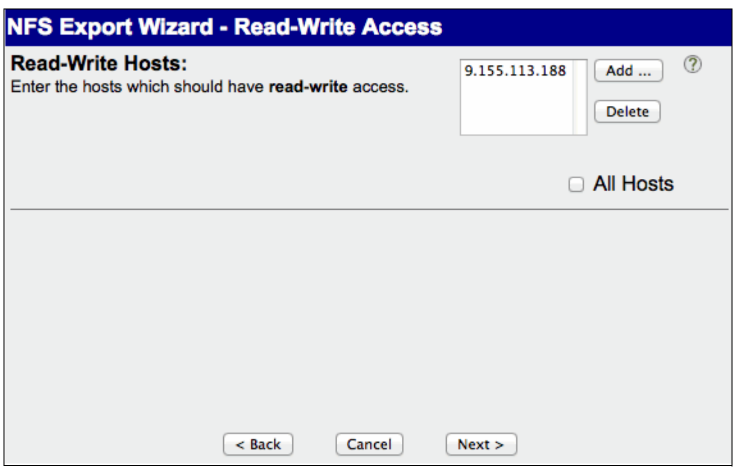


Figure 7-16 Read-Write Hosts page

4. Add the IP address of the first bridge by clicking **Add** in Figure 7-16. This will open a pop-up page (Figure 7-17).

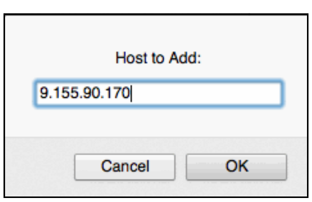


Figure 7-17 Add host to read-write access pop-up

5. Click the **OK** button and repeat this sequence for all bridges.

All bridges need read and write access to be able to run the Compression Accelerator without any issues (Figure 7-18).

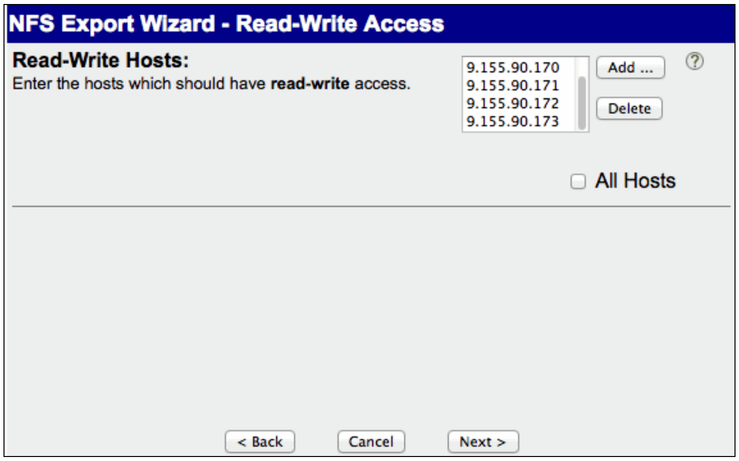


Figure 7-18 All bridges have read and write access

6. Click the **Next** button to reach the root access part (Figure 7-19).
7. Click **Add** to add the IP address of the first bridge.

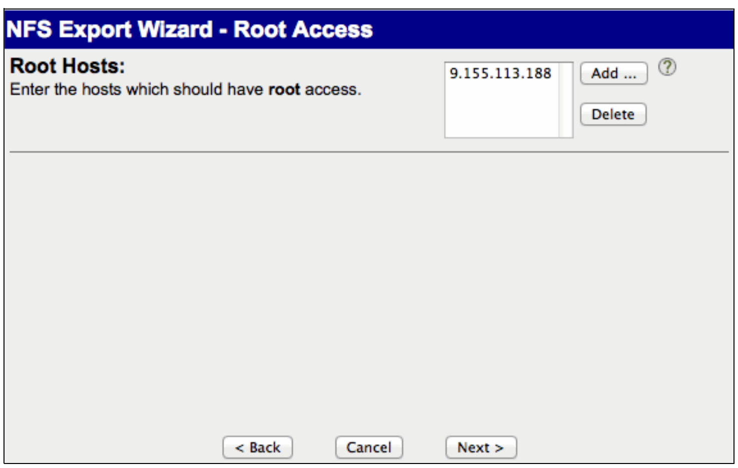


Figure 7-19 Root Access page

8. Fill in the IP address for the first bridge, and click **OK**.

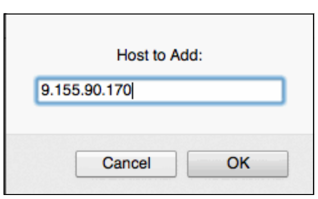


Figure 7-20 Add host for root access

9. Repeat this sequence for all bridges. All bridges also need root access to be able to set the correct access rights (Figure 7-21).

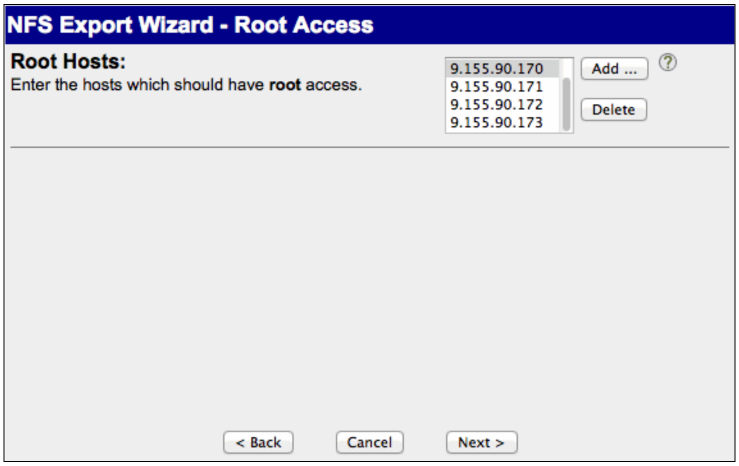


Figure 7-21 All bridges have Root access

The next page displays the security settings for the export. Just leave these as they are (Figure 7-22).

Figure 7-22 Export Security

10.Click **Next**. You will receive a summary of the changes you want make to this export, 11.Click **Commit** to commit your changes (Figure 7-23).

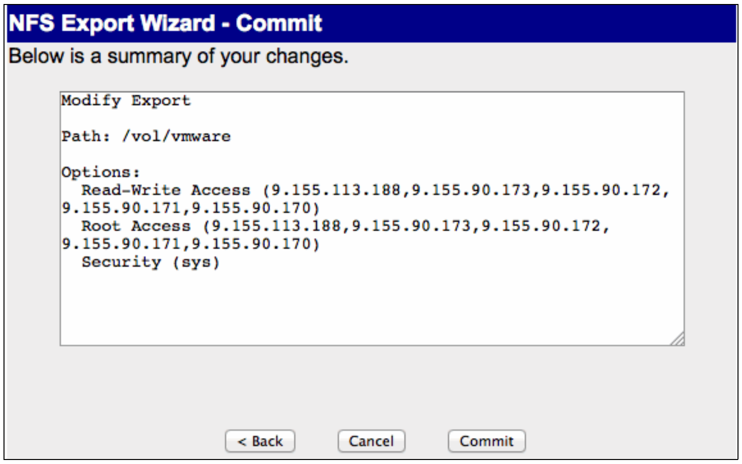


Figure 7-23 Summary of changes

12.Close the pop-up window by clicking **Close** (Figure 7-24).

Figure 7-24 Changes are committed successfully

When this window is closed, you will notice that the options of the export you just manipulated did not change (Figure 7-25).

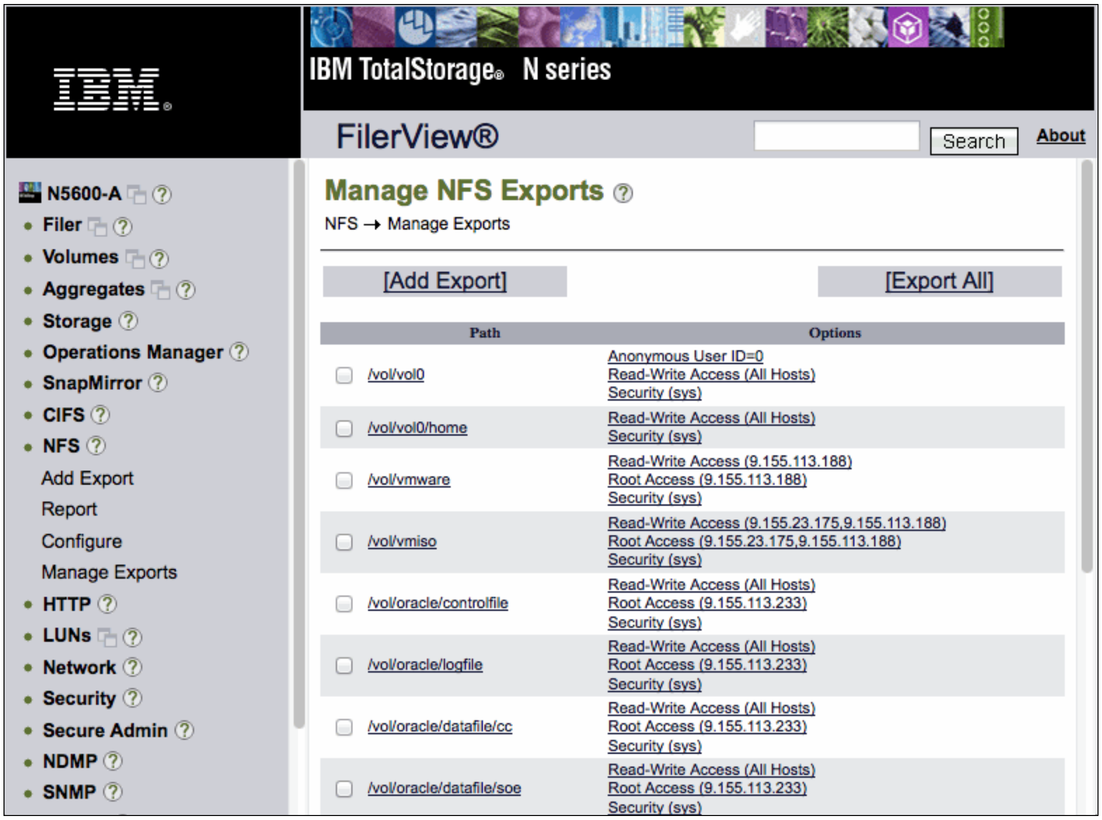


Figure 7-25 Export options did not change

13.Scroll down the page. Click **Refresh** to reload the web page (Figure 7-26).

Figure 7-26 Refresh button on the bottom of the page

The result is shown in Figure 7-27. The IP addresses of all bridges are shown next to Read-Write access and Root access.

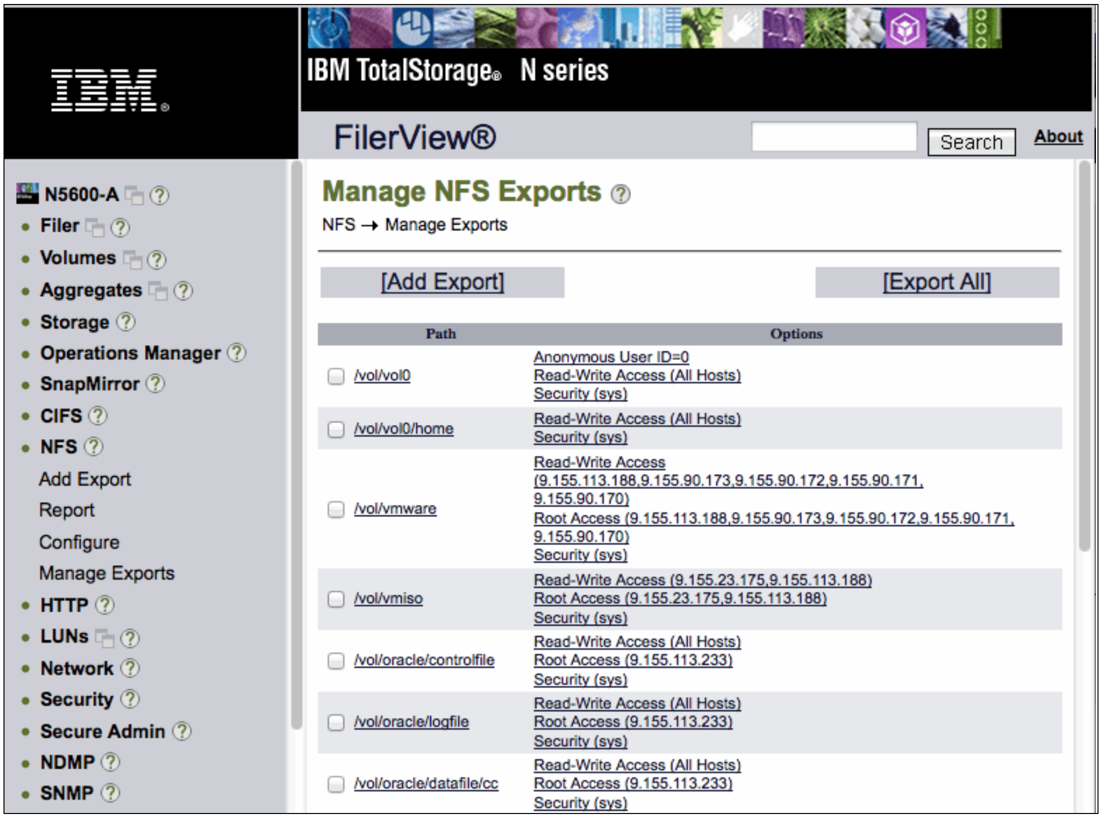


Figure 7-27 Export list reloaded

This concludes the steps you must take on the IBM N series machine. You must repeat all of these steps for each volume you want to compress using the Compression Accelerator.

You have the possibility to complete the previous steps using the command line interface (CLI).

To complete the Compression Accelerator steps, you must connect to the IBM Real-time Compression Appliances web interface, as described in the following section.

7.6.2 Configuring the Compression Accelerator

To configure the Compression Accelerator:

1. Login to the IBM Real-time Compression Appliances web interface (see Figure 7-28).

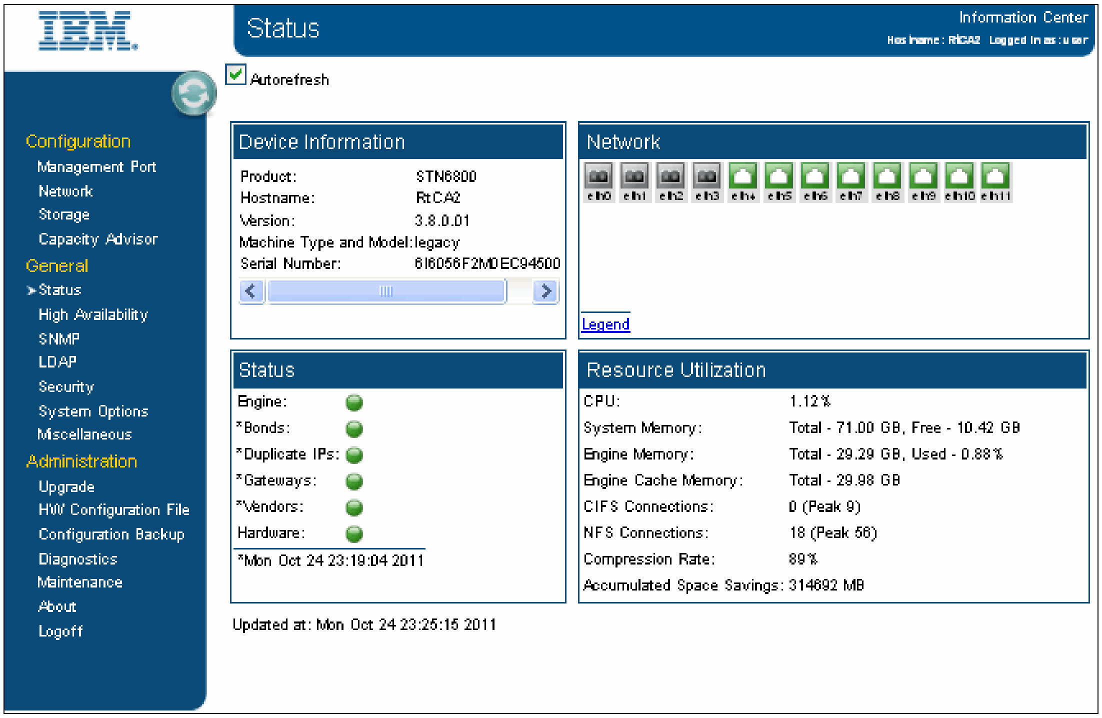


Figure 7-28 RTCA Status panel

2. Select the **Storage** tab in the Configuration sub menu (Figure 7-29).

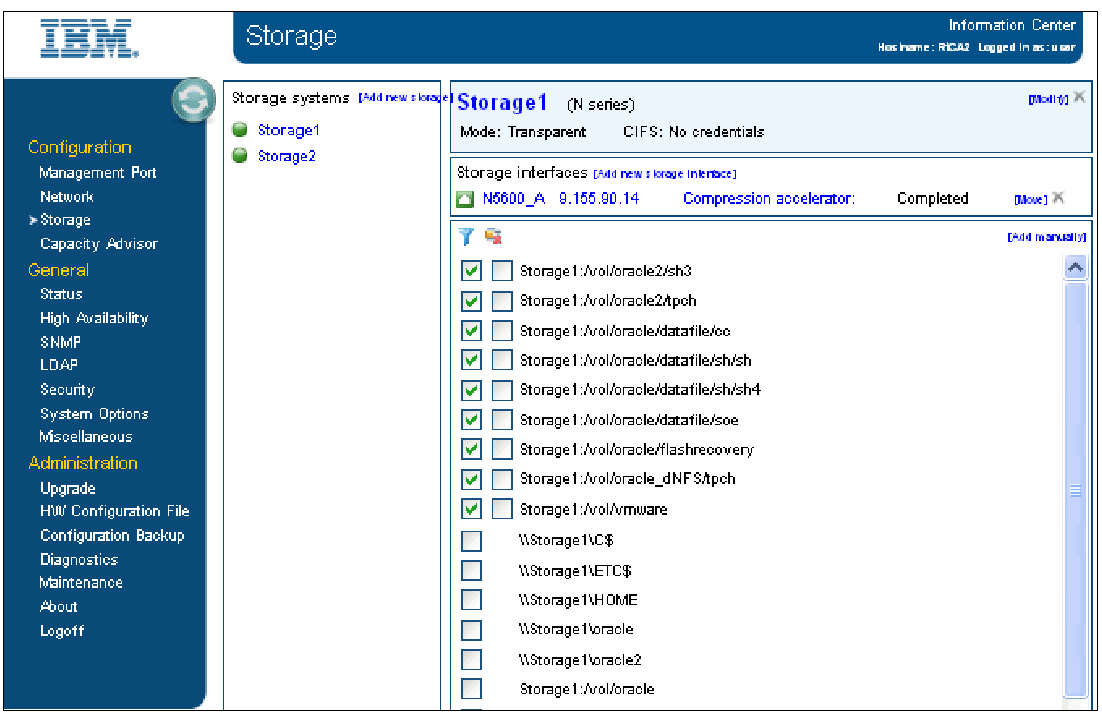


Figure 7-29 Storage view

3. To configure the Compression Accelerator, click the **Compression Accelerator** hot link, which is mentioned on the top of the page. Make sure that the correct storage is selected. If not, select the correct Storage from the side menu (see Figure 7-29 on page 94).

   After you click the **Compression Accelerator** hot link, you will see the Compression Accelerator Configuration page (Figure 7-30). Here, you can set the credentials to compress CIFS shares. On top of that, you can set the “Maximum Total Throughput” and the “Maximum Amount of Changes” per desired hour.

   For more information about the Compression Accelerator, see the companion IBM Redbooks publication *Introduction to IBM Real-time Compression Appliances*, SG24-7953, located at:

   http://www.redbooks.ibm.com/abstracts/sg247953.html?Open

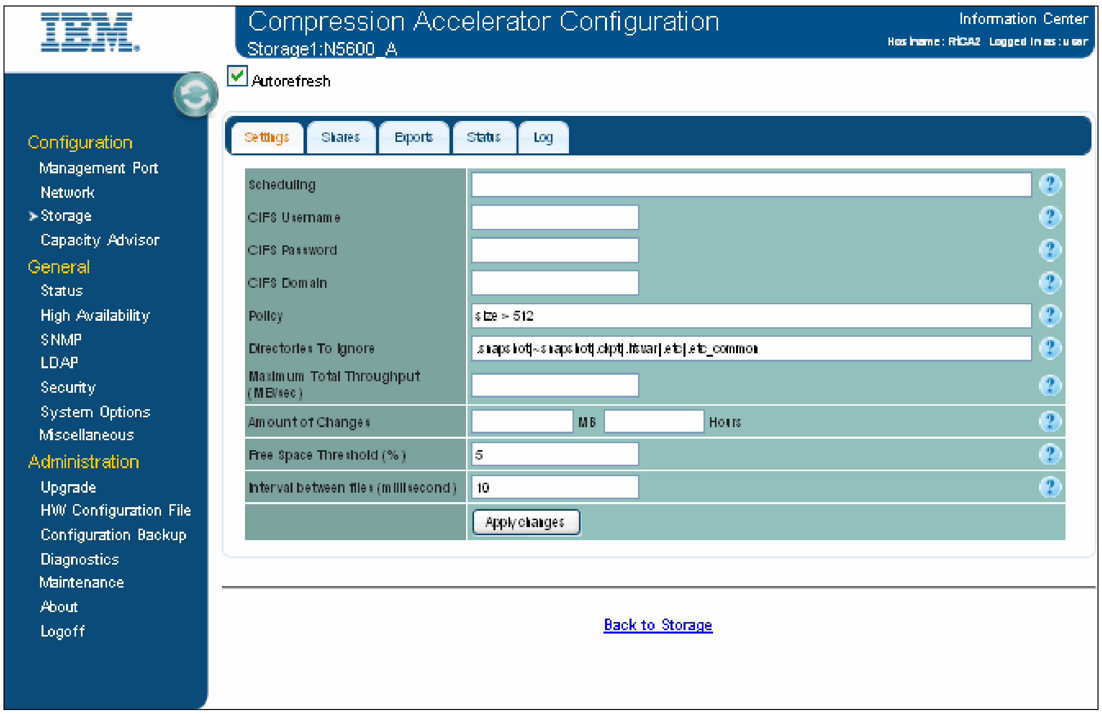


Figure 7-30 Compression Accelerator - Settings tab

   Because VMware only uses NFS exports, you do not need to populate the CIFS credentials.

4. Configure the “Maximum Total Throughput” and the “Maximum Amount of Changes” per desired hour. These are the only changes necessary. They have a direct impact on the snapshot space that is in use. Consider this when configuring the Compression Accelerator.

5. After you set the limitations for the Compression Accelerator, select **Apply changes** to confirm your settings (Figure 7-31).

Figure 7-31 Compression Accelerator settings filled in

For VMware, we only need the Exports tab, so we will skip the Shares tab.

6. Select the export you want to compress by checking the check box in front of the share that you want to compress, and click **Update List** (Figure 7-32).

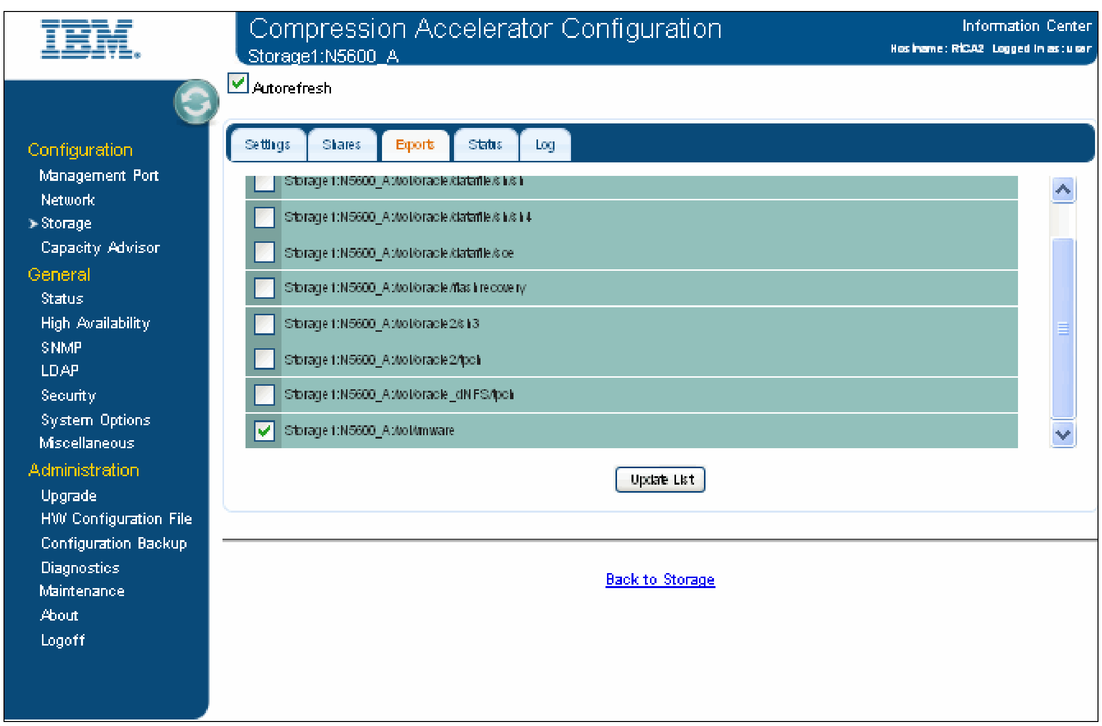


Figure 7-32 Compression Accelerator - Exports tab

7. After you set the shares that need to be compressed, you can start the Compression Accelerator by selecting the **Status** tab, as shown in Figure 7-33.

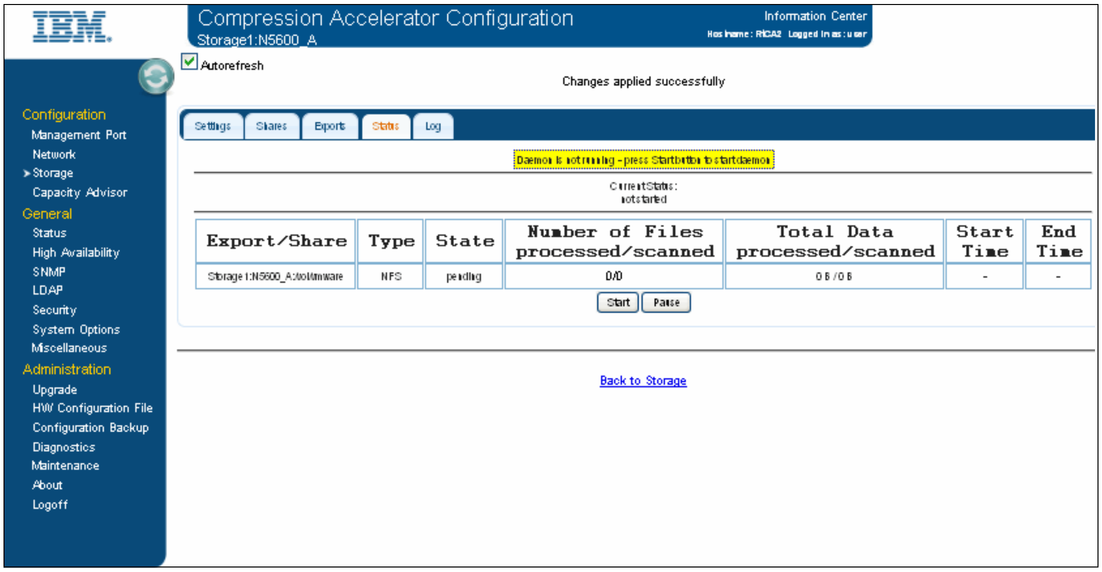


Figure 7-33 Compression Accelerator - Status tab

As the yellow bar is stating, the Compression Accelerator Daemon is not running.

8. To start compressing existing files on the selected shares, click **Start**. The **Start** button will start the daemon that reads the existing files and rewrites them as compressed files.

   The Compression Accelerator Daemon will read all files that are in the selected share and rewrite them as compressed files. During this process it will take into consideration the limits you just set in the Settings tab. After it reaches one of the limits, the daemon will pause itself until all data has been compressed (see Figure 7-34).

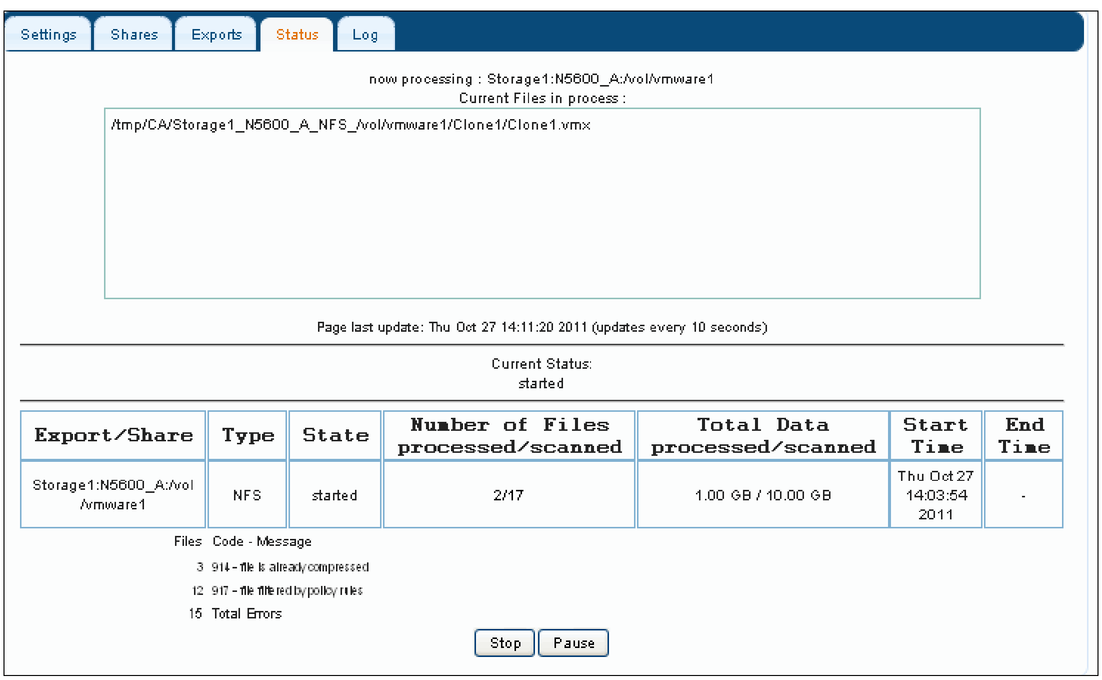


Figure 7-34 Running Compression Accelerator Daemon

Tip: The data written by the Compression Accelerator Daemon will be seen by the storage as newly written data. This will have an effect on your snapshot size, so do take this in consideration.

When the Compression Accelerator Daemon is running, you can follow any actions in the Log tab. In Figure 7-35, for example, you can see that at 12:01:34, the daemon paused itself due to the amount of changes.

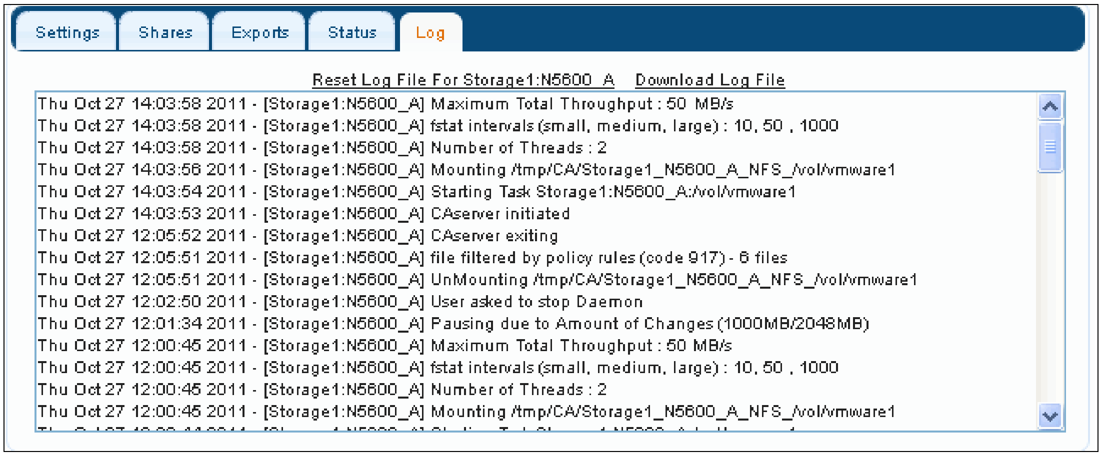


Figure 7-35 Compression Accelerator Daemon 'Log' tab

After all data is compressed, the Status tab will provide you an overview of the start and finish time and the amount of data that was compressed (Figure 7-36).

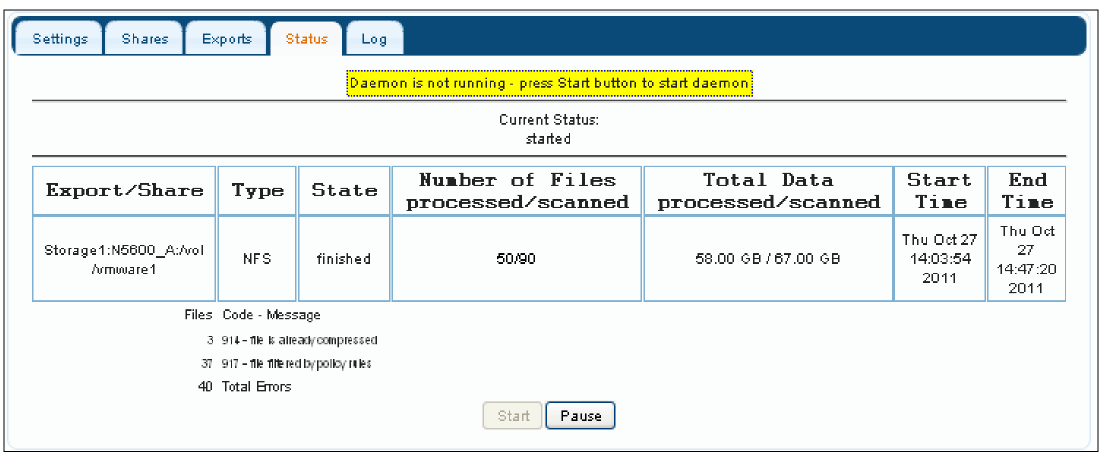


Figure 7-36 Compression Accelerator daemon finished

This concludes our discussion about configuring the Compression Accelerator.

7.7 Our experiences

In this section, we cover the results that all of the steps in this chapter have on your stored data. We also cover all of the native options you might want to implement, such as deduplication.

7.7.1 No compression

We start by showing you the uncompressed data. The data consists of six Windows 2008 Server virtual machines, using about 40 GB of data (Example 7-10).

Example 7-10 Issuing the command df -k and df -Sk on the IBM N series console

```
N5600-A> df -k
Filesystem               total       used      avail capacity  Mounted on
/vol/vol0/          25165824KB  1435764KB 23730060KB       6% /vol/vol0/
/vol/vmiso/         16777216KB   690196KB 16087020KB       4% /vol/vmiso/
/vol/vmware1/       104857600KB 40462636KB 64394964KB      39%  /vol/vmware1/
N5600-A> df -Sk
Filesystem                used       compressed         a-sis     %saved
/vol/vol0/           1435764KB              0KB           0KB         0%
/vol/vmiso/           690196KB              0KB           0KB         0%
/vol/vmware1/       40462636KB              0KB           0KB         0%
```

The VMware vCenter storage view is shown in Figure 7-37.

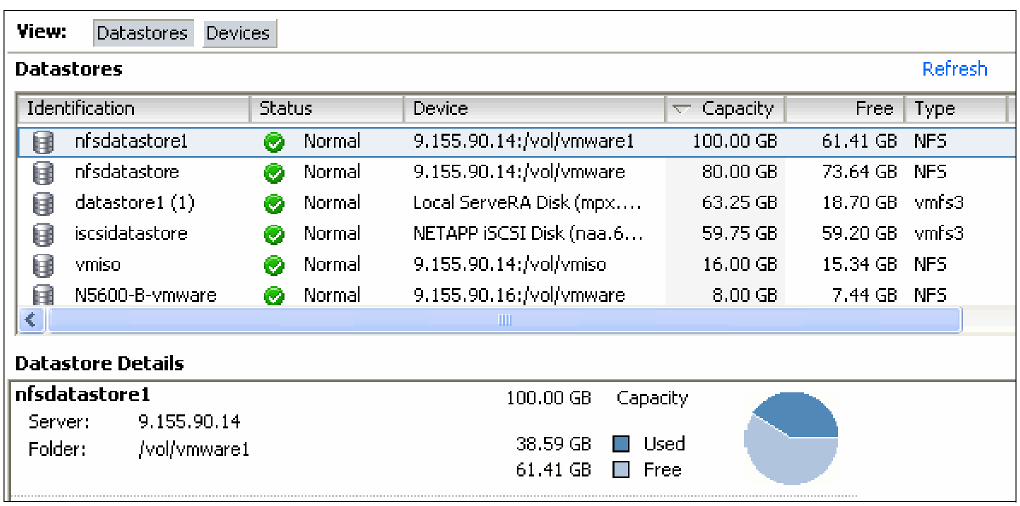


Figure 7-37 VMware vCenter storage view

7.7.2 Only deduplication

Included in the IBM N series offer is the possibility to deduplicate data on a per volume basis. To turn on deduplication on your volume, see Example 7-11.

Example 7-11 Turn on deduplication on volume vmware1

```
N5600-A> sis on /vol/vmvmware1
SIS for "/vol/vmware1" is enabled.
Already existing data could be processed by running "sis start -s /vol/vmware1".
```

As stated in Example 7-11 on page 99, to start the deduplication, you must run the command **sis start -s <path-to-your-volume>** to start processing the existing data on the volume (see Example 7-12 for the results).

Example 7-12 Start the deduplication process

```
N5600-A> sis start -s /vol/vmware1
The file system will be scanned to process existing data in /vol/vmware1.
This operation may initialize related existing metafiles.
Are you sure you want to proceed (y/n)? y
The SIS operation for "/vol/vmware1" is started.
[N5600-A: wafl.scan.start:info]: Starting SIS volume scan on volume vmware1.
```

You can monitor the status of the deduplication process by using the command **sis status (**Example 7-13).

Example 7-13 Monitoring the deduplication process

```
N5600-A> sis status
Path                           State      Status     Progress
/vol/vmiso                     Enabled    Idle       Idle for 00:03:32
/vol/vmware                    Enabled    Idle       Idle for 13:19:15
/vol/vmware1                   Enabled    Active     11 GB Scanned
N5600-A> sis status
Path                           State      Status     Progress
/vol/vmiso                     Enabled    Idle       Idle for 00:08:24
/vol/vmware                    Enabled    Idle       Idle for 00:02:37
/vol/vmware1                   Enabled    Active     10 GB Searched
N5600-A> sis status
Path                           State      Status     Progress
/vol/vmiso                     Enabled    Idle       Idle for 00:09:18
/vol/vmware                    Enabled    Idle       Idle for 00:03:31
/vol/vmware1                   Enabled    Active     235 MB (6%) Done
```

After the process is complete, we can check the results in the IBM N series console and the VMware vCenter GUI.

As shown in Example 7-14, deduplication saved 53% of the space that the virtual machines took. Instead of 40% of the volume, the files only use 18% of the volume.

Example 7-14 Issuing the command df -k and df -Sk on the N series console

```
N5600-A> df -k
Filesystem              total       used      avail capacity  Mounted on
/vol/vol0/          25165824KB  1437136KB 23728688KB       6%  /vol/vol0/
/vol/vmiso/         16777216KB   690196KB 16087020KB       4%  /vol/vmiso/
/vol/vmware1/       104857600KB 19047572KB 85810028KB      18%  /vol/vmware1/
N5600-A> df -Sk
Filesystem               used      compressed          a-sis     %saved
/vol/vol0/          1437148KB             0KB            0KB         0%
/vol/vmiso/          690196KB             0KB            0KB         0%
/vol/vmware1/       19047572KB            0KB     21415064KB        53%
```

The VMware vCenter GUI only shows the free space, which is where we notice that indeed there is about 82% of free space in the volume (Figure 7-38).

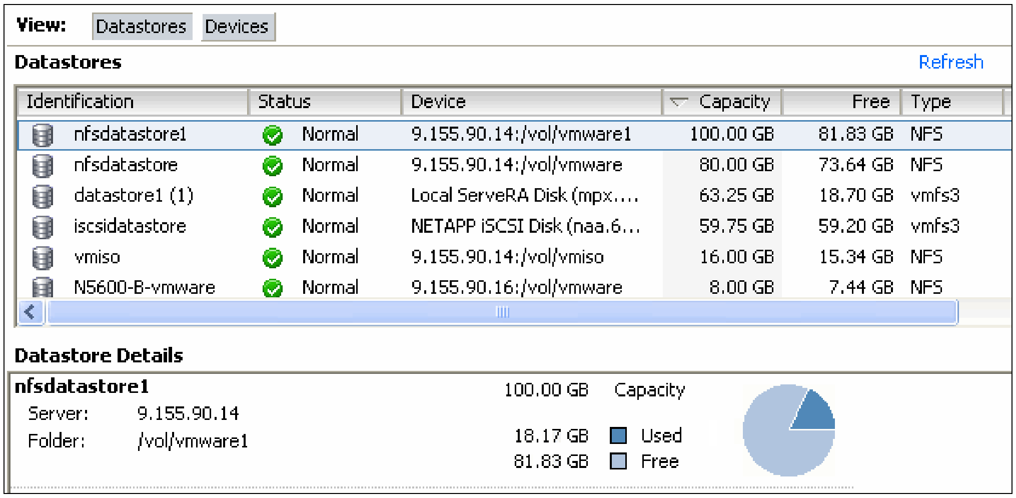


Figure 7-38 VMware vCenter storage view

7.7.3 Turning on compression

When we turn on compression and run the Compression Accelerator daemon, the deduplication gains we previously had disappear. This is due to the fact that the Compression Accelerator daemon rewrites all uncompressed data as compressed data.
The storage will see this as newly written data, and thus deduplication, which is a background process, and snapshots, will be influenced (Example 7-15).

Example 7-15 Issuing the command df -k and df -Sk on the IBM N series console

```
N5600-A> df -k
Filesystem              total        used       avail capacity  Mounted on
/vol/vol0/          25165824KB  1436852KB 23728972KB        6%  /vol/vol0/
/vol/vmiso/         16777216KB   690196KB 16087020KB        4%  /vol/vmiso/
/vol/vmware1/       104857600KB 11300908KB 93556692KB      11%  /vol/vmware1/
N5600-A> df -Sk
Filesystem               used        compressed           a-sis      %saved
/vol/vol0/          1436856KB               0KB             0KB          0%
/vol/vmiso/          690196KB               0KB             0KB          0%
/vol/vmware1/       11300908KB              0KB            12KB          0%
```

As shown in Example 7-15, the saved space we used to have using deduplication was reduced to 0%. The used space is only 11%, which is better than when only using deduplication. In the VMware storage view, it looks like Figure 7-39 on page 102.

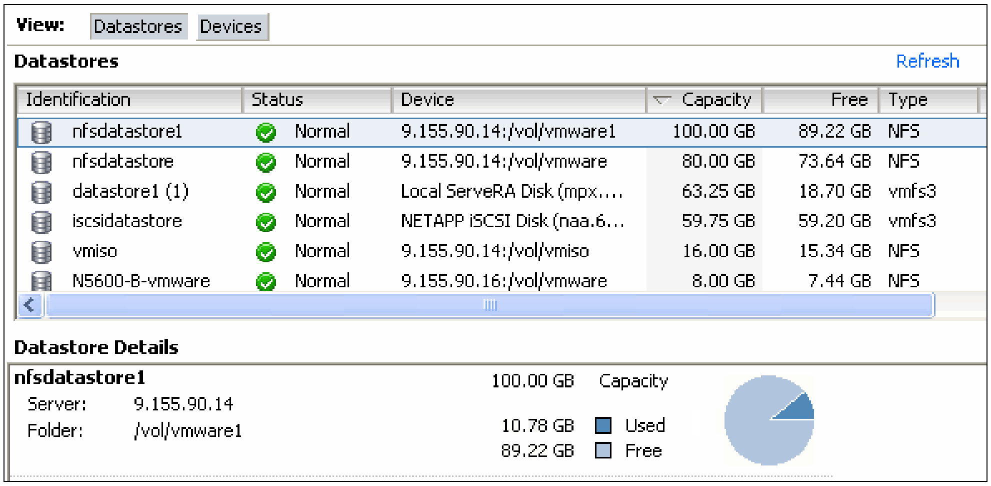


Figure 7-39 VMware vCenter storage view

7.7.4 Combining compression and deduplication

After the compression completes, we can restart the deduplication process. Because the compressed blocks are the same as the uncompressed blocks, only smaller, the effect of deduplication will be a complement to the saved space (Example 7-16).

> **Tip:** Because we rewrote all of the data, the existing data is no longer valid, and we must run the **`sis start`** command with the **`-s`** option. The **`-s`** option ensures that all existing data is rescanned and thus provides a better deduplication ratio as new data is written.

Example 7-16 Restarting deduplication on the vmware1 volume

```
N5600-A> sis start -s /vol/vmware1
The file system will be scanned to process existing data in /vol/vmware1.
This operation may initialize related existing metafiles.
Are you sure you want to proceed (y/n)? y
The SIS operation for "/vol/vmware1" is started.
[N5600-A: wafl.scan.start:info]: Starting SIS volume scan on volume vmware1.
```

Again, we can monitor the process of the deduplication, as shown in Example 7-13 on page 100. The result of the compression and deduplication together is shown in Example 7-17.

Example 7-17 Issuing the command df -k and df -Sk on the IBM N series console

```
N5600-A> df -k
Filesystem              total       used      avail capacity  Mounted on
/vol/vol0/          25165824KB  1437008KB 23728816KB       6%  /vol/vol0/
/vol/vmiso/         16777216KB   690196KB 16087020KB       4%  /vol/vmiso/
/vol/vmware1/       104857600KB  8079108KB 96778492KB       8%  /vol/vmware1/
N5600-A> df -Sk
Filesystem               used       compressed          a-sis      %saved
/vol/vol0/           1437004KB              0KB            0KB          0%
/vol/vmiso/           690196KB              0KB            0KB          0%
/vol/vmware1/        8069500KB              0KB      3242420KB         29%
```

Figure 7-40 shows the result seen from a VMware point of view.

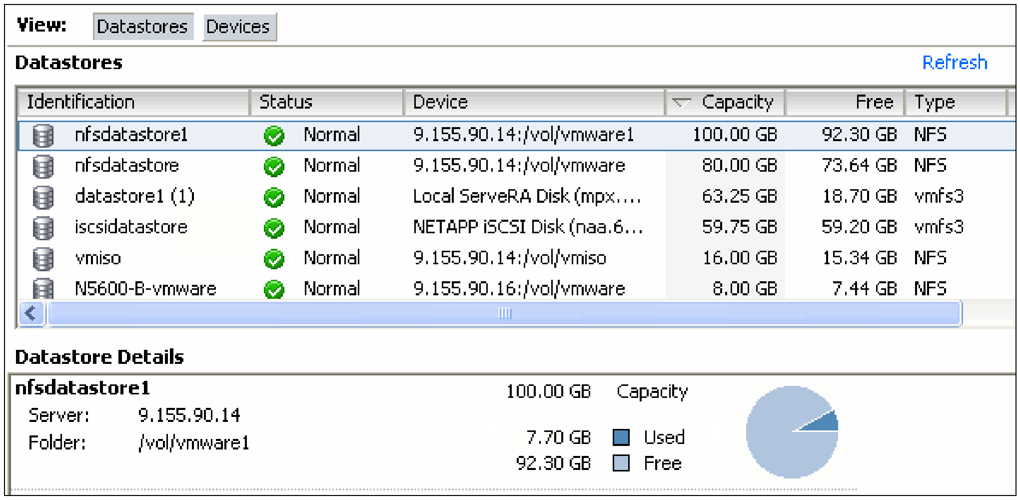


Figure 7-40 VMware vCenter storage view

7.7.5 Conclusion

In Table 7-1, we have summarized the results of all the tests. It shows clearly that the compression results are compatible with the deduplication results and that customers can benefit from using both technologies together.

Table 7-1 Test results

Space used	Compression	Deduplication	Totals
38.59 GB	0%	0%	0%
18.17 GB	0%	53%	53%
10.78 GB	72%	0%	72%
7.70 GB	72%	29%	80%

Results: These results are just an example. The actual results might differ from these results, depending on the type of data you have and the compression rate of this data.

8

Oracle use cases

This chapter provides a practical approach about how to implement the IBM Real-time Compression Appliance when implementing an Oracle database on Red Hat Enterprise Linux (RHEL) with NFS using the N series as a storage foundation.

We first start by describing the setup of our test environment and then go on with the implementation.

8.1 Implementation for RTCA

This chapter deals with multiple concepts common to an enterprise environment. The reader is assumed to have a basic understanding of UNIX platforms (Linux), NAS environments (NFS), databases (Oracle) and N series storage systems.

Treat this document as a starting reference point. It gives the bare minimum requirements for deployment of Oracle on N series using RTCA. It is intended as a reference for compression.

8.1.1 Defining the test environment

This document provides design implementation for testing on native Operating System NFS on the Oracle 11g R2 with Linux RHEL 5.7.

You can use this set of techniques to implement compression in your database environment.

In this section, we provide three sets of testing cases:

- Data set already implemented before the RTCA
- Data set generated/imported after implementation of the RTCA
- Recovery process if the RTCA has a problem or is not being used anymore

Generally, these are the steps we implement:

1. Install the Oracle database. Configure the datafile using NFS export to the storage. Prepare and populate it with generated data. The data is generated using data model for TPC-H with scale 20. For existing customers that already use an Oracle database, you usually have your own data sets to simulate the characteristics of your business process.

   We also generate some data that is using Oracle OLTP Compression.

2. Implement nondisruptive RTCA with transparent mode. NFS export configured on the RTCA with NNC (No New Compression). This will make sure that nothing changed on the transferred data (Without Compression).

3. Activate compression. This can be either using the Compression Accelerator to compress existing data or fresh loaded data.

   The Real-time Compression Appliance (RTCA) software compresses new files written through it to a share or export that is configured as compressed in the compression filters list. If a file exists in this share or export prior to the activation of export filters, it is not automatically compressed. To compress such files, the software's Compression Accelerator utility must be used. If you choose to leave it uncompressed, it will remain uncompressed even when updated through the software.

   > **Tip:** Any Oracle datafile that was created before activating RTCA will stay uncompressed. Optionally, you can enable the Compression Accelerator for a particular export path, or migrate to a new datafile.

4. Manually recover the data if it had some problem or you decide to run without the RTCA.

Environment

The environment includes:

- IBM System x® x3650-m2 running on Linux RHEL 5.7.

  We adjust Linux OS settings for Oracle. For production database systems, Oracle recommends to tune these values to optimize the performance of the system. See the operating system documentation for more information about tuning kernel parameters.

  Verify that the kernel parameters shown in Table 8-1 are set to values greater than or equal to the recommended value shown.

Table 8-1 Linux kernel parameter

Parameter	Value	File
semmsl semmns semopm semmni	250 32000 100 128	/proc/sys/kernel/sem
shmall	2097152	/proc/sys/kernel/shmall
shmmax	Half the size of physical memory (in bytes). For more information, see the My Oracle Support Note 567506.1.	/proc/sys/kernel/shmmax
shmmni	4096	/proc/sys/kernel/shmmni
file-max	6815744	/proc/sys/fs/file-max
ip_local_port_range	Minimum: 9000 Maximum: 65500	/proc/sys/net/ipv4/ip_local_port_range
rmem_default	4194304	/proc/sys/net/core/rmem_default
rmem_max	4194304	/proc/sys/net/core/rmem_max
wmem_default	262144	/proc/sys/net/core/wmem_default
wmem_max	1048576	/proc/sys/net/core/wmem_max
aio-max-nr	1048576	/proc/sys/fs/aio-max-nr

Tip: If the current value for any parameter is higher than the value listed in this table, do not change the value of that parameter.

To improve the performance of the software on Linux systems, we also increase the following shell limits for the Oracle user as shown in Table 8-2.

Table 8-2 Setting shell limits for the Oracle user

Shell limit	Item in limits.conf	Hard limit
Maximum number of open file descriptors	nofile	65536
Maximum number of processes available to a single user	nproc	16384

Setting the right NFS mount options can significantly affect both performance and reliability of the I/O subsystem. N series recommended mount options have been thoroughly tested and approved by both N series and Oracle, as shown in Table 8-3.

Table 8-3 Mount options

NFS location	Mount options
Mount options for binaries	rw,bg,hard,rsize=65536,wsize=65536,vers=3,nointr,timeo=600, tcp
Mount options for Oracle datafiles	rw,bg,hard,rsize=65536,wsize=65536,vers=3,nointr,timeo=600, tcp
Mount options for ADR_HOME (11g only)	rw,bg,hard,rsize=65536,wsize=65536,vers=3,nointr,suid, timeo=600, tcp
init.ora parameters	filesystemio _options=setall or directio

- Oracle libraries and binaries located on local file system. Oracle datafile, archive-log, and redo-log configured using NFS exported from N5600 storage. See Table 8-4 for Oracle layout details.

Table 8-4 Oracle layout

Type of files	Description	Mount point	Location
ORACLE_BASE		/u01/app/oracle	local filesystem
ORACLE_HOME	Oracle libraries and binaries	/u01/app/oracle/11.2.0 /dbs	Local filesystem
Data files	Oracle Database datafiles	/export/datafile/<tables pace>/	NFS N series Storage
Log files	Oracle redo and archive logs	/export/logfile /export/flashrecovery	NFS N series Storage

- N5600 with version 7.3.6P1. See Example 8-1.

Example 8-1 N5600 system configuration

```
N5600-A> sysconfig
        Data ONTAP Release 7.3.6P1: Wed Aug 10 10:53:02 PDT 2011 (IBM)
        System ID: 0118054991 (N5600-A); partner ID: 0118054908 (N5600-B)
        System Serial Number: 2868130013711 (N5600-A)
        System Rev: B4
        System Storage Configuration: Single-Path HA
        System ACP Connectivity: NA
        slot 0: System Board
                Processors:         4
                Memory Size:        8192 MB
                Memory Attributes:  Node Interleaving
                                    Bank Interleaving
                                    Hoisting
                                    Chipkill ECC
        Remote LAN Module           Status: Online
```

- 4. RTCA with version 3.7.1; see Example 8-2 for details.

Example 8-2 RTCA version

```
user@9.155.66.157> version
Model:                      STN6800
Version:                    3.7.1.01 (NetCompDefaults)
Compression Engine:         (r,a) (Build 37117)
SNMP:                       (r) (Build 37117)
MIB applianceHardware.mib: 2.0.7
MIB applianceMib.mib:      2.0.0
MIB applianceSoftware.mib: 2.1.3
Web GUI:                    3.7.1.0028
Misc Version:               3.7.0
Serial Number:              6I6056F2MOEC94500932
Hostname:                   RtCA1
```

On the N series N5600, we implement a high-availability schema so that we can add RTCA without any disruption to the Oracle database service. We configure vif1 and vif2 using LACP. Then we configure vif0 as single mode to provide high-availability configuration. See Example 8-3 for the VIF configuration on the N5600-A. We configure a similar configuration for the N5600-B.

Example 8-3 The VIF configuration on N5600-A

```
N5600-A> vif status
default: transmit 'IP Load balancing', VIF Type 'multi_mode', fail 'log'
vif1: 2 links, transmit 'IP Load balancing', VIF Type 'lacp' fail 'default'
     VIF Status    Up     Addr_set
    trunked: vif0
    up:
        e0c: state up, since 24Oct2011 12:16:26 (01:45:31)
                mediatype: auto-1000t-fd-up
                flags: enabled
                active aggr, aggr port: e0a
                input packets 4708962, input bytes 1381626010
                input lacp packets 679, output lacp packets 906
                output packets 10656337, output bytes 12410365595
                up indications 11, broken indications 8
                drops (if) 0, drops (link) 0
                indication: up at 24Oct2011 12:16:26
                        consecutive 0, transitions 19
        e0a: state up, since 24Oct2011 12:16:25 (01:45:32)
                mediatype: auto-1000t-fd-up
                flags: enabled
                active aggr, aggr port: e0a
                input packets 6825587, input bytes 7579951119
                input lacp packets 679, output lacp packets 902
                output packets 1803369, output bytes 2518152876
                up indications 12, broken indications 9
                drops (if) 0, drops (link) 0
                indication: up at 24Oct2011 12:16:25
                        consecutive 0, transitions 21
vif2: 2 links, transmit 'IP Load balancing', VIF Type 'lacp' fail 'default'
     VIF Status    Up     Addr_set
    trunked: vif0
```

```
    down:
        e0b: state up, since 24Oct2011 12:22:31 (01:39:26)
                mediatype: auto-1000t-fd-up
                flags: enabled
                active aggr, aggr port: e0d
                input packets 3011, input bytes 449859
                input lacp packets 665, output lacp packets 1317
                output packets 4613, output bytes 301860
                up indications 22, broken indications 19
                drops (if) 0, drops (link) 0
                indication: up at 24Oct2011 12:22:31
                        consecutive 0, transitions 41
        e0d: state up, since 24Oct2011 12:22:30 (01:39:27)
                mediatype: auto-1000t-fd-up
                flags: enabled
                active aggr, aggr port: e0d
                input packets 46357, input bytes 2940349
                input lacp packets 664, output lacp packets 1312
                output packets 4623, output bytes 302120
                up indications 17, broken indications 14
                drops (if) 0, drops (link) 0
                indication: up at 24Oct2011 12:22:30
                        consecutive 0, transitions 31
vif0: 1 link, transmit 'none', VIF Type 'single_mode' fail 'default'
     VIF Status    Up     Addr_set
    up:
        vif1: state up, since 24Oct2011 12:16:25 (01:45:32)
                mediatype: Enabled virtual interface
                flags: enabled favored
                input packets 11534538, input bytes 8961576401
                output packets 12459698, output bytes 14928517935
                output probe packets 6558, input probe packets 6554
                strike count: 0 of 10
                up indications 2, broken indications 1
                drops (if) 0, drops (link) 0
                indication: up at 24Oct2011 12:16:25
                        consecutive 20079, transitions 4
    down:
        vif2: state down, since 24Oct2011 12:22:30 (01:39:27)
                mediatype: Enabled virtual interface
                flags: enabled
                input packets 49367, input bytes 3390148
                output packets 9236, output bytes 603980
                output probe packets 6558, input probe packets 6554
                strike count: 0 of 10
                up indications 2, broken indications 1
                drops (if) 0, drops (link) 0
                indication: up at 24Oct2011 12:22:30
                        consecutive 19723, transitions 4
```

8.1.2 Database population with TPC-H data model

For testing purposes, in this chapter, we create load scenarios from TPC-H. These synthetic workloads can be used when no real-world data is available to be imported for database load testing.

The Transaction Processing Performance Council (TPC) defines transaction processing and database benchmarks and delivers trusted results to the industry. While the TPC-H is a decision support benchmark, it consists of a suite of business oriented ad-hoc queries and concurrent data modifications.

The version used for this testing is TPC-H version 2.14.2.

For more information, see the following website:

http://www.tpc.org

For those of you who are not familiar with the TPC-H model, see Figure 8-1 to have an idea of what it looks like.

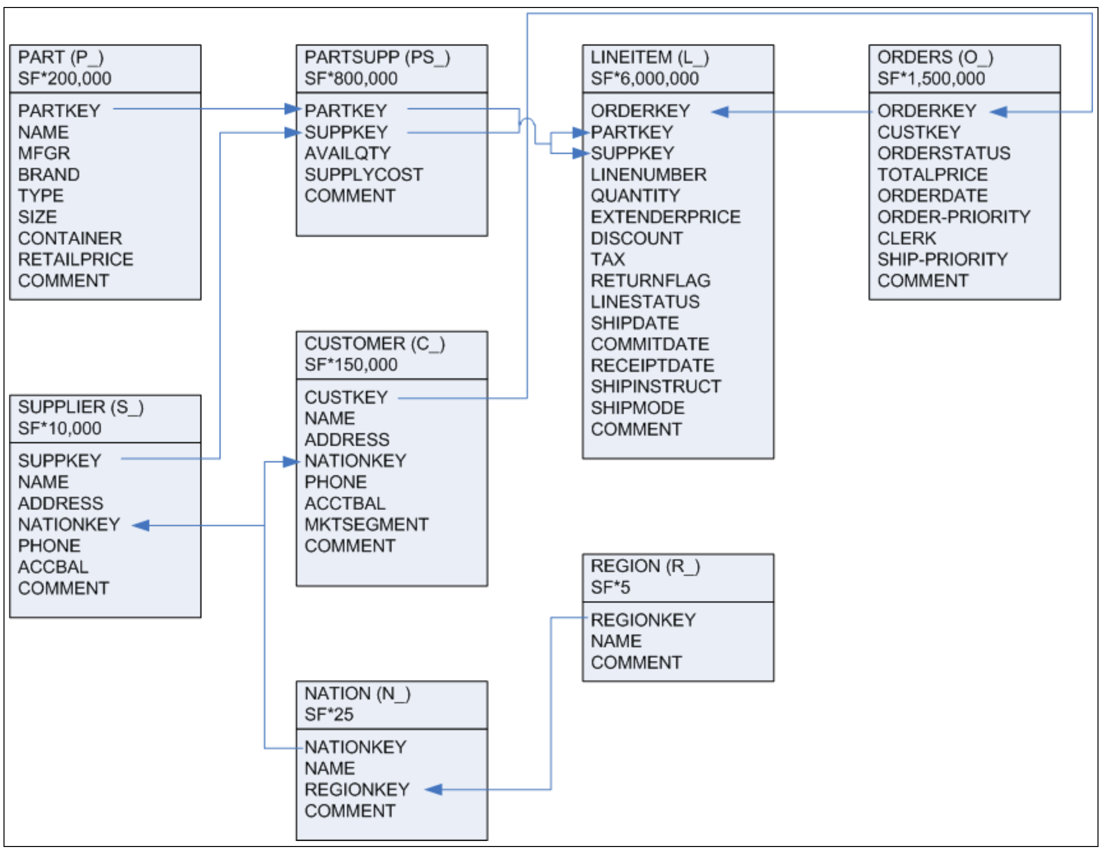


Figure 8-1 TPC-H Schema

Benchmark Scale Factors that are used for testing environment are 1, 20, and 100. The load scenario can be seen in Table 8-5.

Table 8-5 Load Scenario Estimated for Scale Factor 20

Table name	Rows	Bytes/row	Total bytes
H_Part	20*200,000 = 4,000,000	155	591,28MB

Table name	Rows	Bytes/row	Total bytes
H_Supplier	20*10,000 = 200,000	159	30,33MB
H_Partsupp	20*800,000 = 16,000,000	144	2,15GB
H_Customer	20*150,000 = 3,000,000	179	512,12MB
H_Order	20*1,500,000 = 30,000,000	104	2,91GB
H_Lineitem	20*6,000,000 = 120,000,000	112	12,52GB
H_Nation	25	128	3,13KB
H_Region	5	124	620

8.1.3 Generated data set

In this testing environment, we generate data set using tools dbgen provided from the TPC website. Data generated using TPC-H data sets can be seen in Example 8-4. Each table space only contains one Oracle datafile.

Example 8-4 Table space created for compression testing

```
SQL> select * from v$tablespace;

       TS# NAME                           INC BIG FLA ENC
---------- ------------------------------ --- --- --- ---
         0 SYSTEM                         YES NO  YES
         1 SYSAUX                         YES NO  YES
         2 UNDOTBS1                       YES NO  YES
         4 USERS                          YES NO  YES
         3 TEMP                           NO  NO  YES
         6 EXAMPLE                        YES NO  YES
        14 SH4                            YES NO  YES
         9 SH                             YES NO  YES
        12 SH3                            YES NO  YES
         8 SOE                            YES NO  YES
         7 CCDATA                         YES YES YES

       TS# NAME                           INC BIG FLA ENC
---------- ------------------------------ --- --- --- ---
        15 SH5                            YES NO  YES

12 rows selected.

SQL> select tablespace_name,file_name from dba_data_files;

TABLESPACE_NAME
------------------------------
FILE_NAME
--------------------------------------------------------------------------------
USERS
/u01/app/oracle/oradata/orcl/users01.dbf

UNDOTBS1
/u01/app/oracle/oradata/orcl/undotbs01.dbf
```

```
SYSAUX
/u01/app/oracle/oradata/orcl/sysaux01.dbf


TABLESPACE_NAME
------------------------------
FILE_NAME
--------------------------------------------------------------------------------
SYSTEM
/u01/app/oracle/oradata/orcl/system01.dbf

EXAMPLE
/u01/app/oracle/oradata/orcl/example01.dbf

SH4
/export/datafile/sh/sh4/sh4.dbf


TABLESPACE_NAME
------------------------------
FILE_NAME
--------------------------------------------------------------------------------
SH
/export/datafile/sh/sh/sh.dbf

SH3
/export/datafile/sh/sh3/sh3.dbf

SOE
/export/datafile/soe/soe.dbf


TABLESPACE_NAME
------------------------------
FILE_NAME
--------------------------------------------------------------------------------
CCDATA
/export/datafile/cc/ccdata.dbf

SH5
/export/datafile/sh/sh5/sh5.dbf

USERS
/u01/app/oracle/oradata/orcl/users02.dbf


12 rows selected.
```

Table 8-6 on page 114 shows information about the purpose of each table space that we use for testing:

Tablespace SH Data set is loaded using TPC-H with scalefactor 20. Oracle OLTP compression is activated. This tablespace will be compressed using the RTCA Compression Accelerator.

Tablespace SH3 Data set is loaded using TPC-H with scale factor 20. In this tablespace, we did not activate Oracle OLTP compression. This tablespace will be compressed using RTCA Compression Accelerator. We can compare compression ration with tablespace SH.

Tablespace SH4 On the SH4 tablespace, we can see how RTCA manages to perform real time compression since the first time the datafile was created. This tablespace will be created later after RTCA implemented and compression enabled to the exported path.

Tablespace SH5 We only use Oracle OLTP compression with transparent mode to the RTCA. We can then compare with SH, SH3, SH4 table spaces to give us information regarding the compression ratio.

Tablespace CCDATA We populate data using scale factor 100 to reach more than 100 GB data to show usage for larger datafiles.

Tablespace SOE We show you how to recover data when the RTCA is unavailable, or if you decide not to use the RTCA again.

Table 8-6 Oracle tablespace information

Tablespace	Data generator used	Compression mode	Table space purpose
SH	TPC-H scale 20	oracle OLTP compression	Effectiveness of implementing RTCA *using* Oracle OLTP compression enabled.
SH3	TPC-H scale 20	none	Effectiveness of implementing RTCA *without* enabling Oracle OLTP compression.
SH4	TPC-H scale 20	none	Data will be loaded after RTCA ready.
SH5	TPC-H scale 20	oracle OLTP compression	Only using Oracle OLTP without RTCA enabled. No compression on RTCA.
CCDATA	TPC-H scale 100	none	How Compression Accelerator handled more than 100 GB datafiles.
SOE	TPC-H scale 1	none	Recovery.

Table 8-7 shows the tablespace size used after generating the data from TPC-H data sets.

Table 8-7 Size Used on the Oracle before implementing RTCA

Table space	Data generator used	Oracle compression mode	Allocated size on Oracle (MB)	Space used on Oracle (MB)	Compression ratio using Oracle OLTP compression
SH	TPC-H scale 20	oracle OLTP compression	23,117.0	21,980.0	14%
SH3	TPC-H scale 20	none	27,085	25,767	0%
SH4	none	none	-	-	-
SH5	TPC-H scale 20	oracle OLTP compression	23,117	21,980.0	14%
CCDATA	TPC-H scale 100	none	136,084	129,574	-

We calculate the ratio for Oracle OLTP compression by comparing Space Used on Oracle with and without using Oracle OLTP compression.

8.1.4 Checking the capacity used for the Oracle database

The majority of Oracle database size comes from the datafiles. We can check from datafiles to see individual file size and also how much size was already allocated for data. Later on we can manage to shrink back / reclaim the file size after doing some major changes that involve deleting the data.

To obtain how big the database is, we can do the following calculation:

```
database size = sum (datafile) + redo logs + archive logs
```

If the database is in archive log mode, we add archive logs because this also normally consumes space also. In Example 8-5, we can check how big our database is.

Example 8-5 Check database size

```
SQL> select sum(bytes)/1024/1024 from dba_data_files;

SUM(BYTES)/1024/1024
--------------------
              257306

SQL> select sum(bytes)/1024/1024 from v$log;

SUM(BYTES)/1024/1024
--------------------
                 150

SQL> select sum(space_used)/1024/1024 from v$recovery_file_dest;

SUM(SPACE_USED)/1024/1024
-------------------------
               134524.639
```

We can achieve a great compression ratio when enabling the compression mode on the archive log mount point. As for the simplicity of this project, we only calculate the Oracle datafiles as measurement for compression ratio.

8.2 Using the RTCA product

In this test case, we implement the RTCA product without any disruption to the service. Planning and verifying the configuration must be done before inserting the compression appliance.

8.2.1 Inserting the RTCA product into an existing environment

Here we provide a simple checklist to use for an easy setup:

- RTCA

  Configure storage compression mode as transparent mode. With this mode, later on we can easily add which export path that want to be enabled.

  Configure bond and bridge that match the high availability configuration for network storage topology. Active compression engine is used.

- N series

  Verify VIF status because this is the most important part that will keep client/user access to the storage for nondisruptive implementation. We will want to check the passive path location.

- Network Switch

  For active-active link aggregation (EtherChannel or 802.3), also configure a group channel in the network switch. A configuration diagram for the test environment before inserting the RTCA can be seen in Figure 8-2.

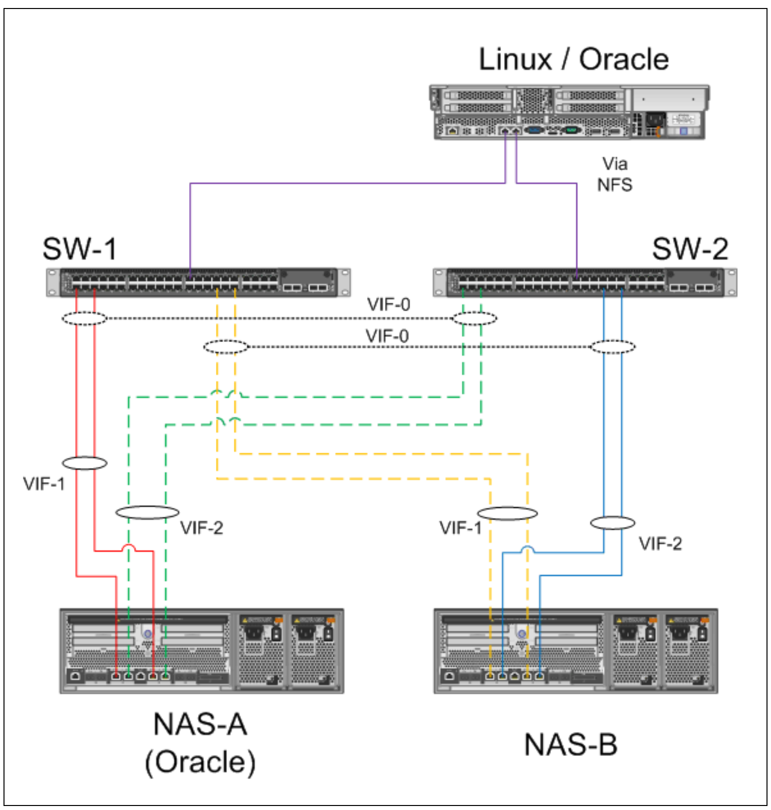


Figure 8-2 Network Configuration before inserting RTCA

After finishing with the set up preparations for the solution, we are ready to insert the RTCA as a smart cable:

1. Unplug passive cables on the N series. Reroute those cables to the network switch (pre-configured ports if active-active link aggregation), and then plug additional cables from the RTCA to the network switch. See Figure 8-3.

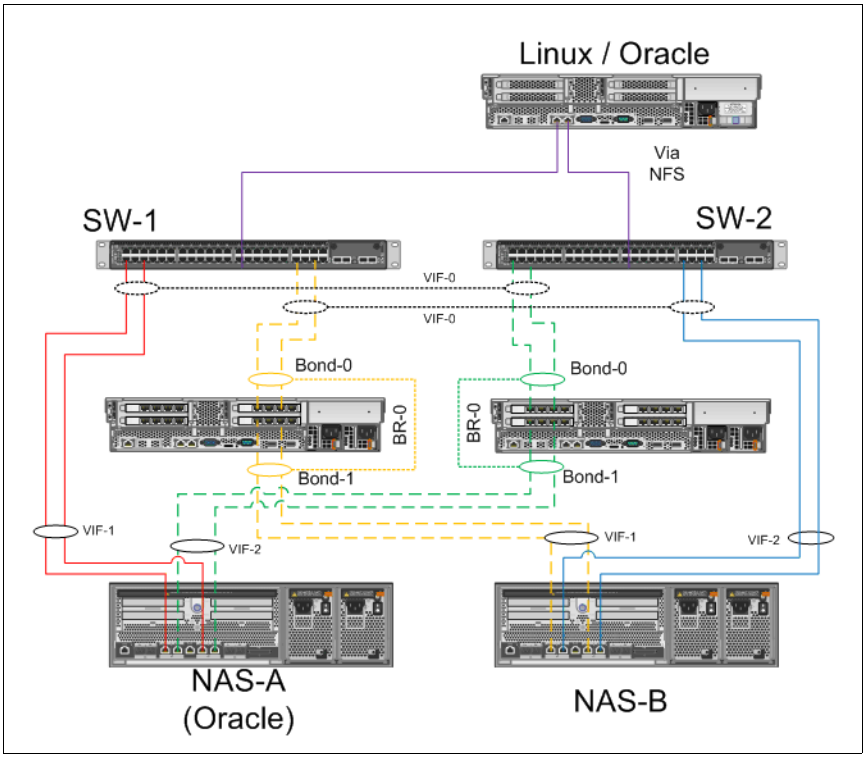


Figure 8-3 Reroute passive VIF

2. Verify that all traffic from the database server is still OK. Verify either from the alert log if the database is still up and running, and also from Linux /var/log/messages if there is an NFS time out error detected. Verify also the VIF status on the N series.
3. Configure storage on RTCA as transparent mode (see Figure 8-4).

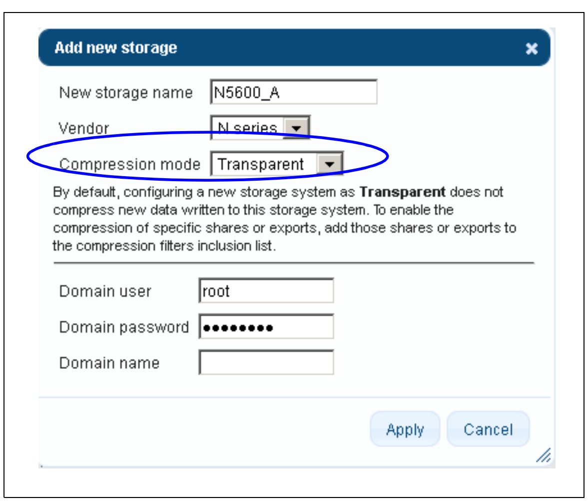


Figure 8-4 Add storage as transparent mode

4. From the N series, perform VIF failover so that active path will use a route to the RTCA (Figure 8-5).

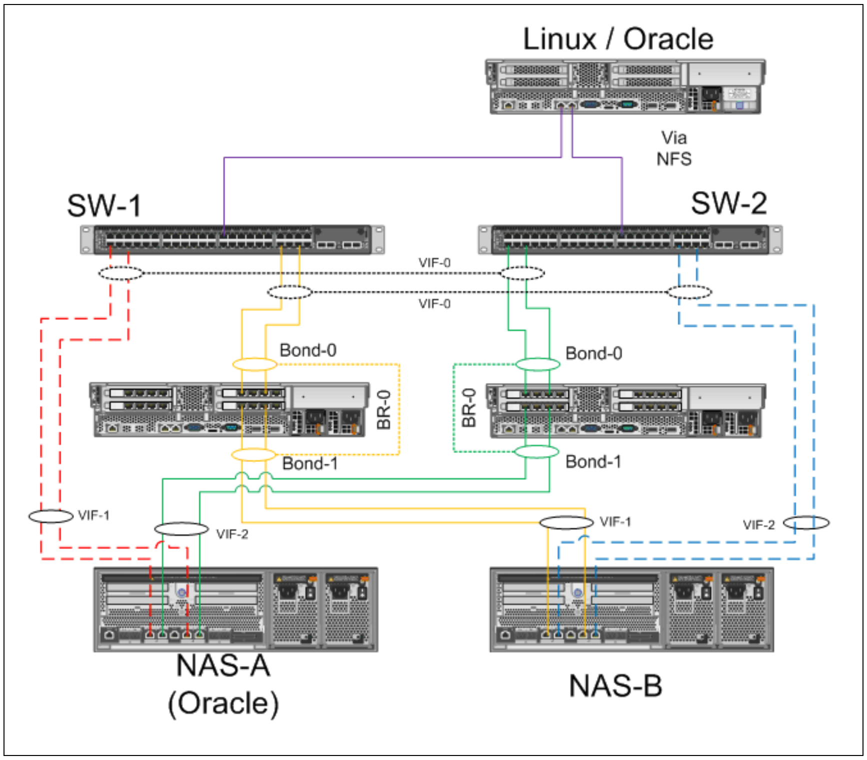


Figure 8-5 Perform VIF failover

5. Repeat this for the other path to the other RTCA for high availability topology.
6. Repeat this for the next N series controller.

The network topology after inserting the RTCA can be seen in Figure 8-6.

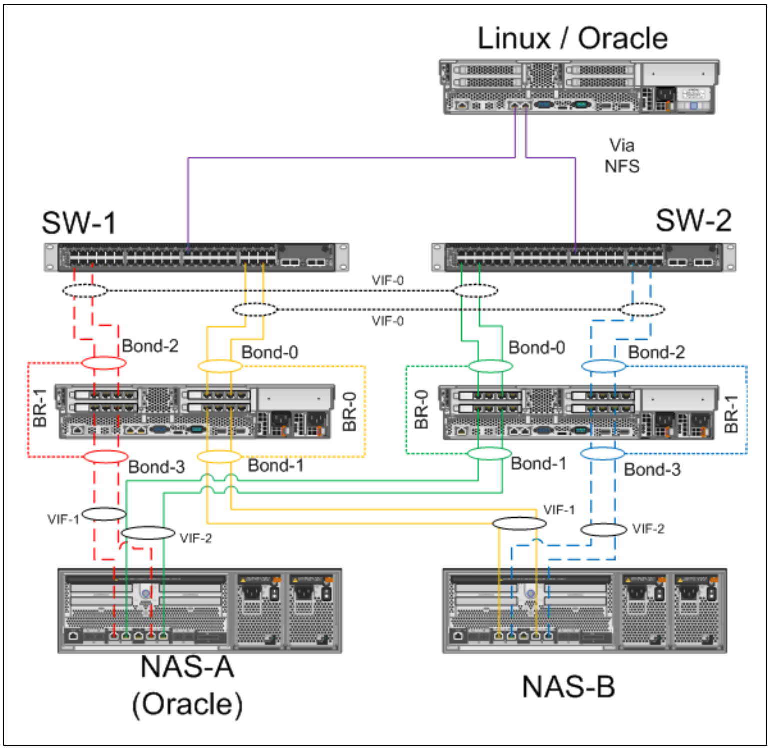


Figure 8-6 Network topology after inserting RTCA

The physical activities are already done. After this step, any I/O request from the host client to the N series will reroute to the RTCA in transparent mode (uncompress).

8.2.2 Enabling compression on the RTCA

Because the storage is already configured in transparent mode, next we configure each export path on the N series and RTCA.

On the N series, for each exported path that needs compression enabled, add all the RTCA bridge IP addresses for read/write access. For the Compression Accelerator to work, also add all RTCA bridge IP addresses for root access.

Example 8-6 is an example of one exported path that we add for the bridge IP address for read/write and root access. 9.155.113.233 is for the Linux/Oracle Server. 9.155.90.170-173 are for bridge IP addresses.

Example 8-6 Specifying exportfs access permission

```
/vol/oracle/datafile/sh/sh4
-sec=sys,rw=9.155.90.173:9.155.90.172:9.155.113.233:9.155.90.171:9.155.90.170,root
=9.155.90.173:9.155.90.172:9.155.90.171:9.155.113.233:9.155.90.170
```

All exported paths that we configured on the N series also show up on the RTCA. We select the export paths to be added to the compression filter. Deselect the No new compression option so that the newly created file can be compressed (see Figure 8-7 on page 120).

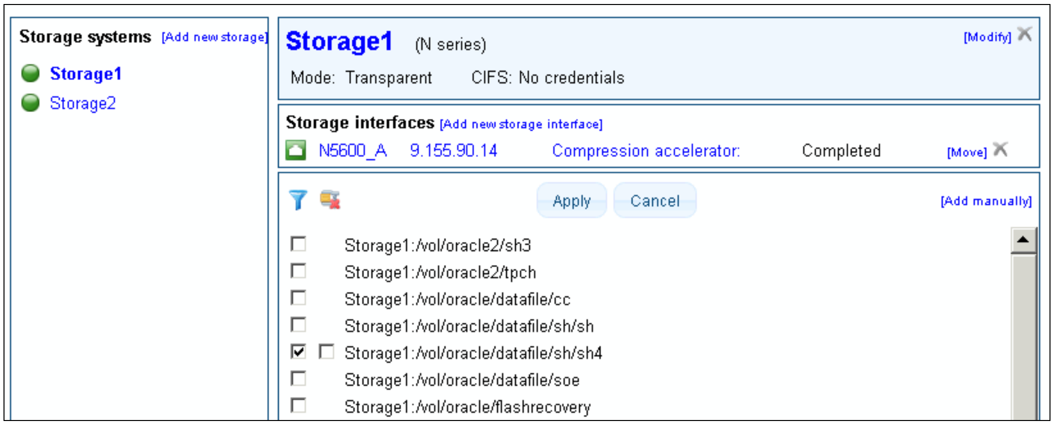


Figure 8-7 Enable compression on the RTCA

To generate the data set for the SH4 tablespace with the TPC-H scale factor 20:

1. Create a new tablespace SH4 on the Oracle database with a datafile located on a compressed export path. Create a username and password that have access to the SH4 tablespace. See Figure 8-7.

   Example 8-7 Mount point path on Linux for SH4

   ```
   9.155.90.14:/vol/oracle/datafile/sh/sh4 on /export/datafile/sh/sh4 type nfs
   (rw,bg,hard,rsize=32768,wsize=32768,nfsvers=3,timeo=600,tcp,nointr,addr=9.155.9
   0.14)
   ```

2. Populate data from TPC-H scale 20 data sets (already prepared) to tablespace SH4.

   Example 8-8 shows the result of compression for tablespace SH4.

   Example 8-8 Compression result

   ```
   SQL> select bytes/1024/1024 from dba_data_files where tablespace_name='SH4';

   BYTES/1024/1024
   ---------------              Allocated size seen on oracle datafile
             27085
   SQL> select sum(bytes)/1024/1024 from dba_segments where tablespace_name =
   'SH4';

   SUM(BYTES)/1024/1024
   --------------------          Space that already used by Oracle
                  25768

   SQL> quit
   Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -
   64bit Production
   With the Partitioning, OLAP, Data Mining and Real Application Testing options
   [root@x3650-m2-84 sh4]# pwd
   /export/datafile/sh/sh4                       Space seen on Linux mount point
   [root@x3650-m2-84 sh4]# du -m                 that served for Orace datafile
   1       ./.snapshot
   27086   .
   [root@x3650-m2-84 sh4]# pwd            Space seen on mount point with
   /mnt/oracle/datafile/sh/sh4            transparant path mode
   ```

```
[root@x3650-m2-84 sh4]# du -m
10188   .
[root@x3650-m2-84 sh4]#
```

Tip: To check real space used on the storage, for RTCA 3.7, temporarily create another export path in transparent mode.

8.2.3 Compression accelerator

The Compression Accelerator supports various configuration parameters for scheduling, throttling, and policy-based file compression that you can use to fine-tune the process. In this section, we test the Compression Accelerator on an already populated datafile.

To enable the Compression Accelerator for datafile SH:

1. Add the compression filter to the exported path. See Figure 8-8.

Figure 8-8 Add SH to compression filter

2. Select the exported path list when configuring the Compression Accelerator. See Figure 8-9.

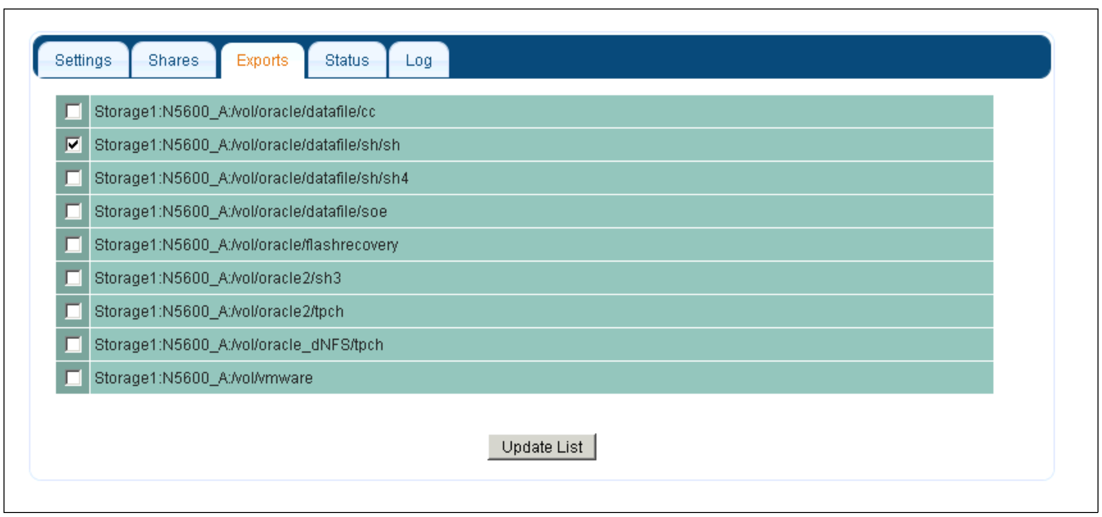


Figure 8-9 Select export path on Compression Accelerator menu

3. We start the Compression Accelerator now without any schedule (see Figure 8-10).

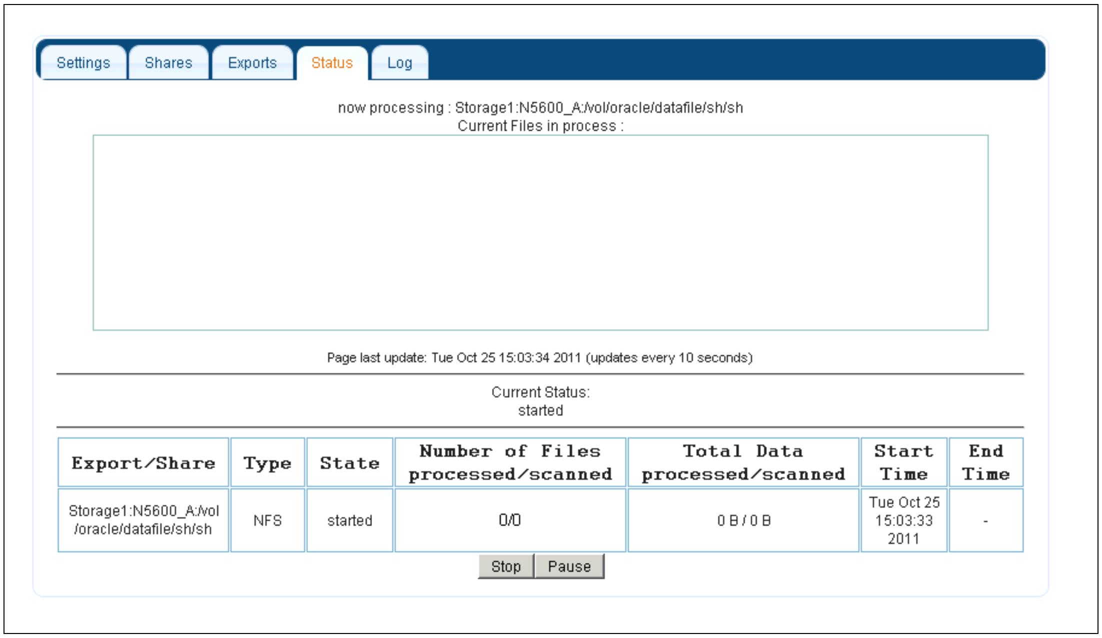


Figure 8-10 Start the Compression Accelerator

We also enable Compression Accelerator on the other datafile. Table 8-8 shows the compression result.

Table 8-8 Compression result

Table space	Data generator used	Oracle compression mode	RTCA enabled	Allocated size on Oracle (MB)	Space used on Oracle (MB)	Real space used (MB)	Compr ession ratio
SH	TPC-H scale 20	oracle OLTP compression	compression accelerator	23,117	21,980	10,533	54.4%
SH3	TPC-H scale 20	none	compression accelerator	27,085.0	25,767.0	10,144	62.5%
SH4	TPC-H scale 20	none	create new tablespace	27,085	25,769	10,190	62.3%
SH5	TPC-H scale 20	oracle OLTP compression	-	23,117	21,980	23,163	0%
CCDATA	TPC-H scale 100	none	compression accelerator	136,084	129,574	55,047	59.5%

```
Compression ratio formula = 1 - (Real Space Used / Allocated size on Oracle ) *
100 %
```

8.3 Recovering data

The compression appliance transparently and automatically compresses and decompresses data in real time, as the data is written and read by the user.

If the compression appliances is unavailable for any reason, the IBM Real-time Compression Appliance data recovery utility can recover files or directories that were previously compressed by the compression appliance.

In this section, we demonstrate how to recover tablespace SOE. See Example 8-9 for the initial location of the SOE datafile.

Example 8-9 Datafile location

```
SQL> select name from v$datafile;

NAME
--------------------------------------------------------------------------------
/u01/app/oracle/oradata/orcl/system01.dbf
/u01/app/oracle/oradata/orcl/sysaux01.dbf
/u01/app/oracle/oradata/orcl/undotbs01.dbf
/u01/app/oracle/oradata/orcl/users01.dbf
/u01/app/oracle/oradata/orcl/example01.dbf
/export/datafile/sh/sh4/sh4.dbf
/export/datafile/sh/sh/sh.dbf
/export/datafile/sh/sh3/sh3.dbf
/export/datafile/soe/soe.dbf
/export/datafile/cc/ccdata.dbf
/export/datafile/sh/sh5/sh5.dbf

NAME
--------------------------------------------------------------------------------
/u01/app/oracle/oradata/orcl/users02.dbf

12 rows selected.
```

Example 8-10 shows the tablespace SOE that is still in compressed mode with the path that used on the Oracle database. This is an illustration captured when RTCA still available.

Example 8-10 Table space SOE mounted on export compressed path

```
[root@x3650-m2-84 soe]# pwd
/export/datafile/soe
[root@x3650-m2-84 soe]# ls -l
total 3355656
-rw-r----- 1 oracle oinstall 3436191744 Oct 21 14:27 soe.dbf
```

8.3.1 Emergency path management

To recover data that previously compressed when RTCA is unavailable, proceed as follows:

1. We create another export path and connect it directly from N series storage to the Linux server without RTCA. We call this with a clear path. As we can see, because the datafile is seen in clear path, we get a smaller size than what we see when RTCA is available (see Example 8-11 on page 124).

Example 8-11 Table space SOE mounted on clear path

```
[root@x3650-m2-84 soe]# pwd
/mnt/oracle/datafile/soe
[root@x3650-m2-84 soe]# ls -l
total 661752
-rw-r----- 1 oracle oinstall 676299264 Oct 21 14:27 soe.dbf
```

> **Tip:** The recovery utility can be run on the host using a Linux or Windows operating system. It is possible to perform a parallel recovery process, if needed, to decompress many shares and export paths as long as there is no bottleneck on every part.

2. Run the RTCA data recovery utility by issuing **revert** on the installed path. See Example 8-12.

Example 8-12 Run RTCA data recovery utility

```
bash-3.2$ pwd
/opt/IBM/Real-time Compression Appliance data recovery utility
bash-3.2$ ./revert
```

After running the revert application, it will show the IBM RTCA data recovery utility. See Figure 8-11.

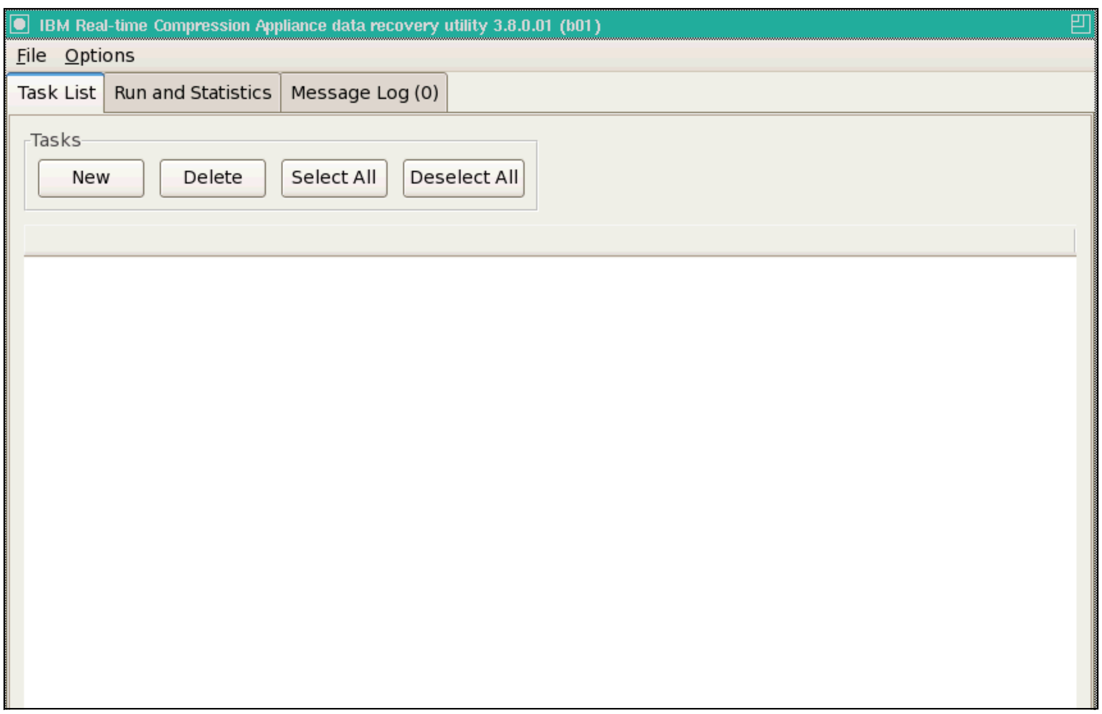


Figure 8-11 RTCA data recovery utility application

3. Create a new recovery task to recover the compressed files. We want to recover the datafile to the original location. To make these changes to the Oracle database, select **Copy in Place**. See Figure 8-12.

Important: Ensure that you have enough space to save the recovered files.

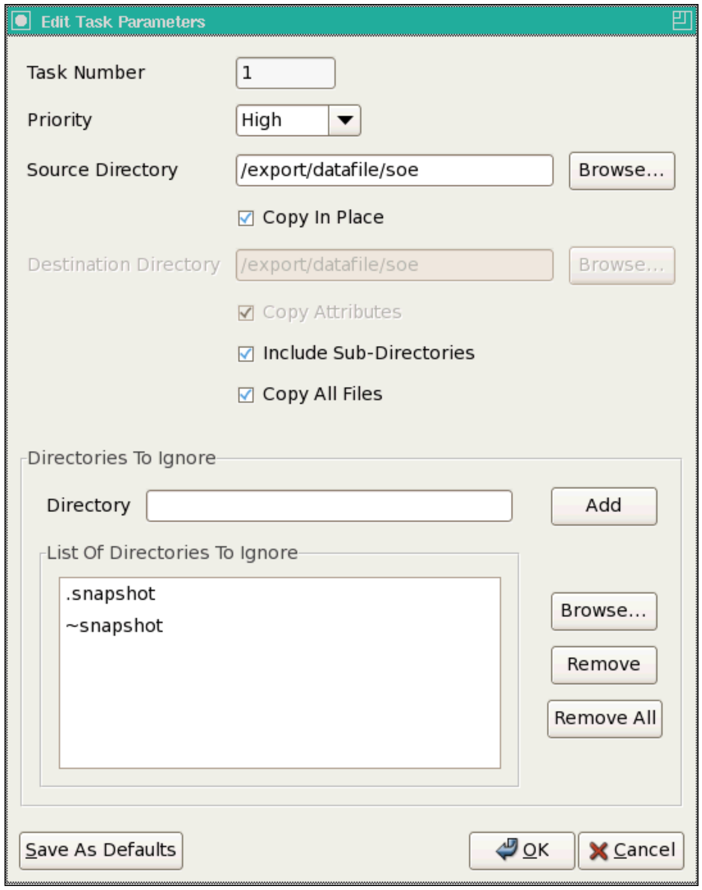


Figure 8-12 Set up new task for recovery files

4. After clicking **OK**, review, on the task list, the status of each task (Figure 8-13).

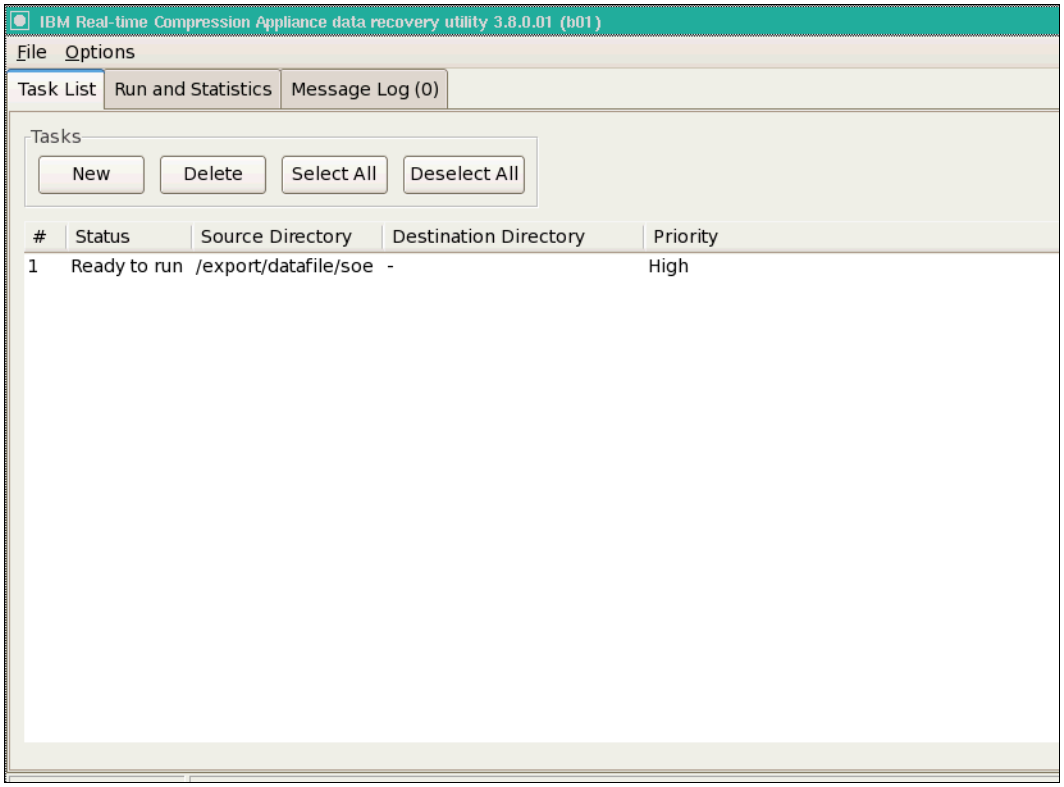


Figure 8-13 List of Task

5. On the Run and Statistics tab, click **Start** to begin the decompression process.

> **Attention:** Ensure that files that are ready to be decompressed are readable. Check from the database server if the datafile is still being locked.

We can see in Example 8-13 that the RTCA data recovery utilities use a temporary file on the same directory while processing the decompression datafile. This is the reason that we must have enough space available while recovering the data.

Example 8-13 Temporary file used while decompression in progress

```
[root@x3650-m2-84 soe]# ls -l        ------>initial file in compressed mode
total 658772
-rw-r----- 1 oracle oinstall 673251840 Oct 28 15:51 soe.dbf

[root@x3650-m2-84 soe]# ls -l        ------>temporary file used when using
total 2036068                               copy in place
-rw-r--r-- 1 oracle oinstall 1407582208 Oct 28 16:54 1403722366.soe.dbf
-rw-r----- 1 oracle oinstall  673251840 Oct 28 15:51 soe.dbf

[root@x3650-m2-84 soe]# ls -l      ------>   file in uncompressed mode
total 3362252
-rw-r----- 1 oracle oinstall 3436191744 Oct 28 15:51 soe.dbf
```

6. When the recovery task is finished, a pop-up message will be displayed. See Figure 8-14.

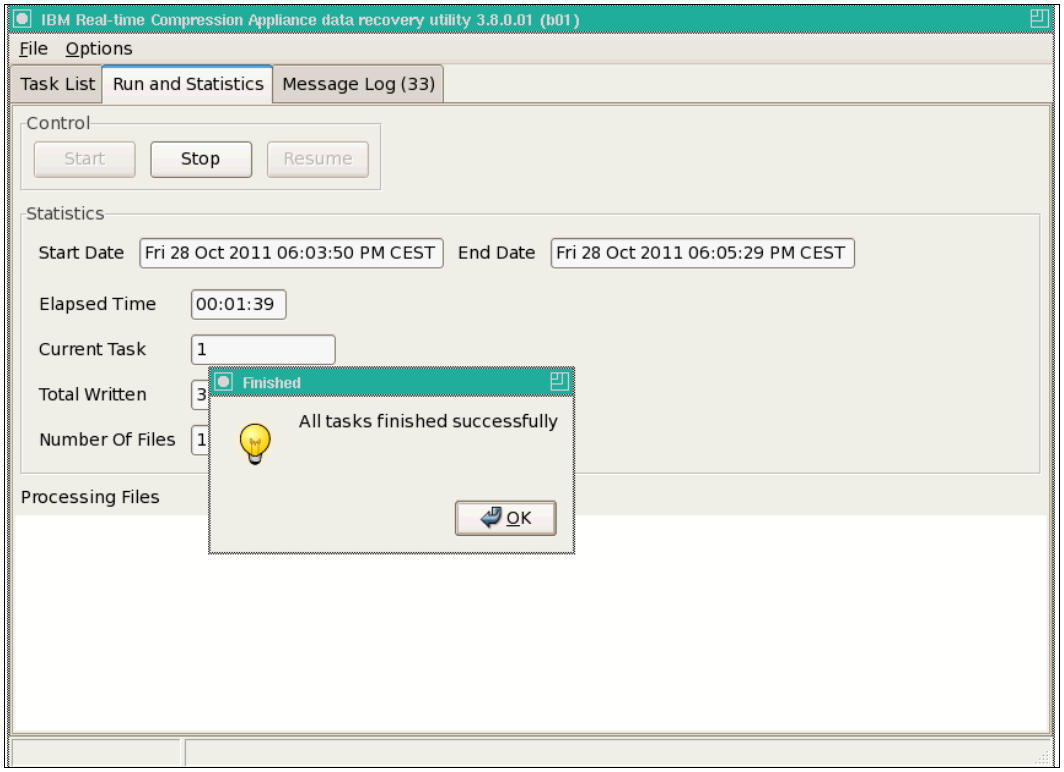


Figure 8-14 Finished task

7. After the recovery task is finished, you must make sure that the uncompressed datafile is mounted to the original location. After this, the Oracle database can be started.

8.3.2 Removing export/share from compression list

If we just want to change the share/export path from compression mode to transparent mode, we must decompress the file first before the datafile is accessible to Oracle database. This way, we can utilize RTCA without using recovery utility software. This technique is quite useful when dealing with an operating system other than Linux or Windows because the recovery utility is still only available on Linux or Windows platform.

Generally, we take the following steps to change the mode to transparent mode for the SOE datafile:

1. Take the SOE tablespace offline. This step must be done on a maintenance window, as shown in Figure 8-14. The command to use is shown in Example 8-14.

Example 8-14 Taking the tablespace offline

```
SQL> alter tablespace soe offline normal
  2  ;
Tablespace altered.
```

2. There are several options to decompress the files. Using the recovery utility, copy/move files from a compressed mode path to a transparent mode path. In this example, we decompress the file using the **move** command to a temporary path that is in transparent mode.
3. Check the current location of the compressed datafile (see Example 8-15).

Example 8-15 Current location of Oracle datafiles

```
SQL> select name from v$datafile;

NAME
--------------------------------------------------------------------------------
-
/u01/app/oracle/oradata/orcl/system01.dbf
/u01/app/oracle/oradata/orcl/sysaux01.dbf
/u01/app/oracle/oradata/orcl/undotbs01.dbf
/u01/app/oracle/oradata/orcl/users01.dbf
/u01/app/oracle/oradata/orcl/example01.dbf
/export/datafile/sh/sh4/sh4.dbf
/export/datafile/sh/sh/sh.dbf
/export/datafile/sh/sh3/sh3.dbf
/export/datafile/soe/soe.dbf
/export/datafile/cc/ccdata.dbf
/export/datafile/sh/sh5/sh5.dbf

NAME
--------------------------------------------------------------------------------
-
/u01/app/oracle/oradata/orcl/users02.dbf

12 rows selected.

SQL>
```

4. Verify the size of the datafile on the operating system mode (Example 8-16).

Example 8-16 Checking size of datafile

```
[root@x3650-m2-84 soe]# ls -l /export/datafile/soe
total 3355656
-rw-r----- 1 oracle oinstall 3436191744 Oct 28 18:11 soe.dbf
```

5. Make the tablespace offline to preserve consistency and to make the file able to be copied (Example 8-17).

Example 8-17 Make tablespace offline

```
SQL> alter tablespace soe offline normal;

Tablespace altered.
```

6. Move files on the Linux host to a transparent mode path. This can be done in many ways, such as creating another export path on the N series using the least directory than on the compressed path. In this case, we use /vol/oracle/datafile/soe as the compressed path. Create another export path on N series /vol/oracle/. This way, when /vol/oracle is mounted on the Linux host, we can also access files on directory to datafile/soe as transparent path (see Example 8-18 on page 129).

Example 8-18   Move file to transparent path

```
[root@x3650-m2-84 soe]# pwd
/export/datafile/soe
[root@x3650-m2-84 soe]# ls -l
total 3355656
-rw-r----- 1 oracle oinstall 3436191744 Oct 28 18:20 soe.dbf
[root@x3650-m2-84 soe]# mv /export/datafile/soe/soe.dbf
/mnt/oracle/datafile/soe/soe_nc.dbf                             compressed path
                                       transparant path

[root@x3650-m2-84 soe]# ls -l
total 3362252
-rw-r----- 1 oracle oinstall 3436191744 Oct 28 18:20 soe_nc.dbf
[root@x3650-m2-84 soe]# mv soe_nc.dbf soe.dbf          change back to original
                                                        file name
```

Precaution: Use caution when accessing compressed data with another transparent exported path. Write access to files already compressed by the software if the bypass path is not supported and might corrupt the file's content. Access to compressed files in the bypass path is limited to read-only access.

7. Remove the compressed export path on the compression filters (see Figure 8-15).

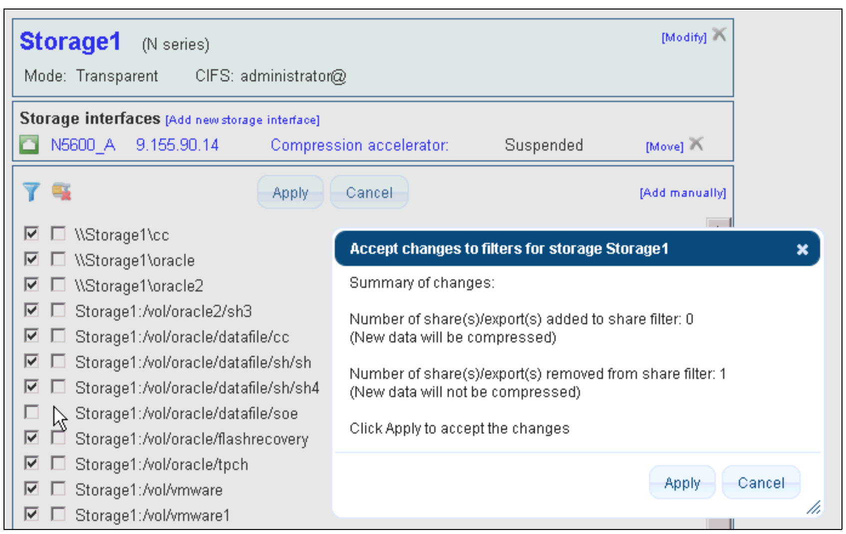


Figure 8-15   Remove export path from compression filters

Important: Verify that the tablespace is in the offline state prior to removing the export path on the compression filter. Removing the export path from the compression filter will result in an inconsistency in the datafile and might cause data corruption.

8. Example 8-19 shows how to bring the Oracle tablespace back online.

Example 8-19   Bring tablespace back online

```
SQL> alter tablespace soe online;

Tablespace altered.
```

9

File serving use cases

In this chapter, we demonstrate a practical approach for using the RTCA product in a file service environment.

For a typical Windows environment, we set up an example CIFS share, with compression by the RTCA product, and deduplication by the underlying NAS controller. We also discuss how to compress automatic CIFS home shares. For a typical UNIX environment, we set up an example export, also with compression by the RTCA product, and deduplication by the underlying NAS controller.

Important: This chapter uses IBM System Storage N series as the NAS system. The various solution and design approaches can be easily applied to other NAS solutions available in the marketplace.

9.1 Setting up a CIFS file service

This section describes several scenarios for using the RTCA product to compress an existing CIFS file service.

Assumptions:

- This section assumes that the initial configuration of the RTCA product and the NAS storage controller have both been completed.
- We assume that the RTCA management port (IP addresses), network (bonds and bridges), and storage (NAS controllers) have already been configured. We also assume that the NAS controller's CIFS protocol has already been configured.
- At this stage, all CIFS (and NFS) traffic is already passing through the RTCA product in transparent mode. However, no data is being compressed or uncompressed.

The CIFS function of an IBM N series storage controller was used for all NAS examples.

9.1.1 Configuring the NAS storage controller

Prior to applying the RTCA product to the CIFS file service, it is necessary to configure some settings, and to create the volumes and CIFS shares.

Selecting the CIFS protocol version

Although the RTCA product introduced partial support for SMB 2.0 in version 3.8, it is still necessary to disable SMBv2 on the NAS storage controller.

Enter the command shown in Example 9-1 to set the CIFS protocol mode on the NAS controller.

Example 9-1 Disabling SMB 2.0 protocol

```
NAS> option cifs.smb2.enable off
```

Any client sessions that use SMB 2.0 to connect by the RTCA product will be stepped down to SMBv1 for the back-end communication to the NAS controller.

Support: Future versions of the RTCA product might include full support for the SMB2 protocol. See the release notes for those versions for advice on how to configure the NAS controller if that is the case.

Disabling SMB signing

The SMB Signing feature is designed to prevent the man-in-the middle attack. Because the compression appliance is an in-line product, the SMB Signing mechanism perceives the appliance as a man-in-the-middle, intercepting traffic between the client and the NAS controller. Therefore the SMB Signing feature needs to be disabled on the NAS controller when using the RTCA product.

Enter the command shown in Example 9-2 to disable SMB signing on the NAS controller.

Example 9-2 Disable SMB signing

```
NAS> option cifs.signing.enable off
```

All other CIFS settings on the NAS controller, such as visibility of the ~snapshot directory, were left as default.

Example CIFS share

For our lab, we created the following CIFS shares on the NAS storage controller.

Although our lab examples rely on the CIFS function of the NAS controller, the focus of this book is not the NAS configuration per se. We assumed that the reader is already a competent NAS administrator, so that material is not described here.

For more information about NAS features and administration, see the *IBM System Storage N series Software Guide*, SG24-7129. Our sample CIFS share is shown in Example 9-3.

Example 9-3 Sample CIFS share

```
NAS> cifs shares cifs_vol1

Name         Mount Point                       Description
----         -----------                       -----------
cifs_vol1    /vol/cifs_vol1
                 everyone / No Access
                 N5600-B\cifsuser1 / Full Control
```

For the purposes of our lab examples, we populated the sample CIFS share with a variety of file types, including the Canterbury corpus, which was obtained from the following website:

http://corpus.canterbury.ac.nz/

Existing data

At the start of our lab, the CIFS share contained some data, as shown in Example 9-4.

Example 9-4 Sample CIFS share at start of the lab

```
NAS> df -Sh /vol/cifs_vol1
Filesystem                used       compressed          a-sis      %saved
/vol/cifs_vol1/           28MB              0KB          608KB          2%
```

You will notice that the sample share contained a small amount of NAS deduplicated data (shown in the “a-sis” column in Example 9-4). This was done to demonstrate the effect that the RTCA product has on existing NAS deduplicated data.

Existing snapshot copies

There were also a number of existing snapshot copies, as shown in Example 9-5.

Example 9-5 Existing snapshot copies at start of the lab

```
NAS> snap list

Volume cifs_vol1
working...

  %/used       %/total  date          name
```

```
---------- ---------- ------------ --------
  3% ( 3%)    0% ( 0%)  Oct 18 12:00  hourly.0
  6% ( 3%)    0% ( 0%)  Oct 18 08:00  hourly.1
  7% ( 1%)    0% ( 0%)  Oct 18 00:00  nightly.0
  8% ( 1%)    0% ( 0%)  Oct 17 20:00  hourly.2
 10% ( 1%)    0% ( 0%)  Oct 17 16:00  hourly.3
 11% ( 2%)    0% ( 0%)  Oct 17 12:00  hourly.4
 12% ( 2%)    0% ( 0%)  Oct 17 08:00  hourly.5
 13% ( 1%)    0% ( 0%)  Oct 17 00:00  nightly.1
```

There had been little change to the sample volume, so the percentage changed was also very low.

Dummy workload

To test the effect of enabling RTCA compression with a running CIFS workload, we configured a Windows PC to continuously run an *IOmeter* benchmark on the sample share.

The IOmeter product was downloaded from the following website:

http://www.iometer.org

Note that this was not intended as a performance benchmark and was chosen as a simple way to provide a continuous CIFS workload during our implementation.

9.1.2 Configuring the RTCA product

This section assumes that the initial configuration of the RTCA product was completed.

We assume that the RTCA management port (IP addresses), network (bonds and bridges), and storage (NAS controllers) are already configured.

For a description of the initial set up procedure, see the companion book to *Introduction to IBM Real-time Compression Appliances*, SG24-7953. The IBM Redbook is located at:

http://www.redbooks.ibm.com/abstracts/sg247953.html?Open

Authentication and authorization

You must configure the RTCA with a valid NAS domain user name and password to allow it to list the available CIFS shares. This list of CIFS shares will later be used to configure the compression filters and Compression Accelerator function.

To configure the CIFS credentials on the RTCA product:

1. Log in to the RTCA web interface.
2. In the left-hand menu, under Configuration, select **Storage**.
3. In the Storage systems view, select your NAS controller:
   a. If your NAS controller does not appear in the list, select **[Add new storage]**, and follow the process.
4. In the upper-right corner, select **[Modify]** to configure the CIFS credentials.
   a. Set the storage parameters as follows:
      - Vendor: (set to match your NAS vendor)
      - Compression mode: Transparent
      - No New Compression: Off
      - Domain user: administrator

- Domain password: (the administrator's password)
- Domain name: (your domain)

Be aware that the CIFS credentials are only used to *enumerate the CIFS shares* on the NAS controller. They have no effect on the security negotiation between the CIFS clients and the NAS controller.

NAS administrative accounts

In most NAS storage controllers, any valid user can enumerate the available shares, so this does not need to be an administrative user account. However, if the NAS controller supports *Access Based Share Enumeration*, where the list of shares is filtered relative to the user's access rights, then it might be necessary to use an administrative account. See the NAS administrator for advice on this setting. The GUI for this process is shown in Figure 9-1.

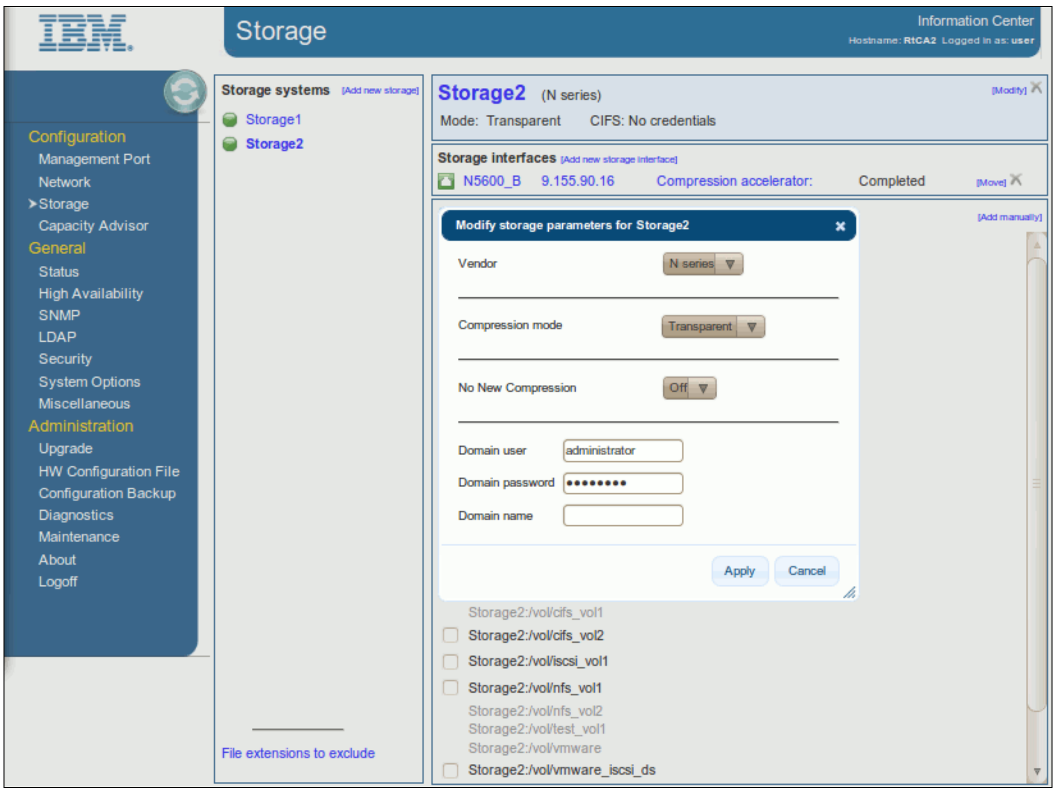


Figure 9-1 Setting the CIFS credentials

As shown in Example 9-6, you can list the active CIFS sessions on the NAS controller to confirm that the RTCA does not present its own user credentials.

Example 9-6 List the active CIFS sessions

```
NAS> cifs sessions

Server Registers as 'N5600-B' in workgroup 'REDBOOK'
Filer is using en_US for DOS users
Using Local Users authentication
====================================================
PC IP(PC Name) (user)#shares  #files

NAS>
```

We assumed that the Storage interfaces (IP address, and so on) for the NAS controller are already efined.

9.1.3 Storage compression mode

When a storage system is defined to the RTCA, it is assigned a compression mode. This controls whether the system leaves the new data unchanged (transparent mode) or compresses all new data by default (compressed mode). In our lab, we initially set the storage controllers to transparent mode.

Transparent mode

In this mode, no new user access is compressed or uncompressed, by default. All CIFS (and NFS) traffic is already passing through the RTCA product in transparent mode. To view the list of available CIFS (and NFS) shares on the NAS controller:

1. Log in to the RTCA web interface.
2. In the left-hand menu, under Configuration, select **Storage**.
3. In the Storage systems view, select your NAS controller. This will update the display to show the available CIFS shares. The GUI for this process is shown in Figure 9-2.

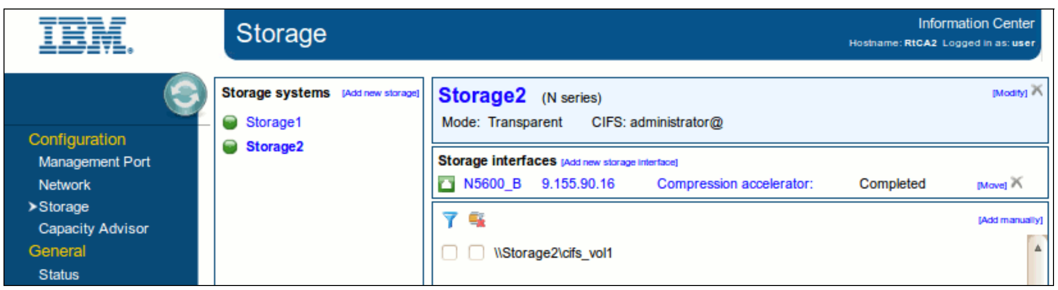


Figure 9-2 Viewing the available CIFS shares in transparent mode

For the example CIFS share in Figure 9-2, in storage transparent mode, the two check boxes have the following effect:

- Left box - Compression filter:
  - Unchecked = No compression or decompression
  - Checked = New data is compressed, existing data is uncompressed
- Right box - No new compression (only visible when left box is checked):
  - Unchecked = New data IS compressed, existing data is uncompressed
  - Checked = New data is NOT compressed, existing data is uncompressed

Compressed mode

In this mode, all new user access is compressed and uncompressed, by default. This can be altered with the “No new compression” attribute to selectively disable new compression, while retaining the ability to read existing compressed data. In storage compressed mode, the effect of the compression filter check box shown in Figure 9-2 is reversed.

- Compression filter:
  - Unchecked = New data is compressed, existing data is uncompressed
  - Checked = No compression or decompression

See 9.1.7, “Compressing automatic home shares” on page 143 for a description of using the storage controllers in compressed mode.

9.1.4 Compressing new data

This section describes how to enable compression for new data on a specific CIFS share.

Configuring a compression filter

In storage transparent mode, the compression filters identify on which CIFS shares to compress new data. No new data will be either compressed or uncompressed, unless the CIFS share is enabled with a compression filter.

Enabling a compression filter has no effect on any existing data. For existing data, see 9.1.5, “Compressing existing data” on page 138.

To configure the compression filter for a specific CIFS share:

1. Log in to the RTCA web interface.
2. In the left-hand menu, under Configuration, select **Storage**.
3. In the Storage systems view, select your NAS controller. This will update the display to show the available CIFS shares.
4. Select the check box beside the share that you want to compress (for new data):
   – This will cause a second check box to appear as shown in Figure 9-3.
   – Do not select this check box.

Figure 9-3 Enabling a compression filter for a CIFS share

5. When satisfied with your changes, click **Apply** at the top of share list. This will cause a confirmation dialog box to appear, as shown in Figure 9-4.

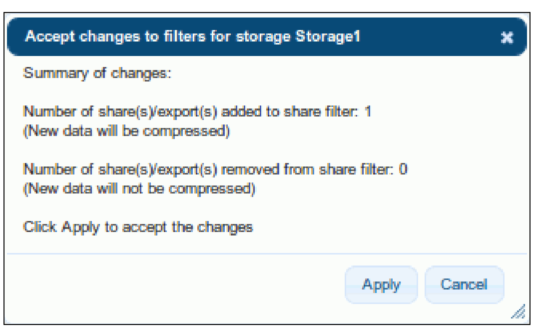


Figure 9-4 Confirm the new compression filter

The panel will then update to show the new compression filter.

At this point, any new (or updated) data written to this CIFS share will be compressed by the RTCA product.

Configuring file extensions to exclude

This section describes how to filter the compression engine to exclude specific file types. Because some data types are either already compressed (for example, ZIP files) or naturally uncompressible, they must be excluded from the RTCA compression process.

To configure the file type exclusions:

1. Log in to the RTCA web interface.
2. In the left-hand menu, under Configuration, select **Storage**.
3. In the Storage systems view, select your NAS controller.
4. In the lower-left corner, select **File extensions to exclude**. This will open a new dialog box, as shown in Figure 9-5.

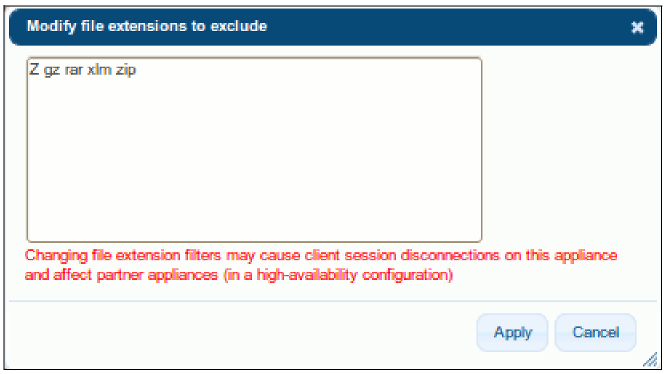


Figure 9-5 Configuring file extensions to exclude

5. Update the exclusion list as required. Click **Apply** to save your changes.

From this point, any new data that matches these file extensions will not compressed by the RTCA product.

9.1.5 Compressing existing data

This section describes how to use the Compression Accelerator to compress the existing data on a CIFS share or group of shares.

Setting a CIFS share to compressed mode (see 9.1.4, "Compressing new data" on page 137) will only compress the new or changes data on the share. Any existing data will remain uncompressed until it is updated by the client.

The RTCA product provides a "Compression Accelerator" feature to compress the existing data independent of any NAS client access.

Compression accelerator

This feature will read, compress, and rewrite the existing data on the nominated CIFS shares. This accelerates the space savings that are delivered by the data compression. Any subsequent client access will then be to the compressed form of the data.

An important difference between the compressed share setting and the Compression Accelerator, is that the compression setting is ongoing, while the CA process is typically only run once to process the existing data.

To configure the Compression Accelerator for the sample CIFS share:

1. Log in to the RTCA web interface.
2. In the left-hand menu, under Configuration, select **Storage**.
3. In the Storage systems view, select your NAS controller:
   a. Ensure that the icon beside your storage controller is **green** (as shown in Figure 9-6):
      - This indicates that this is the active RTCA node for that storage interface.
      - If not, then you must check the other RTCA node, or investigate why the NAS storage interface is offline.

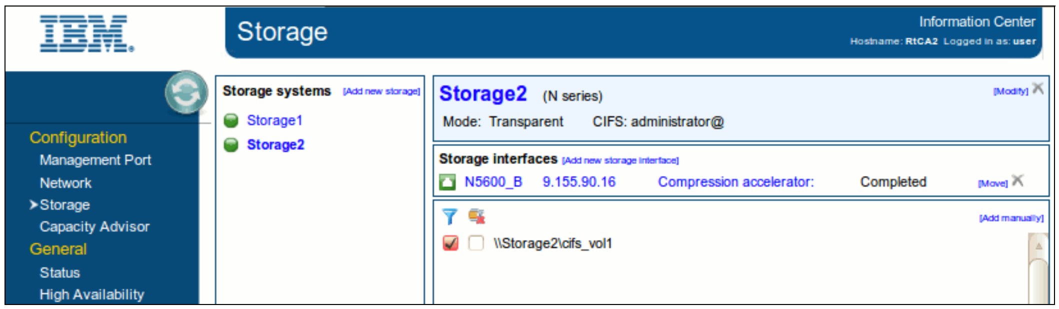


Figure 9-6 Enable the Compression Accelerator for a CIFS share

4. In the Storage Interfaces view, select **Compression Accelerator**. This will open a new panel, as shown in Figure 9-7 on page 140.
5. In the Settings tab, enter the following information:
   – Scheduling = (set as required, for example: 17:00 - 20:00)
   – CIFS username = (set to a user with administrative privileges)
   – CIFS password = (set to the correct password)
   – CIFS domain = (set to your domain)
   – Policy = (if desired, restrict the CA process by file size, timestamp, and so on)
   – Directories to ignore = (typically used to prevent the CA from processing snapshot copies)
   – Maximum total throughput (MB/sec) = (if desired, throttle the CA process to a specified throughput)
   – Amount of changes = (if desired, stop the CA process after the specified amount of change within the time frame)
   – Free space threshold = (if desired, stop the CA process if the free space on the share reaches this limit)
   – Interval between files (ms) = (if desired, throttle the CA process by imposing a delay between file processing)

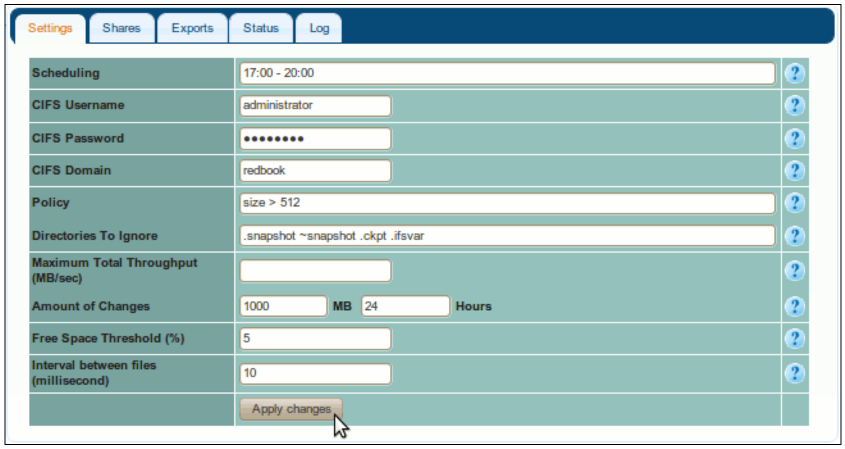


Figure 9-7 Configure the Compression Accelerator

6. In the Shares tab (shown in Figure 9-8), select the CIFS share to compress.

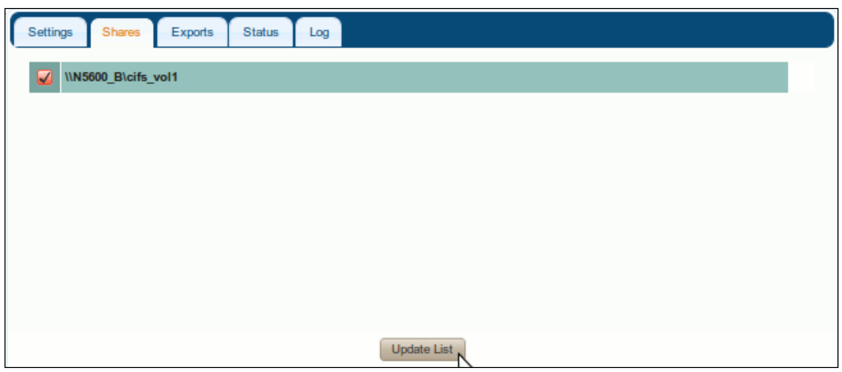


Figure 9-8 Select a CIFS share for the Compression Accelerator

7. Skip the Exports tab.
8. In the Status tab (shown in Figure 9-9), review the current status.
9. Click **Start** to initiate the CA process. This will update the display to show the files being compressed.

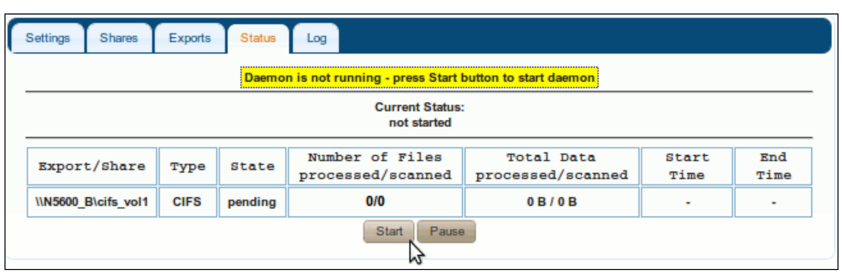


Figure 9-9 Start the Compression Accelerator

10. In the Log tab (shown in Figure 9-10), review the summary of the CA process.

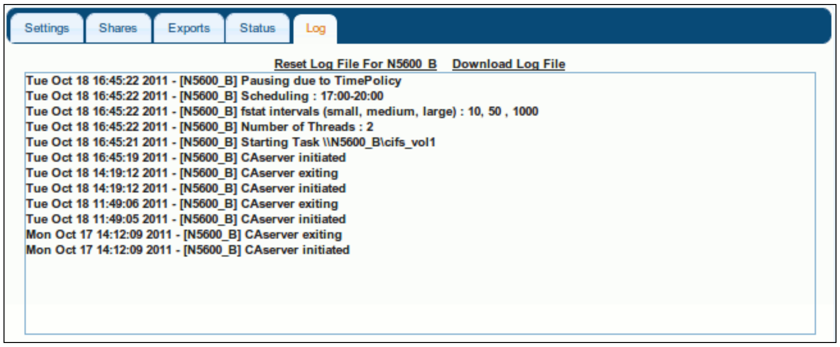


Figure 9-10 Review the log file for the Compression Accelerator

After the CA process has finished successfully, all of the existing data on the CIFS share will be in a compressed format.

Existing data in compressed mode

At the end of our RTCA lab, all of the data on the CIFS share had been compressed.

The reduction in used capacity is shown in Example 9-7. The compression ratio is representative of our particular data set. You need to test the RTCA product in your own environment to determine the estimated capacity saving with your data set.

Example 9-7 Example capacity after compression

```
N5600-B> df -Sh /vol/cifs_vol1

Filesystem                used      compressed       a-sis     %saved
/vol/cifs_vol1/           10MB             0MB         0MB         0%
```

After the RTCA compression, and Compression Accelerator, had completed there was a large change in the active file system versus the snapshot backup. In our example, this was still below the 20% snapshot reserve, so it had no impact on the available file system capacity.

> **Important:** In a production environment, you must consider the impact of RTCA compression on the snapshot reserve and usable volume capacity. It might be prudent to delete some or all snapshots prior to running the Compression Accelerator.

You might notice that the data that was deduplicated by the NAS controller prior to the RTCA process (shown in the “a-sis” column, see “Existing data” on page 133) is now un-deduplicated. This is because the data was rewritten by the Compression Accelerator, which invalidated the block sharing at the file system layer.

9.1.6 Enabling NAS deduplication

One of the features of the RTCA product is that the compressed data is written in a manner that is compatible with subsequent deduplication on the NAS controller. This allows us to exploit the storage efficiency benefits of both RTCA compression and NAS deduplication.

Although the aim of this book is not to describe the features of the NAS controller, the deduplication process is described here because it provides further benefits to the function of the RTCA product.

> **Important:** Schedule the RTCA compression and NAS deduplication in the correct order. Data that has been compressed by the RTCA can be deduplicated. However, previously deduplicated data will be rewritten by the RTCA in compressed form. This effectively un-deduplicates the data (which can then be re-deduplicated if desired).

Example 9-8 shows how to enable NAS deduplication on our sample volume.

Example 9-8 Enabling deduplication on the NAS controller

```
NAS> sis on /vol/cifs_vol1
SIS for "/vol/cifs_vol1" is enabled.
Already existing data could be processed by running "sis start -s /vol/cifs_vol1".

NAS> sis start -s /vol/cifs_vol1
The file system will be scanned to process existing data in /vol/cifs_vol1.
This operation may initialize related existing metafiles.
Are you sure you want to proceed (y/n)? y
The SIS operation for "/vol/cifs_vol1" is started.

[wait for the deduplication process to finish...]

NAS> df -Sh /vol/cifs_vol1
Filesystem                used       compressed         a-sis     %saved
/vol/cifs_vol1/         3940KB              0KB        7276KB        65%
```

For ongoing operations, any new or changed data will be compressed by the RTCA product. The NAS deduplication process will need to be scheduled to process the compressed data on disk (as is normal for the post-process deduplication feature).

Of course, on the CIFS client, the compressed data is still readable in its original form, and the file system capacity is reported correctly, as shown in Example 9-9.

Example 9-9 Viewing the logical data capacity on the CIFS client

```
N:\>dir /s
 Volume in drive N is cifs_vol1
 Volume Serial Number is 0101-482A

 Directory of N:\

[directoy listing not shown]

     Total Files Listed:
             125 File(s)     29,526,465 bytes
              36 Dir(s)   8,585,900,032 bytes free
```

However, the true used capacity can been seen on the client by opening Explorer, right-clicking the CIFS drive letter, and selecting **Properties**, as shown in Figure 9-11.

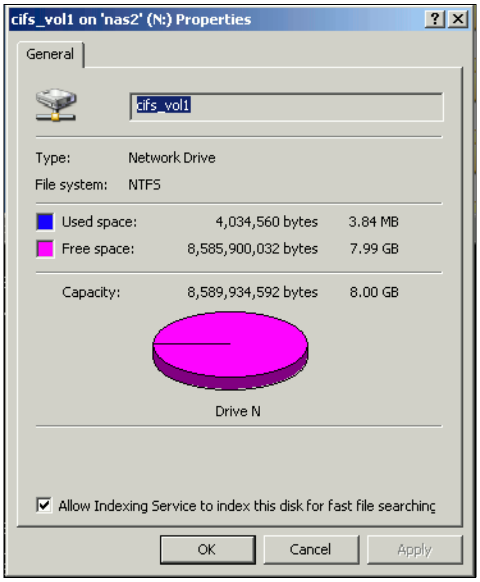


Figure 9-11 Viewing the physical data capacity on the CIFS client

The slight variation in reported capacity is due to the different ways that the NAS controller and Windows report disk usage.

9.1.7 Compressing automatic home shares

This section describes how to correctly configure the RTCA to work with auto home shares.

The IBM N series storage controller has the ability to dynamically create a CIFS share on demand when a user access their home directory. When the user disconnects, the share is automatically removed.

This reduces the load on the NAS controller, because it now only needs to maintain those CIFS home shares that are actively in use at any particular time. It also reduces the administrative burden of needing to define individual user home shares, which might be many thousands of users in a large environment. Another interesting feature of automatic home shares is that they are only visible to the individual account owner, even the NAS administrator cannot list this type of share.

However, this presents a challenge with the RTCA. This is because if the auto home shares cannot be listed by an administrative user, then they cannot be shown in the compression filter GUI. Nor can they be processed by the Compression Accelerator. This is a form of *access based share enumeration*.

Configuring the NAS controller

This section describes how to configure the auto home share feature on the IBM N series storage controller.

Configuration:

- This book does not normally seek to describe how to configure the NAS controller, and is mainly concerned with integration with the RTCA product.
- The process to configure the auto home shares on the IBM N series controller is described here because it needs to be understood for a successful RTCA deployment.

Step 1: Create a parent directory for the home shares

Follow these steps to configure the parent directory and CIFS share:

1. Log in to the NAS administrative interface (CLI examples shown here).
2. Create a new volume to contain the user home shares, as shown in Example 9-10:
   a. Set the size and other parameters to suit your requirements.
   b. The CIFS home shares will be created as sub-directories under this volume.

Example 9-10 Create a NAS volume to contain the home shares

```
NAS> vol create cifs_home -s none aggr0 10g
Creation of volume 'cifs_home' with size 10g on containing aggregate
'aggr0' has completed.
```

3. Configure the NAS file system security mode for CIFS access, as shown in Example 9-11.

   This might already be correct, depending on the NAS default security style.

Example 9-11 Configure the NAS security style for the volume

```
NAS> qtree status cifs_home
Volume   Tree     Style Oplocks  Status
-------- -------- ----- -------- ---------
cifs_home          unix  enabled  normal

NAS> qtree security /vol/cifs_home ntfs
Thu Oct 13 15:09:53 GMT [N5600-B: wafl.quota.sec.change:notice]: security style
for /vol/cifs_home/ changed from unix to ntfs
```

4. Create a CIFS share for the containing volume, as shown in Example 9-12:
   a. Remove access permissions for everyone.
   b. Create access permissions for administrator.

Example 9-12 Create a CIFS share and set the access permissions

```
NAS> cifs shares -add cifs_home /vol/cifs_home
The share name 'cifs_home' will not be accessible by some MS-DOS workstations

NAS> cifs shares cifs_home
Name         Mount Point                          Description
----         -----------                          -----------
cifs_home    /vol/cifs_home
        everyone / Full Control

NAS> cifs access -delete cifs_home everyone
```

```
1 share(s) have been successfully modified

NAS> cifs access cifs_home administrator "Full Control"
1 share(s) have been successfully modified

NAS> cifs shares cifs_home
Name         Mount Point                       Description
----         -----------                       -----------
cifs_home    /vol/cifs_home
        N5600-B\administrator / Full Control
```

The CIFS share is now ready for administrative access.

> **Tip:** This CIFS share will not be accessed directly by the users. Instead it will be used by the RTCA product to access the contents of the auto home shares that are in sub-directories of the containing volume.

Step 2: Set the parent directory location

Follow these steps to configure the parent directory location:

1. Check whether a CIFS home directory has already been configured, as shown in Example 9-13. If this parameter is already configured you must investigate before continuing.

Example 9-13 Check the CIFS home directory location

```
NAS> cifs homedir
No CIFS home directory paths.

NAS> rdfile /etc/cifs_homedir.cfg
#
# This file contains the path(s) used by the filer to determine if a
# CIFS user has a home directory. See the System Administrator's Guide
# for a full description of this file and a full description of the
# CIFS homedir feature.
#
# There is a limit to the number of paths that may be specified.
# Currently that limit is 1000.
# Paths must be entered one per line.
#
# After editing this file, use the console command "cifs homedir load"
# to make the filer process the entries in this file.
#
# Note that the "#" character is valid in a CIFS directory name.
# Therefore the "#" character is only treated as a comment in this
# file if it is in the first column.
#
# Two example path entries are given below.
# /vol/vol0/users1
# /vol/vol1/users2
#
```

2. Configure the CIFS home directory parameter, as shown in Example 9-14 on page 146.

   Be careful with the **wrfile** command because it is easy to make a mistake and overwrite the configuration file with blank data.

Example 9-14 Set the CIFS home directory location

```
NAS> wrfile -a /etc/cifs_homedir.cfg /vol/cifs_home

NAs> cifs homedir load

NAS> cifs homedir
/vol/cifs_home
```

The CIFS service now knows where to look for the user's home directories when automatically creating the user's home shares.

Step 3: Set the auto home directory name style

Follow these steps to configure the naming standard for the auto home shares:

1. Configure the naming style for the auto home share, as shown in Example 9-15

   In our lab, we chose the default naming style. You must set this to match the requirements of your environment.

Example 9-15 Set the CIFS home directory name style

```
NAS> options cifs.home_dir_namestyle ntname
```

See the NAS product manual for a description of the other directory naming styles.

Step 4: Create the user's home directories

Follow these steps to create the user's home directory on the NAS controller:

1. Log in to the NAS client (for example, a Windows PC).
2. Create the home directory on the NAS controller, as shown in Example 9-16:
   a. Detach any existing CIFS connections.
   b. Authenticate to the NAS controller as the "administrator" user account.
   c. Create the CIFS user's home directory (under the homedir location that we configured in "Step 2: Set the parent directory location" on page 145).

Example 9-16 Make a home directory for the CIFS user/s

```
C:\>net use * /d
You have these remote connections:

                    \\nas2\IPC$
Continuing will cancel the connections.

Do you want to continue this operation? (Y/N) [N]: y
The command completed successfully.

C:\>net use \\nas2 /user:administrator
The password or user name is invalid for \\nas2.

Enter the password for 'administrator' to connect to 'nas2':********
The command completed successfully.

C:\>mkdir \\nas2\cifs_home\cifsuser1
```

At this point the CIFS auto home share is ready to access from the NAS client.

Step 5: Access the auto home share

Follow these steps to connect the user to their new home directory on the NAS controller:

1. Log in to the NAS client (for example, a Windows PC).
2. Connect to the new auto home share, as shown in Example 9-17.:
   a. Detach any existing CIFS connections.
   b. Make a connection to the new CIFS share.
   c. Authenticate to the NAS controller as the “cifsuser1” user account.
   (Of course, you must change this to suit your user account name.)

Example 9-17 Connect to the new CIFS auto home share

```
C:\>net use * /d
You have these remote connections:

                    \\nas2\IPC$
Continuing will cancel the connections.

Do you want to continue this operation? (Y/N) [N]: y
The command completed successfully.

C:\>net use h: \\nas2\cifsuser1 /user:cifsuser1
The password or user name is invalid for \\nas2\cifsuser1.

Enter the password for 'cifsuser1' to connect to 'nas2':
The command completed successfully.
```

Assuming that you configured the share permissions and security style correctly, the user can now read and write to their new CIFS home share.

When the user disconnects their CIFS session, the share will automatically be removed. Of course, the underlying directory and its contents remain on the NAS controller. They will be ready to be accessed on demand the next time the user connects to the system.

Configuring the RTCA product

At this point, with the NAS controller configured to the RTCA in transparent mode, the CIFS auto home shares will function as designed, being created on demand and removed when idle. However, in transparent mode, their contents will be neither compressed or uncompressed by the RTCA product.

As shown in Figure 9-12, the new CIFS auto home share does not appear in the compression filter GUI on the RTCA product. This is because the auto home shares are only visible to their respective users, and only while those users have active CIFS sessions.

Figure 9-12 Compression filters cannot see the CIFS auto home shares

For the same reason, it is also not possible to configure the Compression Accelerator to process the auto home shares. The new share does not appear in the GUI for the Compression Accelerator, as shown in Figure 9-13. The share that we created for the volume that contains the home shares is visible.

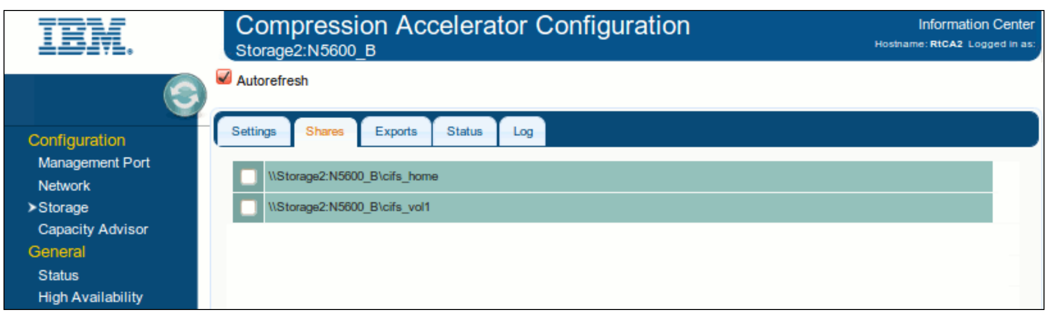


Figure 9-13 Compression accelerator cannot see the CIFS auto home shares

To compress the auto home shares, we must configure the RTCA product, as shown in the following steps.

Task 1: Changing the storage mode to compressed

When RTCA's storage mode for a specific NAS controller is changed to compressed, then all NAS access (CIFS shares and NFS exports) will be compressed by default. This is necessary because although we cannot specifically compress the auto home shares, they are included in the new default behavior. See 9.1.3, "Storage compression mode" on page 136 for a description of this feature.

> **Important:** The switch from transparent mode to compressed mode is a *one-way process*. It is not possible to change back to transparent mode.
>
> Be sure that you understand the impact of changing this mode before continuing.

To change the NAS controller to storage compressed mode:

1. Log in to the RTCA web interface.
2. In the left-hand menu, under Configuration, select **Storage**.
3. In the Storage systems view, select your NAS controller.

   This will update the display to show the current compression mode.
4. In the top-right corner of the storage display, as shown in Figure 9-14, select **[Modify]**.

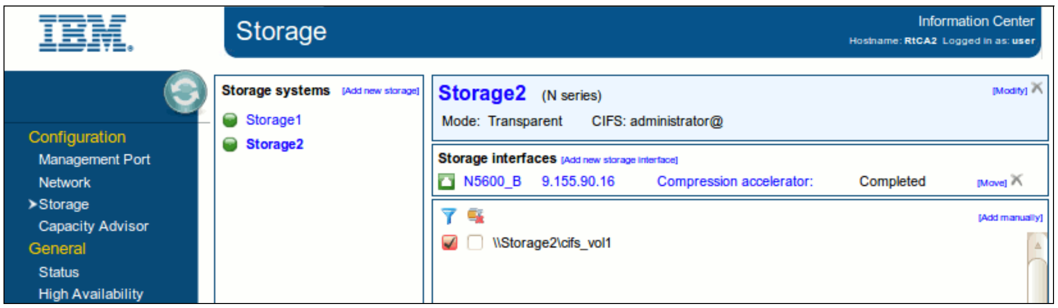


Figure 9-14 Configure the storage compression mode

5. The Modify storage parameters dialog box will appear, as shown in Figure 9-15.
   a. Select **Compressed** in the Compression mode field.
   b. Click **Apply**.

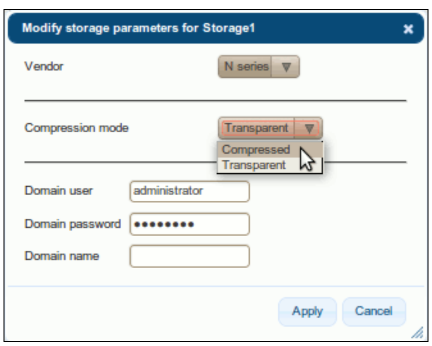


Figure 9-15 Select the compression mode

6. The dialog box will update with a warning message, as shown in Figure 9-16 on page 150:
   a. Be sure to read the warning message before continuing with this process.

      Also note that the No new compression field appears (NNC), and cannot be changed from On. We will modify this field later to enable compression.
   b. Click **Apply**.

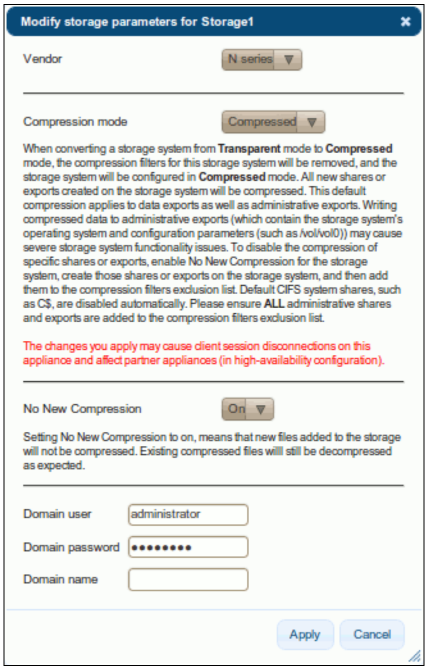


Figure 9-16 The compressed mode warning message

7. The display will then return to the main storage view, as shown in Figure 9-17.

   The list of CIFS shares (and NFS exports) has now changed. Because all access now defaults to compressed mode, the meaning of the check boxes is reversed to now indicate which shares *NOT* to compress.

   All access to the NAS controller will now default to compressed mode. However, although any existing data will be uncompressed as required, no new data will be compressed until the No new compression (NNC) setting is changed.

   To change the NNC setting, select **[Modify]** in the top-right corner of the storage display, as shown in Figure 9-17.

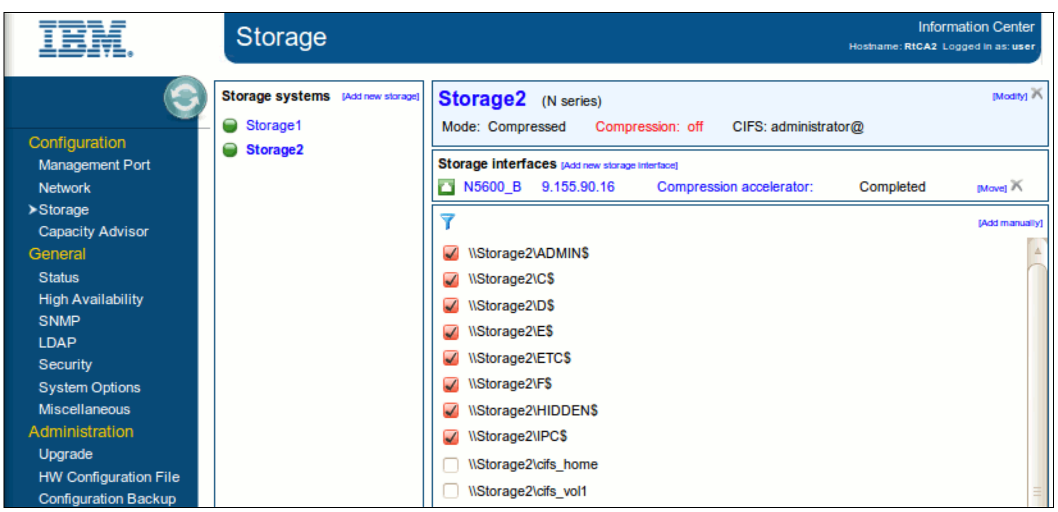


Figure 9-17 The storage compressed mode interface (compression disabled)

8. The Modify storage parameters dialog will open, as shown in Figure 9-18.
   a. Change the NNC setting to **Off**.
   b. Click **Apply**.

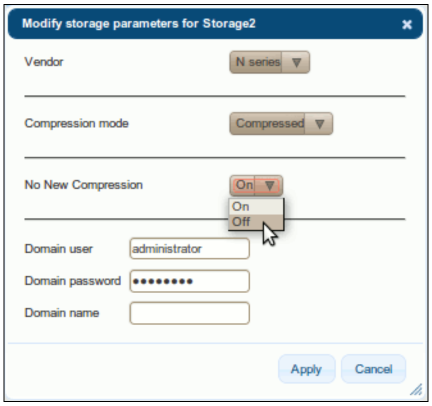


Figure 9-18 Change the NNC setting in compressed mode

9. The display will then return to the main storage view, as shown in Figure 9-19. Review the Mode and Compression settings to check your work.

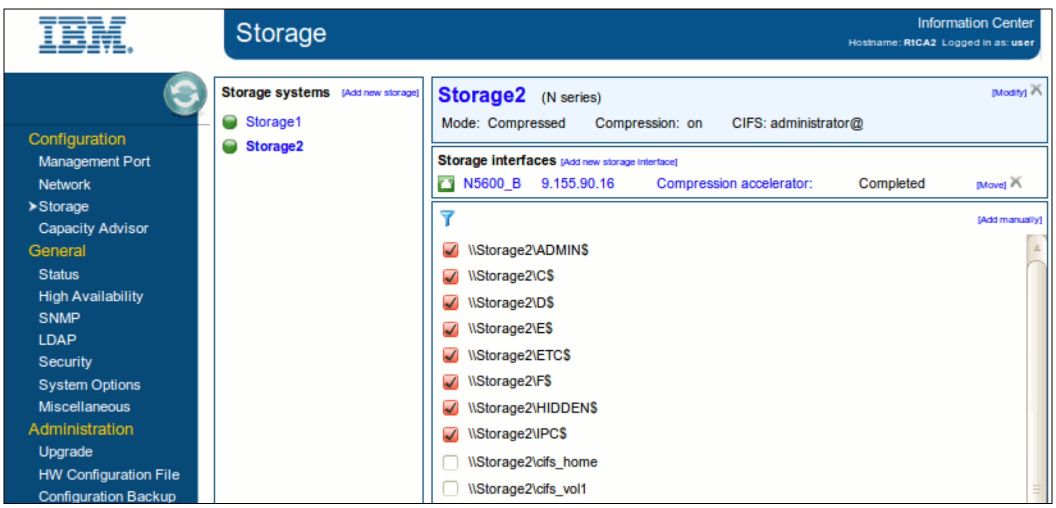


Figure 9-19 The storage compressed mode interface (compression enabled)

With the RTCA product now configured in compressed mode, all CIFS (and NFS) shares on the NAS controller will be compressed be default. This includes the CIFS auto home shares.

Warning: The "root" volume on the NAS controller must *never* be compressed:

- This is so, because the NAS controller accesses this volume directly (not by the RTCA path) and then will not be able to read the operating system files correctly.
- Be sure to exclude the NAS root volume from any write access by the RTCA compressed mode.

However, the existing data on the NAS controller will not be compressed. For that we must configure the Compression Accelerator, as described in the next task.

Task 2: Configuring the Compression Accelerator

Because the CIFS auto home share itself is not visible in the Compression Accelerator GUI (see Figure 9-13 on page 148), we must configure the CA to process a point lower in the NAS file system.

Configuring the Compression Accelerator here is identical to the process described in 9.1.5, “Compressing existing data” on page 138:

1. Repeat the steps as described earlier, but in the Shares tab, select the volume share (that contains the auto home shares)

   In our lab example, this is the “cifs_home” share, as shown in Figure 9-20.

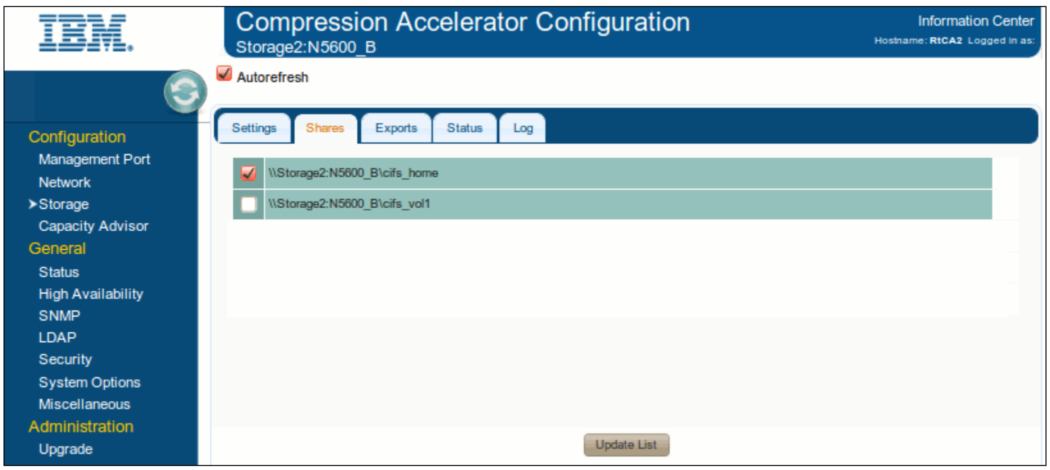


Figure 9-20 Select the CIFS share to process with the Compression Accelerator

The Compression Accelerator will then process the existing data in the volume that contains the CIFS auto home shares. Because the NAS controller is now set to compressed mode, the RTCA will compress any new writes from the client, and also uncompress data when accessed from the client.

9.1.8 Accessing the compressed data

This section describes the CIFS client’s view of the NAS data after it is compressed by the RTCA product.

Accessing data through the compressed path

When the RTCA is configured correctly, the CIFS client can access the compressed data exactly as expected. The file contents, and all file meta-data, are apparently unchanged.

Figure 9-21 shows a file that was accessed from a compressed (and deduplicated) CIFS share.

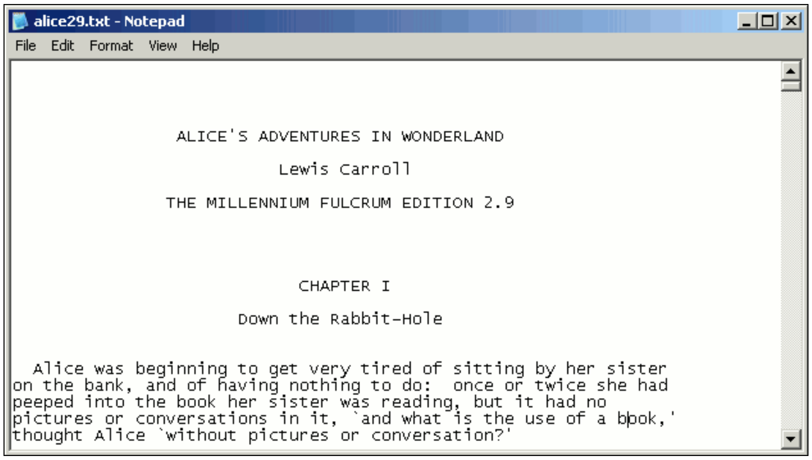


Figure 9-21 Correct access to the compressed data

> **Tip:** Any compressed data will still be accessed correctly if the NNC setting is enabled on an existing compressed share. The only effect is that new data will not be compressed.

Accessing data through the transparent path

If the RTCA is configured incorrectly, the CIFS client cannot access the compressed data. The file contents will appear corrupt to the CIFS client.

For the purpose of demonstration only, the volume containing the sample CIFS share was duplicated on the NAS controller by the NAS *volcopy* command. This volume contained the RTCA compressed data, but was configured for transparent access, bypassing the RTCA product's decompression algorithms.

The result of incorrectly accessing the compressed data is shown in Figure 9-22.

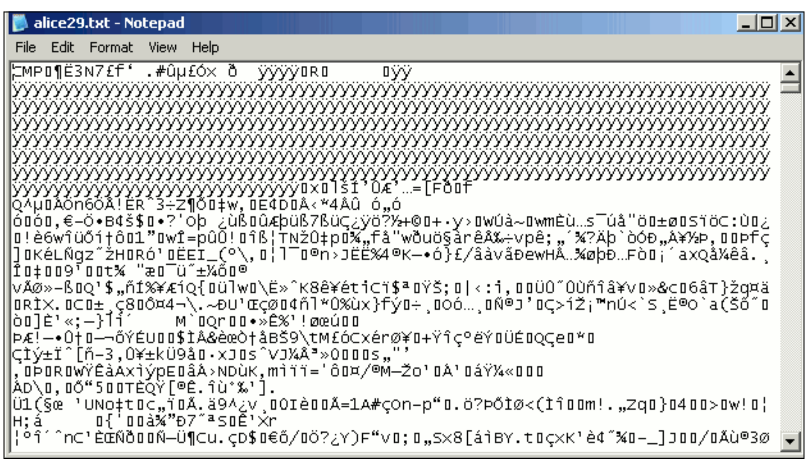


Figure 9-22 Incorrect access to the compressed data

Notice that Figure 9-22 shows the same file as in Figure 9-21, but it is now unusable. The first three characters of the file, `CMP` identify this as an RTCA compressed file.

During normal operations, access to the compressed data must always be through the RTCA compressed path. However, this is the way the files will appear if the RTCA is unavailable (for example, at a DR site with no local RTCA product).

If the RTCA is unavailable, the file can be manually uncompressed with the RTCA data recovery tool, as described in the companion IBM Redbook *Introduction to IBM Real-time Compression Appliances*, SG24-7953, located at:

http://www.redbooks.ibm.com/abstracts/sg247953.html?Open

Tip: Any files that had been excluded from compression (for example, ZIP files) will still appear as normal, because they were not processed by the RTCA product.

9.1.9 Calculating the compression ratio

Although it is not the intent of this book to provide guidance regarding RTCA performance or compression ratios, Table 9-1 shows sample compression ratios from our lab environment.

Table 9-1 Example compression ratios (with compression and deduplication)

	Original size	After RTCA	After NAS deduplication
CIFS_VOL1	28 MB	10 MB	3.9 MB
		Approx. 64% reduction	Approx 61% reduction
			Total 86% reduction

Obviously, the RTCA performance and compression ratios will vary depending on the NAS hardware platform and specific data set used. We encourage you to test the product in your environment before any production implementation.

9.2 Setting up an NFS file service

This section describes several scenarios for using the RTCA product to compress an existing NFS file service.

Assumptions:

- This section assumes that the initial configuration of the RTCA product and the NAS storage controller have both been completed.
- We assume that RTCA's management port (IP addresses), network (bonds and bridges), and storage (NAS controllers) are already configured. We also assume that the NAS controller's NFS protocol has already been configured.
- At this stage all NFS (and CIFS) traffic is already passing through the RTCA product in transparent mode. However, no data is being compressed or uncompressed.

The NFS function of an IBM N series storage controller was used for all NAS examples.

9.2.1 Configuring the NAS storage controller

Prior to applying the RTCA product to the NFS file service, it is necessary to configure some settings, and to create the volumes and NFS exports.

Selecting the NFS protocol version

Currently the RTCA product only supports version 3 of the NFS protocol (NFSv3).

Any clients using NFSv2 or NFSv4 will pass transparently through the appliance. Data written to with these versions of the NFS protocol is not compressed, and any already compressed data is read without being decompressed. See 9.2.7, “Accessing the compressed data” on page 170 to see how this will look on the client.

Enter the command shown in Example 9-18 to set the NFS protocol mode on the NAS controller.

Example 9-18 Disabling the NFSv4 protocol on the NAS controller

```
NAS> options nfs.v3.enable on
NAS> options nfs.v4.enable off
```

When NFSv3 is enabled, it is not possible to disable NFSv2. However, this is a legacy version of the protocol and has other limitations that make its use undesirable.

Tip: Although we disabled NFSv4 in our example, you can still use this protocol if necessary. However, all NFSv4 access through the RTCA will be in transparent mode.

Disabling Kerberos security

The NFSv3 protocol includes optional support for Kerberos (KRB) security, which is designed to prevent a man-in-the middle attack. Because the compression appliance is an in-line product, the KRB security perceives the appliance as a man-in-the-middle that intercepts traffic between the client and the NAS controller. Therefore the KRB security must be disabled on the NAS controller when using the RTCA product.

The NFSv4 protocol includes mandatory KRB security, which is one of the reasons why it is only supported in transparent mode. Enter the command shown in Example 9-19 to disable KRB security for NFSv3 on the NAS controller.

Example 9-19 Disable kerberos for NFS traffic

```
NAS> options nfs.kerberos.enable off
```

All other NFS settings on the NAS controller, such as visibility of the .snapshot directory, were left as default.

Security:

- The NFS security style is also configured in the /etc/exports file, and `exportfs` command.
- The export option `sec=` is only supported with the `sys` value. The options `krb5`, `krb5i` and `krb5p` are not supported.

Example NFS export

For our lab, we created the following NFS export on the NAS storage controller.

Although our lab examples rely on the NFS function of the NAS controller, the focus of this book in not NAS configuration per se. We have assumed that the reader is already a competent NAS administrator, and that material is not described here.

For more information about NAS features and administration, see the Redbooks publication, *IBM System Storage N series Software Guide*, SG24-7129.

Our sample NFS export is shown in Example 9-20. The name `myclient` was the hostname of our UNIX client used to mount the NFS export.

Example 9-20 Sample NFS export

```
NAS> exportfs
/vol/nfs_vol1    sec=sys,rw=myclient,root=myclient,nosuid
```

For the purposes of our lab examples, we populated the sample NFS export with a variety of file types, including data extracted into multiple files from the Freebase database, which was obtained from the following website:

http://download.freebase.com/datadumps/

Existing data

At the start of our lab, the NFS export contained some data, as shown in Example 9-21.

Example 9-21 Sample NFS export at start of the lab

```
N5600-B> df -Sh /vol/cifs_vol1
Filesystem                 used       compressed          a-sis     %saved
/vol/nfs_vol1/             11GB              0GB            0GB         0%
```

The sample export did not contain any NAS deduplicated or NAS compressed data.

Existing snapshot copies

There were also a number of existing snapshot copies, as shown in Example 9-22.

Example 9-22 Existing snapshot copies at start of the lab

```
N5600-B*> snap list

Volume nfs_vol1
working...

  %/used       %/total  date          name
----------  ----------  ------------  --------
 11% (11%)    2% ( 2%)  Oct 18 12:00  hourly.0
 20% (11%)    4% ( 2%)  Oct 18 08:00  hourly.1
 20% ( 0%)    4% ( 0%)  Oct 18 00:00  nightly.0
 20% ( 0%)    4% ( 0%)  Oct 17 20:00  hourly.2
 20% ( 0%)    4% ( 0%)  Oct 17 16:00  hourly.3
 27% (11%)    6% ( 2%)  Oct 17 12:00  hourly.4
 33% (11%)    8% ( 2%)  Oct 17 08:00  hourly.5
 33% ( 0%)    9% ( 0%)  Oct 17 00:00  nightly.1
```

There had been little change to the sample volume, so the percentage change was also very low.

Dummy workload

To test the effect of enabling RTCA compression with a running NFS workload, we configured a UNIX system to continuously run a Filebench benchmark on the sample export.

The Filebench product was downloaded from the following website:

http://sourceforge.net/apps/mediawiki/filebench/

This was not intended as a performance benchmark, and was chosen as a simple way to provide a continuous NFS workload during our implementation.

9.2.2 Configuring the RTCA product

This section assumes that the initial configuration of the RTCA product has been completed.

Assumptions

We assume that the RTCA's management port (IP addresses), network (bonds and bridges), and storage (NAS controllers) have already been configured.

Authentication and authorization

To enable the RTCA product to proxy the NFS client access you must add it to the NFS exports permissions on the NAS controller. This will allow the RTCA to list the available NFS exports, and will later be used by the compression filters and Compression Accelerator function.

Access:

- Here we configure the NAS controller to allow NFS access from the RTCA product.
- This is opposite the CIFS example, where we configured the RTCA with the correct user credentials to access the NAS controller.

To configure the host access parameters for the sample NFS export:

1. Log in to the NAS administrative interface (CLI examples shown here).
2. Add the IP addresses for the RTCA bridge interfaces to the **/etc/hosts** file, as shown in Example 9-23:
   a. Review the contents of the file.
   b. Append the bridge IP addresses and logical names to the file.
   c. Review the changes.

Example 9-23 Add the RTCA bridge addresses to the host file on the NAS controller

```
N5600-B> rdfile /etc/hosts

#Auto-generated by setup Tue Sep  6 12:47:40 GMT 2011
127.0.0.1     localhost
10.0.0.1      N5600-B N5600-B-e0a
10.0.0.100    myclient

N5600-B> wrfile -a /etc/hosts 10.0.0.10 rtca2-br0
N5600-B> wrfile -a /etc/hosts 10.0.0.11 rtca2-br1
N5600-B> wrfile -a /etc/hosts 10.0.0.12 rtca1-br0
N5600-B> wrfile -a /etc/hosts 10.0.0.13 rtca1-br1
```

```
N5600-B> rdfile /etc/hosts

#Auto-generated by setup Tue Sep  6 12:47:40 GMT 2011
127.0.0.1      localhost
10.0.0.1       N5600-B N5600-B-e0a
10.0.0.100     myclient
10.0.0.10      rtca2-br0
10.0.0.11      rtca2-br1
10.0.0.12      rtca1-br0
10.0.0.13      rtca1-br1
```

Obviously, you must replace the **10.0.0.X** addresses with the correct IP addresses for the bridges on your RTCA product.

> **Ping:** If you decide to check your configuration, use the **ping** command to test the bridge interfaces from the NAS controller. Be aware that you will NOT be able to ping the active bridge interface.

3. Add the logical names for the RTCA bridge interfaces to the **/etc/netgroups** file, as shown in Example 9-24:

   Later, this will simplify the configuration of the NFS host access parameters:

   a. Review the contents of the file.
   b. Append the bridge logical names to the file (in the format shown).
   c. Review the changes.

Example 9-24 Add RTCA bridge names to the netgroup file on the NAS

```
N5600-B> rdfile /etc/netgroup

/etc/netgroup: No such file or directory

N5600-B> wrfile -a /etc/netgroup rtca (rtca1-br0,,) (rtca1-br1,,) (rtca2-br0,,)
(rtca2-br1,,)

N5600-B> rdfile /etc/netgroup

rtca (rtca1-br0,,) (rtca1-br1,,) (rtca2-br0,,) (rtca2-br1,,)
```

4. Add the RTCA bridge interfaces to the export parameters, as shown in Example 9-25:

   a. Review the current export parameters.
   b. Update the export parameters. The **@rtca** field refers to the netgroup configured in Example 9-24.
   c. Review the changes.

Example 9-25 Add the RTCA netgroup entry to the sample NFS export parameters

```
N5600-B> exportfs -q /vol/nfs_vol1

/vol/nfs_vol1    -sec=sys,(ruleid=1),rw=myclient,root=myclient,nosuid

N5600-B>
exportfs -p sec=sys,rw=myclient:@rtca,root=myclient:@rtca,nosuid /vol/nfs_vol1
```

```
N5600-B> exportfs -q /vol/nfs_vol1
/vol/nfs_vol1 -sec=sys,(ruleid=2),rw=myclient:@rtca,root=myclient:@rtca,nosuid
```

Ignore the (ruleid=X) in the example, which indicates that the NAS controller's access cache rule that represents this particular set of host access parameters.

At this point, the RTCA product has sufficient authorization to access and update (compress) the contents of the sample NFS export.

We assume that the Storage interfaces (IP address, and so on) for the NAS controller have already been defined.

9.2.3 Storage compression mode

When a storage system is defined to the RTCA it is assigned a compression mode. This controls whether the system leaves the new data unchanged (transparent mode) or compresses all new data by default (compressed mode). In our lab, we initially set the storage controllers to transparent mode.

Transparent mode

This is the default mode when a new storage controller is defined to the RTCA. In this mode, no new user access is compressed, or uncompressed, by default; however, all CIFS (and NFS) traffic is already passing through the RTCA product in transparent mode. Follow these steps to view the list of available NFS (and CIFS) shares on the NAS controller:

1. Log in to the RTCA web interface.
2. In the left-hand menu, under Configuration, select **Storage**.
3. In the Storage systems view, select your NAS controller. This will update the display to show the available CIFS shares, as shown in Figure 9-23.

Figure 9-23 Viewing the available NFS exports in transparent mode

For the example NFS export in Figure 9-23, in storage transparent mode, the two check boxes have the following effect:

- Left box - Compression filter:
  - Unchecked = No compression or decompression
  - Checked = New data is compressed, existing data is uncompressed
- Right box - No new compression (only visible when left box is checked):
  - Unchecked = New data IS compressed, existing data is uncompressed
  - Checked = New data is NOT compressed, existing data is uncompressed

See 9.1.4, "Compressing new data" on page 137 for a description of how to configure the compression filter to compress new data.

Compressed mode

If the RTCA's storage mode for a specific NAS controller is change to compressed, all NAS access (CIFS shares and NFS exports) to that controller is compressed by default.

This can be fine-tuned with the No new compression attribute to selectively disable new compression, while retaining the ability to read existing compressed data. In storage compressed mode, the effect of the compression filter check box, shown in Figure 9-23 on page 159, is reversed.

- Compression filter:
  - Unchecked = New data is compressed, existing data is uncompressed
  - Checked = No compression or decompression

> **Important:** The switch from transparent mode to compressed mode is a *one-way process*. It is not possible to change back to transparent mode.
>
> Be sure that you understand the impact of changing this mode before continuing.

Follow these steps to change the NAS controller to storage compressed mode:

1. Log in to the RTCA web interface.
2. In the left-hand menu, under Configuration, select **Storage**.
3. In the Storage systems view, select your NAS controller to update the display to show the current compression mode.
4. In the top-right corner of the storage display, as shown in Figure 9-24, select **[Modify]**.

Figure 9-24 Configuring the storage compression mode

5. The Modify storage parameters dialog box will appear, as shown in Figure 9-25:
   a. Select **Compressed** in the Compression mode field. The CIFS user credentials are not required to process NFS exports.
   b. Click **Apply**.

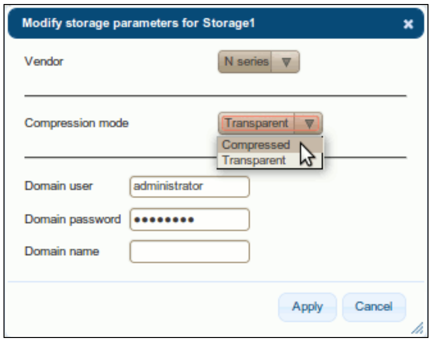


Figure 9-25 Select the compression mode

6. The dialog box will update with a warning message, as shown in Figure 9-26:
   a. Be sure to read the warning message before continuing with this process.

      Also note that the No new compression field appears (NNC), and cannot be changed from On. We will modify this field later to enable compression.

      Again, the CIFS user credentials are not required to process NFS exports.
   b. Click **Apply**.

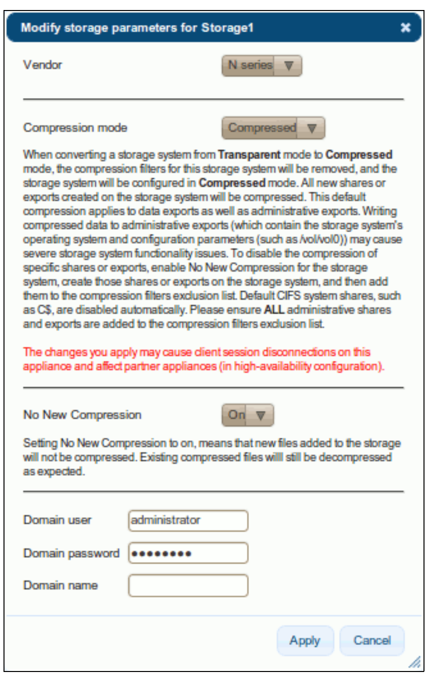


Figure 9-26 The compression mode warning message

7. The display will then return to the main storage view, as shown in Figure 9-27.

   The list of NFS export (and CIFS shares) has now changed. Because all access now defaults to compressed mode, the meaning of the check boxes is reversed to now indicate which exports to *NOT* compress.

   All access to the NAS controller will now default to compressed mode. However, although any existing data will be uncompressed as required, no new data will be compressed until the No new compression (NNC) setting is changed.

   To change the NNC setting, select **[Modify]** in the top right corner of the storage display, as shown in Figure 9-27.

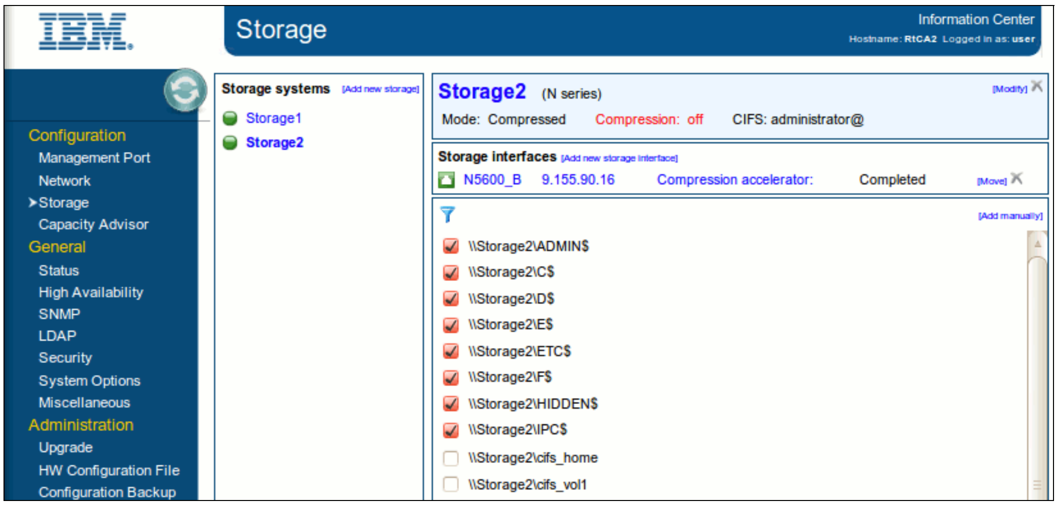


Figure 9-27 The storage compressed mode interface (compression disabled)

8. When the Modify storage parameters dialog opens, as shown in Figure 9-28:

   a. Change the NNC setting to **Off**. Again, the CIFS user credentials are not required to process NFS exports.

   b. Click **Apply**.

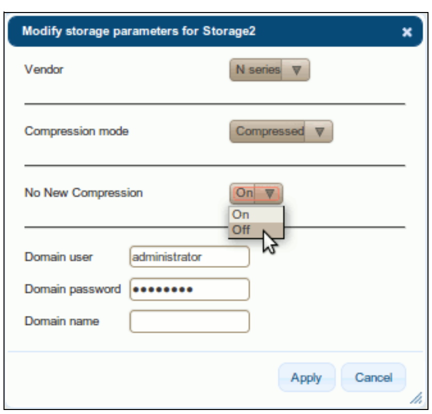


Figure 9-28 Change the NNC setting in compressed mode

9. The display will then return to the main storage view, as shown in Figure 9-29. Review the Mode and Compression settings to check your work.

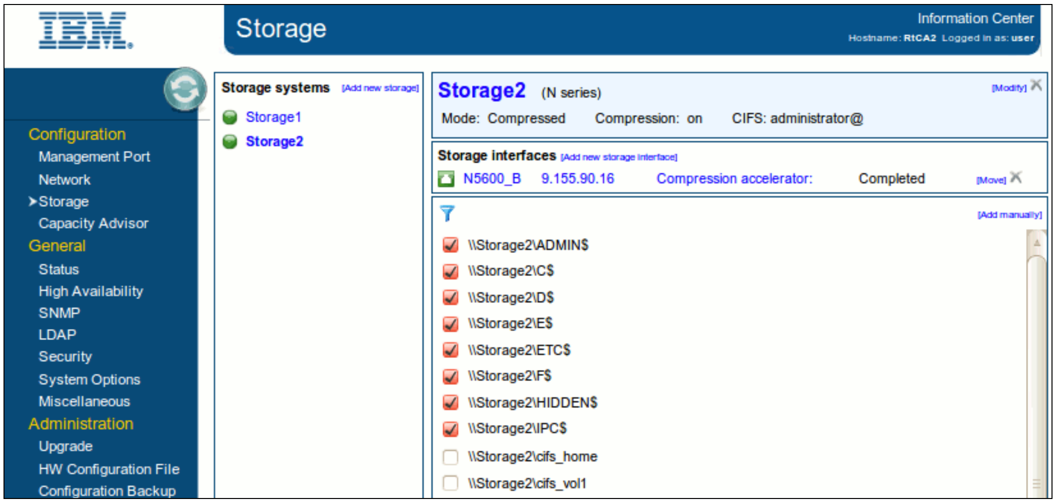


Figure 9-29 The storage compressed mode interface (compression enabled)

With the RTCA product now configured in compressed mode, all NFS exports (and CIFS shares) on the NAS controller will be compressed be default.

Warning: The root volume on the NAS controller must *never* be compressed:

- This is so, because the NAS controller accesses this volume directly (not by the RTCA path) and then will not be able to read the operating system files correctly.
- Be sure to exclude the NAS root volume from any write access by the RTCA compressed mode.

However, the existing data on the NAS controller will not be compressed. For that we must configure the Compression Accelerator, as described in 9.2.5, “Compressing existing data” on page 165.

9.2.4 Compressing new data

This section describes how to enable compression for new data on a specific NFS export.

Configuring a compression filter

In storage transparent mode, the compression filters are used to identify on which NFS exports to compress new data. No new data will be either compressed, or uncompressed, unless the NFS export is enabled with a compression filter.

Enabling a compression filter has no effect on any existing data. For existing data, see 9.2.5, “Compressing existing data” on page 165.

To configure the compression filter for a specific CIFS share:

1. Log in to the RTCA web interface.
2. In the left-hand menu, under Configuration, select **Storage**.

3. In the Storage systems view, select your NAS controller. This will update the display to show the available NFS exports, as shown in Figure 9-30.
4. Select the check box beside the export that you want to compress (for new data):
   - This will cause a second check box to appear (for the “No new compression” option).
   - Do not select this check box.

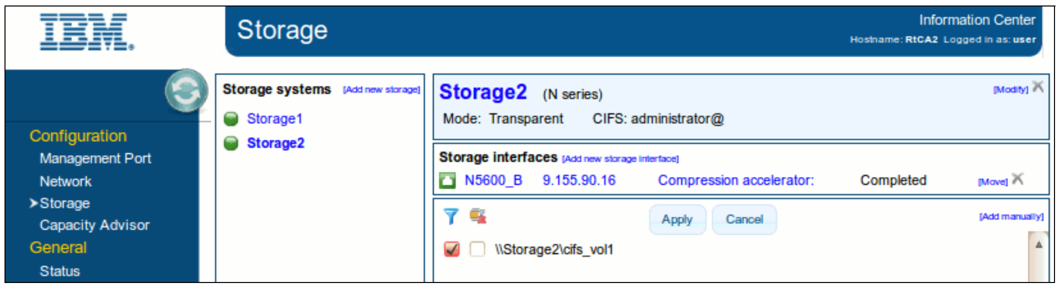


Figure 9-30 Enabling the compression filter for an NFS export

5. When satisfied with your changes, click **Apply** at the top of the share list. A confirmation dialog box is displayed, shown in Figure 9-31.

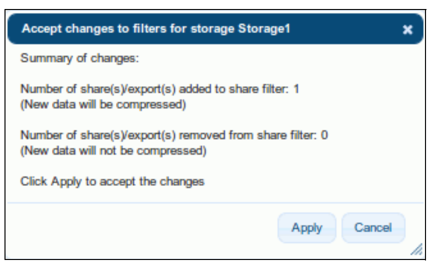


Figure 9-31 Confirm the new compression filter

The display will then update to show the new compression filter.

At this point any new (or updated) data written to this NFS export will be compressed by the RTCA product.

Configuring file extensions to exclude

This section describes how to filter the compression engine to exclude specific file types.

Because some data types are either already compressed (for example, GZIP files) or naturally uncompressible (for example, PACS data), they must be excluded from the RTCA compression process.

To configure the file type exclusions:

1. Log in to the RTCA web interface.
2. In the left-hand menu, under Configuration, select **Storage**.
3. In the Storage systems view, select your NAS controller.
4. In the lower left corner, select **File extensions to exclude**. A new dialog box opens, as shown in Figure 9-32.

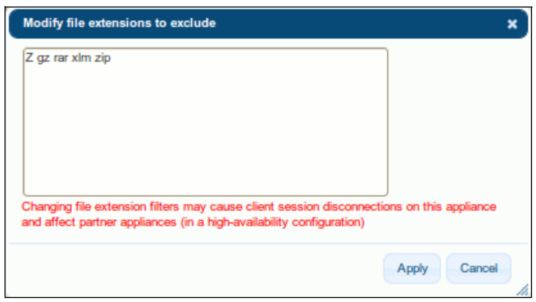


Figure 9-32 Configuring file extensions to exclude

5. Update the exclusion list as required, and then click **Apply** to save your changes.

From this point, any new data that matches these file extensions will not be compressed by the RTCA product.

9.2.5 Compressing existing data

This section describes how to use the Compression Accelerator to compress the existing data on an NFS export or group of exports. Setting an NFS share to compressed mode (see 9.2.4, “Compressing new data” on page 163) will only compress the new or changed data on the share. Any existing data will remain uncompressed until it is updated by the client. The RTCA product provides a *Compression Accelerator* feature to compress the existing data independent of any NAS client access.

Compression accelerator

This feature will read, compress, and rewrite the existing data on the nominated NFS export. This accelerates the space savings that are delivered by the data compression. Any subsequent client access will then be to the compressed form of the data.

An important difference between the compressed share setting and the Compression Accelerator, is that the compression setting is ongoing, while the CA process is typically only run once to process the existing data.

To configure the Compression Accelerator for the sample NFS export:

1. Log in to the RTCA web interface.
2. In the left-hand menu, under Configuration, select **Storage**.
3. In the Storage systems view, select your NAS controller.

   Ensure that the icon beside your storage controller is green, as shown in Figure 9-33:

   – This indicates that this is the active RTCA node for that storage interface.

   – If not, then you must check the other RTCA node or investigate why the NAS storage interface is offline.

Figure 9-33 Enable the Compression Accelerator for an NFS export

4. In the Storage Interfaces view, select **Compression Accelerator**. A new panel opens, as shown in Figure 9-34 on page 167.
5. In the Settings tab, enter the following information:

   – Scheduling = (set as required, for example: 17:00 - 20:00)

   – CIFS username = (Not required for NFS access)

   – CIFS password = (Not required for NFS access)

   – CIFS domain = (Not required for NFS access)

   – Policy = (if desired, restrict the CA process by file size, timestamp, and so on)

   – Directories to ignore = (typically used to prevent the CA from processing snapshot copies)

   – Maximum total throughput (MB/sec) = (if desired, throttle the CA process to a specified throughput)

   – Amount of changes = (if desired, stop the CA process after the specified amount of change within the time frame)

   – Free space threshold = (if desired, stop the CA process if the free space on the share reaches this limit)

   – Interval between files (ms) = (if desired, throttle the CA process by imposing a delay between file processing)

6. Click **Apply Changes**.

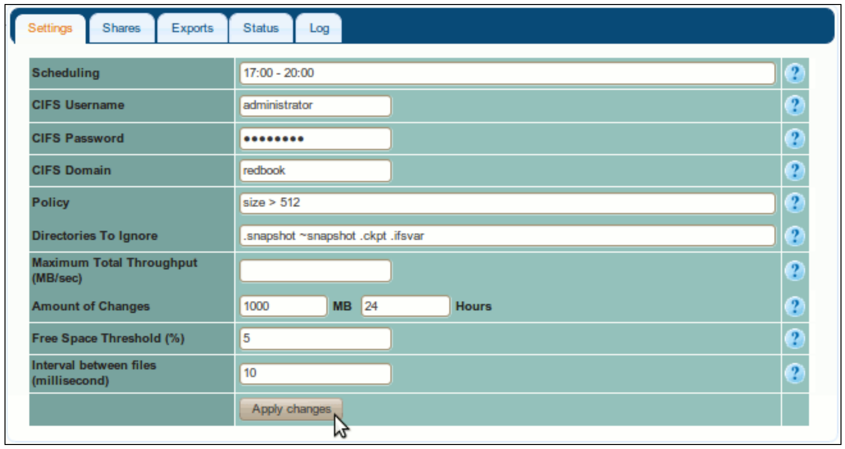


Figure 9-34 Configure the Compression Accelerator

7. Skip the Shares tab.
8. In the Exports tab (shown in Figure 9-35), select the NFS export to compress.
9. Click **Update List**.

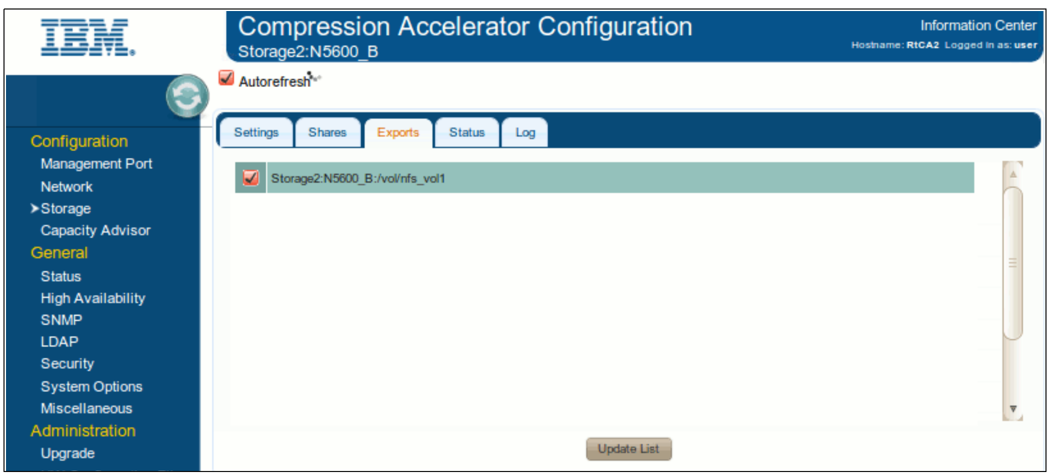


Figure 9-35 Select an NFS export for the Compression Accelerator

10. In the Status tab (shown in Figure 9-36 on page 168), review the current status.
11. Click **Start** to initiate the CA process to update the display to show the files being compressed.

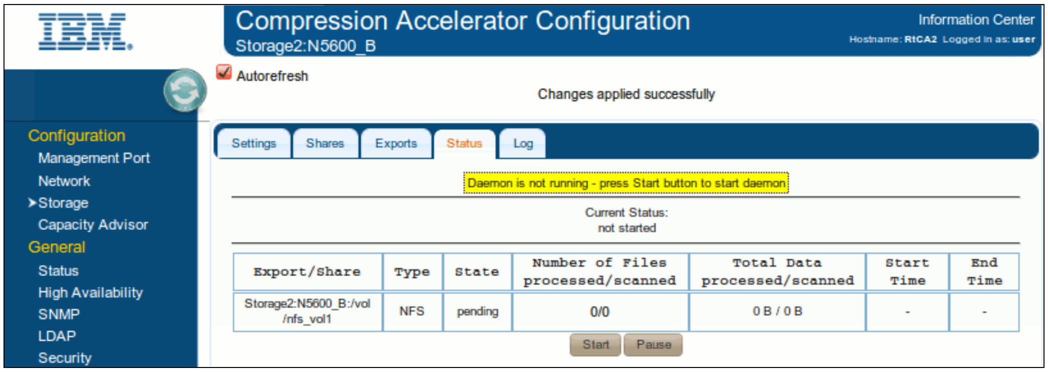


Figure 9-36 Start the Compression Accelerator

12. In the Log tab (shown in Figure 9-37), review the summary of the CA process.

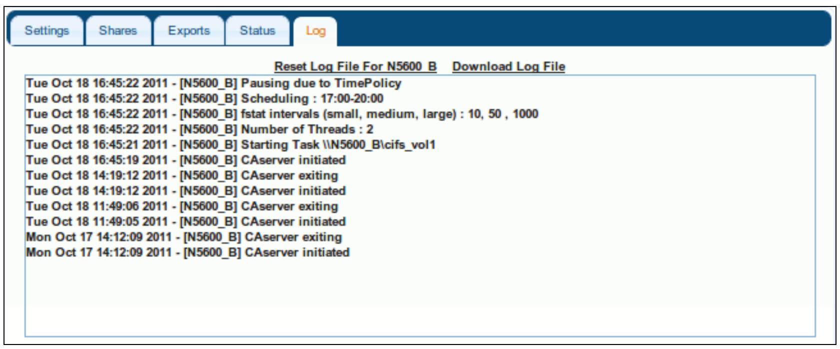


Figure 9-37 Review the log file for the Compression Accelerator

After the Compression Accelerator process has finished successfully, all of the existing data on the NFS export will be in a compressed format.

Existing data in compressed mode

At the end of our RTCA lab, all of the data on the NFS export had been compressed.

The reduction in used capacity is shown in Example 9-26. The compression ratio is representative of our particular data set. You must test the RTCA product in your own environment to determine the estimated capacity saving with your data set.

Example 9-26 Example data capacity after compression

```
N5600-B> df -Sh /vol/nfs_vol1
Filesystem                used       compressed         a-sis     %saved
/vol/nfs_vol1/          2184MB              0MB           0MB         0%
```

After the RTCA compression and Compression Accelerator completed, there was a large change in the Active File System (IBM AFS®) versus the snapshot backup. In our example, this was still below the 20% snapshot reserve, so it had no impact on the available file system capacity.

> **Important:** In a production environment, you must consider the impact of RTCA compression on the snapshot reserve and usable volume capacity. It might be prudent to delete some or all snapshots prior to running the Compression Accelerator.

9.2.6 Enabling NAS deduplication

One of the features of the RTCA product is that the compressed data is written in a manner that is compatible with subsequent deduplication on the NAS controller. This allows us to use the storage efficiency benefits of both RTCA compression and NAS deduplication. Example 9-27 shows how to enable NAS deduplication on our sample volume.

Example 9-27 Enabling deduplication on the NAS controller

```
NAS> sis on /vol/nfs_vol1

SIS for "/vol/nfs_vol1" is enabled.
Already existing data could be processed by running "sis start -s /vol/nfs_vol1".

NAS> sis start -s /vol/nfs_vol1

The file system will be scanned to process existing data in /vol/nfs_vol1.
This operation may initialize related existing metafiles.
Are you sure you want to proceed (y/n)? y
The SIS operation for "/vol/nfs_vol1" is started.

[wait for the deduplication process to finish...]

NAS> df -Sh /vol/nfs_vol1
Filesystem                 used       compressed          a-sis     %saved
/vol/nfs_vol1/            646MB              0MB         1542MB        70%
```

Of course, on the NFS client, the compressed data is still readable in its original form. The file system capacity is reported correctly, as shown in Example 9-28.

Example 9-28 Viewing the logical data capacity on the CIFS client

```
user@host:/nfs/nfs_vol1$ du -h .

[directoy listing not shown]

...
...
9.4G
```

However, the true used capacity can be seen on the client with the **df** command, as shown in Example 9-29.

Example 9-29 Viewing the physical data capacity on the CIFS client

```
user@host:/nfs/nfs_vol1$ df -m /nfs/nfs_vol1

Filesystem           1M-blocks       Used    Available   Use%   Mounted on
nas2:/vol/nfs_vol1       40960        647        40314     2%   /nfs/nfs_vol1
```

The slight variation in reported capacity is due to the different ways that the NAS controller and UNIX report disk usage.

9.2.7 Accessing the compressed data

This section describes the NFS client's view of the NAS data after it has been compressed by the RTCA product.

Accessing data through the compressed path

When the RTCA is configured correctly, the NFS client can access the compressed data exactly as expected. The file contents, and all file meta-data, are apparently unchanged.

Example 9-30 shows a file that was accessed from a compressed (and deduplicated) NFS export.

Example 9-30 Correct access to the compressed data

```
stevpem@tiamut:/$ file /nfs/nfs_vol1/FreeBase/data.aaaa

/nfs/nfs_vol1/FreeBase/data.aaaa: UTF-8 Unicode English text

stevpem@tiamut:/$ head --lines=1 /nfs/nfs_vol1/FreeBase/data.aaaa

/m/010001k/common/topic/notable_types/music/track
```

Accessing data through the transparent path

If the RTCA is configured incorrectly, the NFS client will not be able to access the compressed data. The file contents will appear corrupt to the NFS client.

For the purpose of demonstration only, the volume containing the sample NFS export was duplicated on the NAS controller by the NAS **volcopy** command. This volume contained the RTCA compressed data, but was configured for transparent access, bypassing the RTCA product's decompression algorithms.

The result of incorrectly accessing the compressed data can be seen in Example 9-31.

Example 9-31 Incorrect access to the compressed data

```
stevpem@tiamut:/nfs/nfs_vol2$ file /nfs/nfs_vol2/data.aaaa

/nfs/nfs_vol2/data.aaaa: data

stevpem@tiamut:/$ head --bytes=100 /nfs/nfs_vol2/data.aaaa

CMPt?;?Qq$???a??????x8?????????????????????????????????????????????????????????????
```

Example 9-31 shows the same file as Example 9-30, but now it is unusable. The first three characters of the file, "CMP" identify this as an RTCA compressed file.

However, any files that had been excluded from compression (for example, *.zip files) will still appear as normal, because they were not processed by the RTCA product.

> **Note:** During normal operations, access to the compressed data should always be through the RTCA compressed path. However, this is the way the files will appear if the RTCA is unavailable (for example, at a DR site with no local RTCA product).

9.2.8 Calculating the compression ratio

Although it is not the intent of this book to provide guidance regarding RTCA performance or compression ratios, Table 9-2 shows the compression ratio from our lab environment.

Table 9-2 Example compression ratios (with compression and deduplication)

	Original size	After RTCA	After NAS dedupe
NFS_VOL1	8764 MB	2184 MB	646 MB
		Approx 75% reduction	Approx 70% reduction
			Total 93% reduction

Obviously, the RTCA performance and compression ratios will vary depending on the NAS hardware platform and specific data set used. We encourage you to test the product in your environment before any production implementation.

Abbreviations and acronyms

ACL	Access Control List
BPDU	Bridge Protocol Data Unit
CLI	Command Line Interface
DBCS	Double-Byte Character Set
DNFS	Direct NFS
DR	Disaster Recovery
DSA	Dynamic System Analysis
ESCC	European Storage Competence Center
GUI	Graphical User Interface
HA	High Availability
I/O	Input/Output
IBM	International Business Machines Corporation
ILM	Information Life-cycle Management
IPL	Initial Program Load
IT	Information Technology
ITSO	International Technical Support Organization
LACP	Link Aggregation Control Protocol
LDAP	Lightweight Directory Access Protocol
LSM	Link Status Mirroring
MIB	Management Information Base
MRM	Monitoring and Reporting Manager
MTU	Maximum Transmission Unit
NAS	Network Attached Storage
NNC	No New Compression
RACE	Random Access Compression Engine
RTCA	IBM Real-time Compression Appliance
SNMP	Simple Network Management Protocol
STP	Spanning Tree Protocol
TPC	Transaction Processing Performance Council
UNC	Uniform Naming Convention
UPM	Unified Protocol Manager
VIF	Virtual Network Interface
VLAN	virtual LAN
VM	Virtual Machine

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

IBM Redbooks publications

The following IBM Redbooks publications provide additional information about the topics in this document. Note that some publications referenced in this list might be available in softcopy only:

- *Introduction to IBM Real-time Compression Appliances,* SG24-7953
- *IBM System Storage Solutions Handbook,* SG24-5250
- *IBM System Storage N series Software Guide,* SG24-7129
- *IBM System Storage N series Hardware Guide*, SG24-7840
- *Implementing IBM Storage Data Deduplication Solutions*, SG24-7888
- *IBM System Storage N series MetroCluster,* REDP-4259
- *IBM System Storage N series MetroCluster Planning Guide,* REDP-4243
- *IBM System Storage N series with VMware vSphere 4.1*, SG24-7636

You can search for, view, or download Redbooks publications, Redpaper publications, Technotes, draft publications, and Additional materials, as well as order hardcopy Redbooks publications, at this Web site:

ibm.com/redbooks

Other publications

These publications are also relevant as further information sources:

- *Real-time Compression Appliances: Installation and Planning Guide, Version 3 Release 7,* GI13-1221-00
- *Real-time Compression Appliances: Administration Guide, Version 3 Release 7,* GI13-1221-00
- *Real-time Compression Appliances: CLI Command Reference, Version 3 Release 7*
- *IBM Real-time Compression Appliances Version 3 Release 7 - Release Notes*
- *IBM Real-time Compression Appliances Version 3 Release 8 - Release Notes*, located at:

  https://www-304.ibm.com/support/docview.wss?uid=ssg1S7003693

Online resources

These Web sites are also relevant as further information sources:

- IBM Real-time compression:

  http://www-03.ibm.com/systems/storage/solutions/rtc/
- IBM Real-time Compression Appliance STN6500:

  http://www-03.ibm.com/systems/storage/network/rtc/stn6500/
- IBM Real-time Compression Appliance STN6800:

  http://www-03.ibm.com/systems/storage/network/rtc/stn6800/index.html
- IBM RTCA Support overview:

  http://www.ibm.com/storage/support/rtc
- Reducing NAS Costs with Real-time Data Compression:

  http://wikibon.org/wiki/v/Reducing_NAS_Costs_with_Real-time_Data_Compression
- IBM ProtecTIER Deduplication Solutions:

  http://www-03.ibm.com/systems/storage/tape/protectier/
- Network-attached storage:

  http://www.ibm.com/systems/storage/network/
- IBM support: Documentation:

  http://www.ibm.com/support/entry/portal/Documentation
- IBM Storage – Network Attached Storage: Resources:

  http://www.ibm.com/systems/storage/network/resources.html
- Introduction to VMware vSphere:

  http://www.vmware.com/pdf/vsphere4/r40/vsp_40_intro_vs.pdf
- ESXi Configuration Guides:

  http://www.vmware.com/pdf/vsphere4/r41/vsp_41_esxi_server_config.pdf
  http://www.vmware.com/pdf/vsphere4/r40/vsp_40_esxi_server_config.pdf
- vSphere Datacenter Administration Guides:

  http://www.vmware.com/pdf/vsphere4/r41/vsp_41_dc_admin_guide.pdf
  http://www.vmware.com/pdf/vsphere4/r40/vsp_40_admin_guide.pdf
- All VMware vSphere 4 documentation is located here:

  http://www.vmware.com/support/pubs/vs_pubs.html.

Help from IBM

IBM Support and downloads:

ibm.com/support

IBM Global Services:

ibm.com/services

Index

IBM Redbooks

IBM Real Time Compression Appliance Application Integration Guide

(0.2"spine)
0.17"<->0.473"
90<->249 pages

IBM Real Time Compression Appliance Application Integration Guide

Get familiar with integrating database and virtualization environments

See how seamless compression integrates into storage settings

Take storage efficiency to the next level

Continuing its commitment to developing and delivering industry-leading storage technologies, IBM is introducing the IBM Real-time Compression Appliances for NAS, an innovative new storage offering that delivers essential storage efficiency technologies, combined with exceptional ease of use and performance.

In an era when the amount of information, particularly in unstructured files, is exploding, but budgets for storing that information are stagnant, IBM Real-time Compression technology offers a powerful tool for better information management, protection, and access. IBM Real-time Compression can help slow the growth of storage acquisition, reducing storage costs while simplifying both operations and management. It also enables organizations to keep more data available for use rather than storing it offsite or on harder-to-access tape, so they can support improved analytics and decision making.

IBM Real-time Compression Appliances provide online storage optimization through real-time data compression, delivering dramatic cost reduction without performance degradation.

This IBM Redbooks publication is an easy-to-follow guide that describes how to design solutions successfully using IBM Real-time Compression Appliances (IBM RTCAs). It explains best practices for RTCA solution design, application integration, and practical RTCA use cases. This is a companion book to *Introduction to IBM Real-time Compression Appliances*, SG24-7953.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-8029-00 ISBN 0738436798