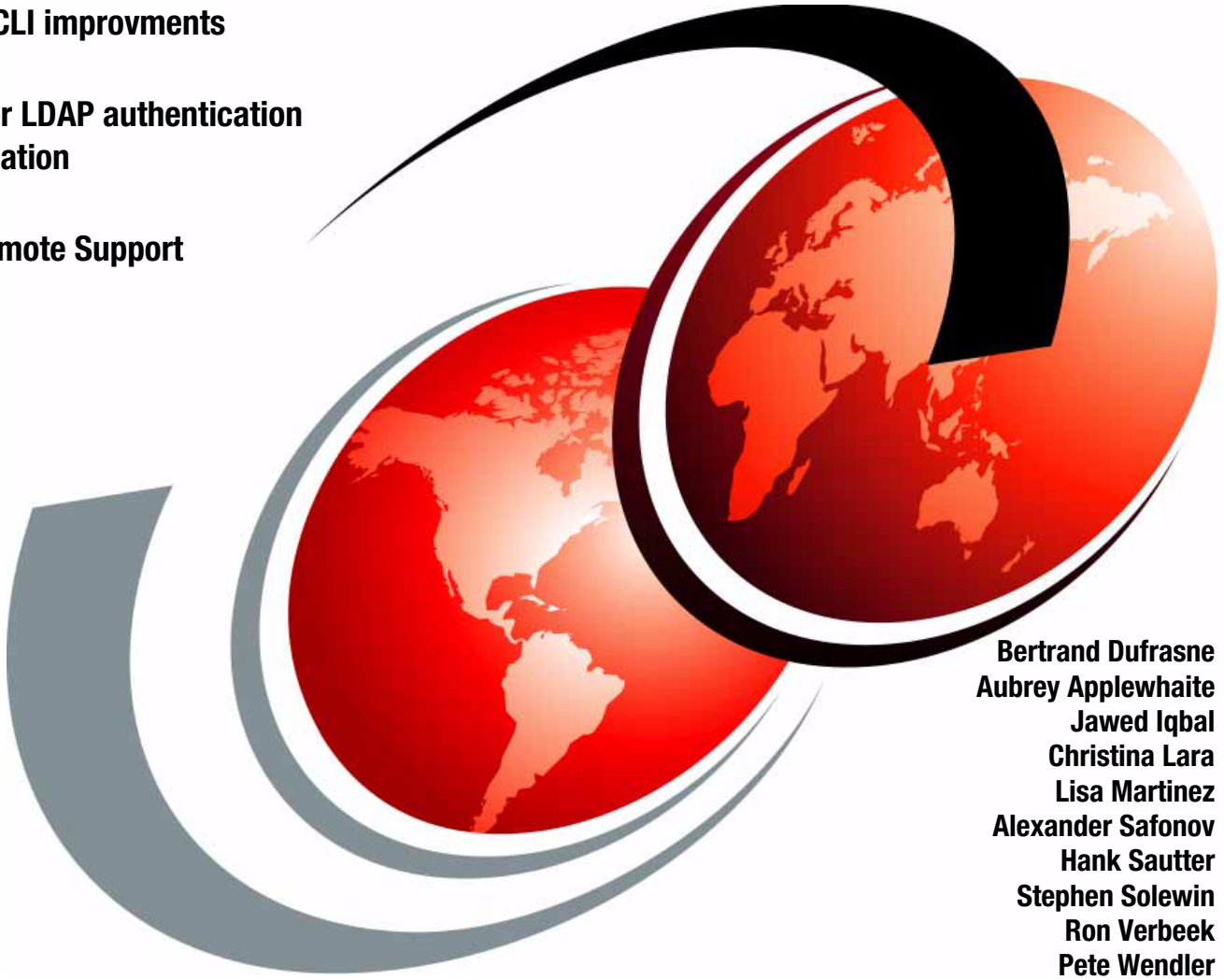
IBM

IBM XIV Storage System: Architecture, Implementation, and Usage

Non Disruptive Code load
GUI and XCLI improvments

Support for LDAP authentication
TPC Integration

Secure Remote Support



Bertrand Dufrasne
Aubrey Applewhaite
Jawed Iqbal
Christina Lara
Lisa Martinez
Alexander Safonov
Hank Sautter
Stephen Solewin
Ron Verbeek
Pete Wendler

Redbooks

ibm.com/redbooks

International Technical Support Organization

IBM XIV Storage System: Architecture, Implementation, and Usage

September 2009

SG24-7659-01

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

Second Edition (September 2009)

This edition applies to Version 10, Release 1, of the XIV Storage System software.

Contents

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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites 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:

AIX 5L™
AIX®
Domino®
DS8000®
FlashCopy®
IBM®
Lotus®
NetView®
Nextra™
Power Systems™
POWER5™
POWER6®
PowerVM™
POWER®
Redbooks®
Redbooks (logo) ®
S/390®
System i®
System p®
System Storage™
System x®
System z®
Tivoli®
WebSphere®
XIV®

The following terms are trademarks of other companies:

Emulex, HBAnyware, and the Emulex logo are trademarks or registered trademarks of Emulex Corporation.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

Snapshot, and the NetApp logo are trademarks or registered trademarks of NetApp, Inc. in the U.S. and other countries.

Novell, SUSE, the Novell logo, and the N logo are registered trademarks of Novell, Inc. in the United States and other countries.

Oracle, JD Edwards, PeopleSoft, Siebel, and TopLink are registered trademarks of Oracle Corporation and/or its affiliates.

QLogic, and the QLogic logo are registered trademarks of QLogic Corporation. SANblade is a registered trademark in the United States.

Red Hat, and the Shadowman logo are trademarks or registered trademarks of Red Hat, Inc. in the U.S. and other countries.

VMware, the VMware "boxes" logo and design are registered trademarks or trademarks of VMware, Inc. in the United States and/or other jurisdictions.

Java, Solaris, Sun, Sun Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Active Directory, ESP, Microsoft, MS, Windows Server, Windows Vista, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel Xeon, Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

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

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Mozilla®, Firefox®, as well as the Firefox logo are owned exclusively by the Mozilla Foundation . All rights in the names, trademarks, and logos of the Mozilla Foundation, including without limitation

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

Summary of changes

This section describes the technical changes made in this edition of the book and in previous editions. This edition may also include minor corrections and editorial changes that are not identified.

Summary of Changes
for SG24-7659-01
for IBM XIV Storage System: Architecture, Implementation, and Usage
as created or updated on February 12, 2010.

September 2009, Second Edition

This revision reflects the addition, deletion, or modification of new and changed information described below.

New information

- LDAP based authentication
- XIV® GUI and XCLI improvements
- Non Disruptive Code Load
- Support for VIOS clients
- Integration with Tivoli® Storage Productivity Center (TPC)
- XIV Remote Support Center (XRSC)

Changed information

- Hardware components and new machine type/model.
- Updates to host attachment to reflect new Host Attachment Kits (HAKs)
- Copy services will be covered in a separate publication

Preface

This IBM® Redbooks® publication describes the concepts, architecture, and implementation of the IBM XIV Storage System (2810-A14 and 2812-A14).

The XIV Storage System is designed to be a scalable enterprise storage system that is based upon a grid array of hardware components. It can attach to both Fibre Channel Protocol (FCP) and IP network Small Computer System Interface (iSCSI) capable hosts. This system is a good fit for clients who want to be able to grow capacity without managing multiple tiers of storage. The XIV Storage System is well suited for mixed or random access workloads, such as the processing of transactions, video clips, images, and e-mail, and industries, such as telecommunications, media and entertainment, finance, and pharmaceutical, as well as new and emerging workload areas, such as Web 2.0.

In the first few chapters of this book, we provide details about several of the unique and powerful concepts that form the basis of the XIV Storage System logical and physical architecture. We explain how the system was designed to eliminate direct dependencies between the hardware elements and the software that governs the system.

In subsequent chapters, we explain the planning and preparation tasks that are required to deploy the system in your environment. We present a step-by-step procedure describing how to configure and administer the system. We provide illustrations about how to perform those tasks by using the intuitive, yet powerful XIV Storage Manager GUI or the Extended Command Line Interface (XCLI).

This edition of the book contains comprehensive information on how to integrate the XIV Storage System for authentication in an LDAP environment.

The book also outlines the requirements and summarizes the procedures for attaching the system to various host platforms.

We also discuss the performance characteristics of the XIV system and present options available for alerting and monitoring, including an enhanced secure remote support capability.

This book is intended for those people who want an understanding of the XIV Storage System and also targets readers who need detailed advice about how to configure and use the system.

The team who wrote this book

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, San Jose Center.

Bertrand Dufrasne is an IBM Certified Consulting I/T Specialist and Project Leader for System Storage™ disk products at the International Technical Support Organization, San Jose Center. He has worked at IBM in various I/T areas. He has authored many IBM Redbooks publications and has also developed and taught technical workshops. Before joining the ITSO, he worked for IBM Global Services as an Application Architect. He holds a Masters degree in Electrical Engineering from the Polytechnic Faculty of Mons (Belgium).

Aubrey Applewhaite is an IBM Certified Consulting I.T Specialist working for the Storage Services team in the UK. He has worked for IBM since 1996 and has over 20 years experience in the I.T industry having worked in a number of areas, including System x® servers, operating system administration, and technical support. He currently works in a customer facing role providing advice and practical expertise to help IBM customers implement new storage technology. He specializes on XIV, SVC, DS8000®, and DS5000 hardware. He holds a Bachelor of Science Degree in Sociology and Politics from Aston University and is also a VMware® Certified Professional.

Christina Lara is a Senior Test Engineer, currently working on the XIV storage test team in Tucson, AZ. She just completed a one-year assignment as Assistant Technical Staff Member (ATSM) to the Systems Group Chief Test Engineer. Christina has just began her 9th year with IBM, having held different test and leadership positions within the Storage Division over that last several years. Her responsibilities included System Level Testing and Field Support Test on both DS8000 and ESS800 storage products and Test Project Management. Christina graduated from the University of Arizona in 1991 with a BSBA in MIS and Operations Management. In 2002, she received her MBA in Technology Management from the University of Phoenix.

Lisa Martinez is a Senior Software Engineer working in the DS8000 and XIV System Test Architecture in Tucson, Arizona. She has extensive experience in Enterprise Disk Test. She holds a Bachelor of Science degree in Electrical Engineering from the University of New Mexico and a Computer Science degree from New Mexico Highlands University. Her areas of expertise include the XIV Storage System and IBM System Storage DS8000, including Copy Services, with Open Systems and System z®.

Alexander Safonov is a Senior IT Specialist with System Sales Implementation Services, IBM Global Technology Services Canada. He has over 15 years of experience in the computing industry, with the last 10 years spent working on Storage and UNIX® solutions. He holds multiple product and industry certifications, including Tivoli Storage Manager, AIX®, and SNIA. Alexander spends most of his client contracting time working with Tivoli Storage Manager, data archiving, storage virtualization, replication, and migration of data. He holds an honors degree in Engineering from the National Aviation University of Ukraine.

Hank Sautter is a Consulting IT Specialist with Advanced Technical Support in the US. He has 17 years of experience with S/390® and IBM disk storage hardware and Advanced Copy Services functions while working in Tucson Arizona. His previous 13 years of experience include IBM Processor microcode development and S/390 system testing while working in Poughkeepsie, NY. He has worked at IBM for 30 years. Hank's areas of expertise include enterprise storage performance and disaster recovery implementation for large systems and open systems. He writes and presents on these topics. He holds a BS degree in Physics.

Stephen Solewin is an XIV Corporate Solutions Architect, based in Tucson, Arizona. He has 13 years of experience working on IBM storage, including Enterprise and Midrange Disk, LTO drives and libraries, SAN, Storage Virtualization, and software. Steve has been working on the XIV product line since March of 2008, working with both clients and various IBM teams worldwide. Steve holds a Bachelor of Science degree in Electrical Engineering from the University of Arizona, where he graduated with honors.

Ron Verbeek is a Senior Consulting IT Specialist with Storage & Data System Services, IBM Global Technology Services Canada. He has over 22 years of experience in the computing industry, with the last 10 years spent working on Storage and Data solutions. He holds multiple product and industry certifications, including SNIA Storage Architect. Ron spends most of his client time in technical pre-sales solutioning, defining and architecting storage optimization solutions. He has extensive experience in data transformation services and information lifecycle consulting. He holds a Bachelor of Science degree in Mathematics from McMaster University in Canada.

Figure 1 The team: Hank, Bertrand, Christina, Alexander, Stephen, Aubrey, Lisa, Ron

Jawed Iqbal is an Advisory Software Engineer and a team lead for Tivoli Storage Manager Client, Data Protection and FlashCopy® Manager products at the IBM Almaden Research Center in San Jose, CA. Jawed joined IBM in 2000 and worked as test lead on several Data Protection products, including Oracle® RDBMS Server, WebSphere®, MS® SQL, MS Exchange, and Lotus® Domino® Server. He holds a Masters degree in Computer Science, BBA in Computer Information Systems, Bachelor in Maths, Stats, and Economics. Jawed also holds an ITIL® certification.

Pete Wendler is a Software Engineer for IBM Systems and Technology Group, Storage Platform located in Tucson, Arizona. In his ten years working for IBM, Peter has worked in client support for enterprise storage products, solutions testing, development of the IBM DR550 archive appliance, and currently holds a position in technical marketing at IBM. Peter received a Bachelor of Science degree from Arizona State University in 1999

Special thanks to:

John Bynum
Worldwide Technical Support Management
IBM US, San Jose

For their technical advice, support, and other contributions to this project, many thanks to:

Rami Elron, Richard Heffel, Aviad Offer, Izhar Sharon, Omri Palmon, Orli Gan, Moshe Dahan, Dave Denny, Ritu Mehta, Eyal Zimran, Carlos Pratt, Darlene Ross, Juan Yanes, John Cherbini, Alice Bird, Alison Pate, Ajay Lunawat, Rosemary McCutchen, Jim Segdwick, Brian Sherman, Bill Wiegand, Barry Mellish, Dan Braden, Kip Wagner, Melvin Farris, Michael Hayut, Moriel Lechtman, Chip Jarvis, Russ Van Duine, Jacob Broido, Shmuel Vashdi, Avi Aharon, Paul Hurley, Martin Tiernan, Jayson Tsingine, Eric Wong, Theeraphong Thitayanun,
IBM

Robby Jackard,
ATS Group, LLC

Thanks also to the authors of the previous edition:

Marc Kremkus, Giacomo Chiapparini, Guenter Rebmann, Christopher Sansone, Attila Grosz, Markus Oscheka.

Become a published author

Join us for a two- to six-week residency program. Help write a book dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You will have the opportunity to team with IBM technical professionals, IBM Business Partners, and Clients.

Your efforts will help increase product acceptance and client satisfaction. As a bonus, you will develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out 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 IBM Redbooks publications form found at:

  ibm.com/redbooks
- Send your comments in an e-mail 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

1

IBM XIV Storage System overview

The IBM XIV Storage System is a fully scalable enterprise storage system that is based on a grid of standard, off-the-shelf hardware components. It has been designed with an easy to use and intuitive GUI that allows administrators to become productive in a very short time.

This chapter provides a high level overview of the IBM XIV Storage System.

1.1 Introduction

The XIV Storage System architecture is designed to deliver performance, scalability, and ease of management while harnessing the high capacity and cost benefits of Serial Advanced Technology Attachment (SATA) drives. The system employs off-the-shelf products as opposed to traditional offerings that use proprietary designs thus requiring more expensive components.

1.2 System models and components

The IBM XIV Storage System family consists of two machine types, the XIV Storage System (Machine type 2812) Model A14 and the XIV Storage System (Machine type 2810) Model A14. The 2812 Model A14 supports a 3-year warranty to complement the 1-year warranty offered by the existing and functionally equivalent, 2810 Model A14.

The majority of hardware and software features are the same for both machine types. The major differences are listed in Table 1-1.

Table 1-1 Machine type comparisons

Machine type	2810-A14	2812-A14
Warranty	1 year	3 years
CPUs per Interface Module	1 or 2	2
CPUs per Data Module	1	1

New orders for both machine types feature a new low voltage CPU (dual CPU in interface modules), for less power consumption.

Both machine types are available in the following configurations:

- 6 modules (including 3 Interface Modules)
- 9 -15 modules (including 6 Interface Modules)

Both machine types include the following components, which are visible in Figure 1-1:

- 3-6 Interface Modules, each with 12 SATA disk drives
- 3-9 Data Modules, each with 12 SATA disk drives
- An Uninterruptible Power Supply (UPS) module complex comprising three redundant UPS units
- Two Ethernet switches and an Ethernet Switch Redundant Power Supply (RPS)
- A Maintenance Module
- An Automatic Transfer Switch (ATS) for external power supply redundancy
- A modem, connected to the Maintenance Module for externally servicing the system (note that the modem (feature number 9101) is not available in all countries.

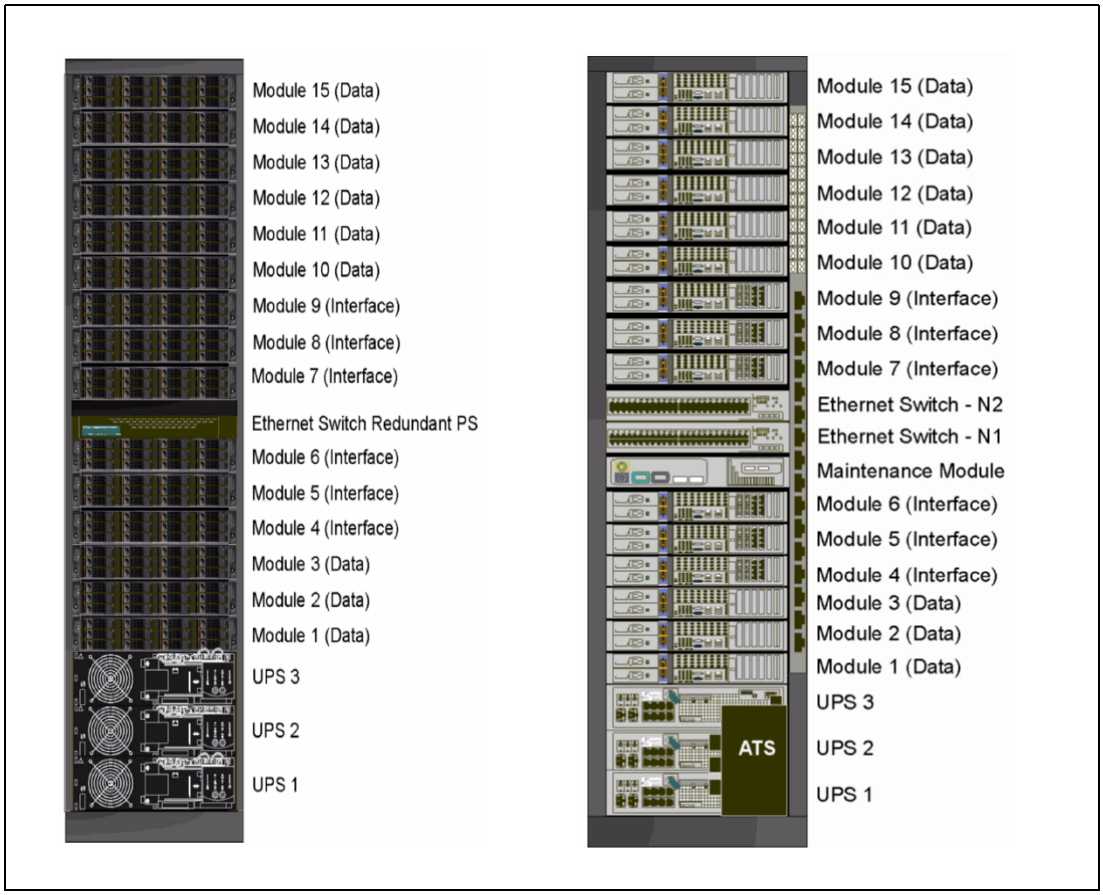


Figure 1-1 IBM XIV Storage System components: Front and rear view

All of the modules in the system are linked through an internal redundant Gigabit Ethernet network, which enables maximum bandwidth utilization and is resilient to at least any single component failure.

The system and all of its components come pre-assembled and wired in a lockable rack.

1.3 Key design features

This section describe the key design features of the XIV Storage System architecture.

We discuss these key design points and underlying architectural concepts in detail in Chapter 2, “XIV logical architecture and concepts” on page 9.

Massive parallelism

The system architecture ensures full exploitation of all system components. Any I/O activity involving a specific logical volume in the system is always inherently handled by all spindles. The system harnesses all storage capacity and all internal bandwidth, and it takes advantage of all available processing power, which is as true for host-initiated I/O activity as it is for system-initiated activity, such as rebuild processes and snapshot generation. All disks, CPUs, switches, and other components of the system contribute to the performance of the system at all times.

Workload balancing

The workload is evenly distributed over all hardware components at all times. All disks and modules are utilized equally, regardless of access patterns. Despite the fact that applications might access certain volumes more frequently than other volumes, or access certain parts of a volume more frequently than other parts, the load on the disks and modules will be balanced perfectly.

Pseudo-random distribution ensures consistent load-balancing even after adding, deleting, or resizing volumes as well as adding or removing hardware. This balancing of all data on all system components eliminates the possibility of a hot-spot being created.

Self-healing

Protection against double disk failure is provided by an efficient rebuild process that brings the system back to full redundancy in minutes. In addition, the XIV Storage System extends the self-healing concept, resuming redundancy even after failures in components other than disks.

True virtualization

Unlike other system architectures, storage virtualization is inherent to the basic principles of the XIV Storage System design. Physical drives and their locations are completely hidden from the user, which dramatically simplifies storage configuration, letting the system lay out the user's volume in the optimal way. The automatic layout maximizes the system's performance by leveraging system resources for each volume, regardless of the user's access patterns.

Thin provisioning

The system supports thin provisioning, which is the capability to allocate actual storage to applications on a just-in-time and as needed basis, allowing the most efficient use of available space and as a result, significant cost savings compared to traditional provisioning techniques. This is achieved by defining a logical capacity that is larger than the physical capacity and utilizing space based on what is consumed rather than what is allocated.

Processing power

The IBM XIV Storage System open architecture leverages the latest processor technologies and is more scalable than solutions that are based on a closed architecture. The IBM XIV Storage System avoids sacrificing the performance of one volume over another, and therefore requires little to no tuning.

1.4 The XIV Storage System software

The IBM XIV system software 10.1 (or later) provides the functions of the system, which include:

- **Bundled Advanced Features**

  All the features of the XIV including advanced features such as migration and mirroring are included free of charge and apply to the entire storage capacity.

- **Non-Disruptive Code Load (NDCL)**

  System software code can be upgraded without requiring downtime. This enables 'non-stop' production environments to remain running while new code is upgraded.

  The code upgrade is run on all modules in parallel and the process is fast enough to minimize impact on hosts applications.

No data migration or rebuild process is allowed during the upgrade. Mirroring, if any, will be suspended during the upgrade and automatically reactivated upon completion.

Storage management operations are also not allowed during the upgrade, although the status of the system and upgrade progress can be queried. It is also possible to cancel the upgrade process up to a point of no return.

Note that the NDCL does not apply to specific components firmware upgrades (for instance, module BIOS and HBA firmware). Those require a phase in / phase out process of the impacted modules.

- **Support for 16 000 snapshots**

  The snapshot capabilities within the XIV Storage System Software utilize a metadata, redirect-on-write design that allows snapshots to occur in a subsecond time frame with little performance overhead. Up to 16 000 full or differential copies can be taken. Any of the snapshots can be made writable, and then snapshots can be taken of the newly writable snapshots. Volumes can even be restored from these writable snapshots.

- **Synchronous remote mirroring to another XIV Storage System**

  Synchronous remote mirroring can be performed over Fibre Channel (FC) or IP network Small Computer System Interface (iSCSI) connections. Synchronous remote mirroring is used when data at local and remote sites must remain synchronized at all times.

- **Support for thin provisioning**

  Thin provisioning allows administrators to over-provision storage within storage pools; this is done by defining logical volume sizes that are larger than the physical capacity of the pool. Unlike other approaches, the physical capacity only needs to be larger than the actual written data, not larger than the logical volumes. Physical capacity of the pool needs to be increased only when actual written data increases.

- **Support for in-band data migration of heterogeneous storage**

  The XIV Storage System is also capable of acting as a host, gaining access to volumes on an existing legacy storage system. The XIV is then configured as a proxy to respond to requests between the current hosts and the legacy storage while migrating all existing data in the background. In addition, XIV supports thick-to-thin data migration, which allows the XIV Storage System to reclaim any allocated space that is not occupied by actual data.

- **Authentication using Lightweight Directory Access Protocol (LDAP)**

  LDAP can be used to provide user logon authentication allowing the XIV Storage System to integrate with Microsoft® Active Directory® (AD) or Sun™ Java™ Systems Directory Server (formerly Sun ONE Directory). Multiple directory servers can be configured to provide redundancy should one become unavailable.

- **Robust user auditing with access control lists**

  The XIV Storage System Software offers the capability for robust user auditing with Access Control Lists (ACLs) in order to provide more control and historical information.

- **Support for Tivoli Storage Productivity Center (TPC)**

  TPC can now discover XIV Storage Systems and all internal components, manage capacity for storage pools including allocated, unallocated, and available capacity with historical trending on utilization. It can also receive events and define policy-based alerts based on user-defined triggers and thresholds.

IBM XIV Storage Manager GUI

The XIV Storage Manager GUI acts as the management console for the XIV Storage System. A simple and intuitive GUI enables storage administrators to manage and monitor all system aspects easily, with almost no learning curve. Figure 1-2 shows one of the top level configuration panels.

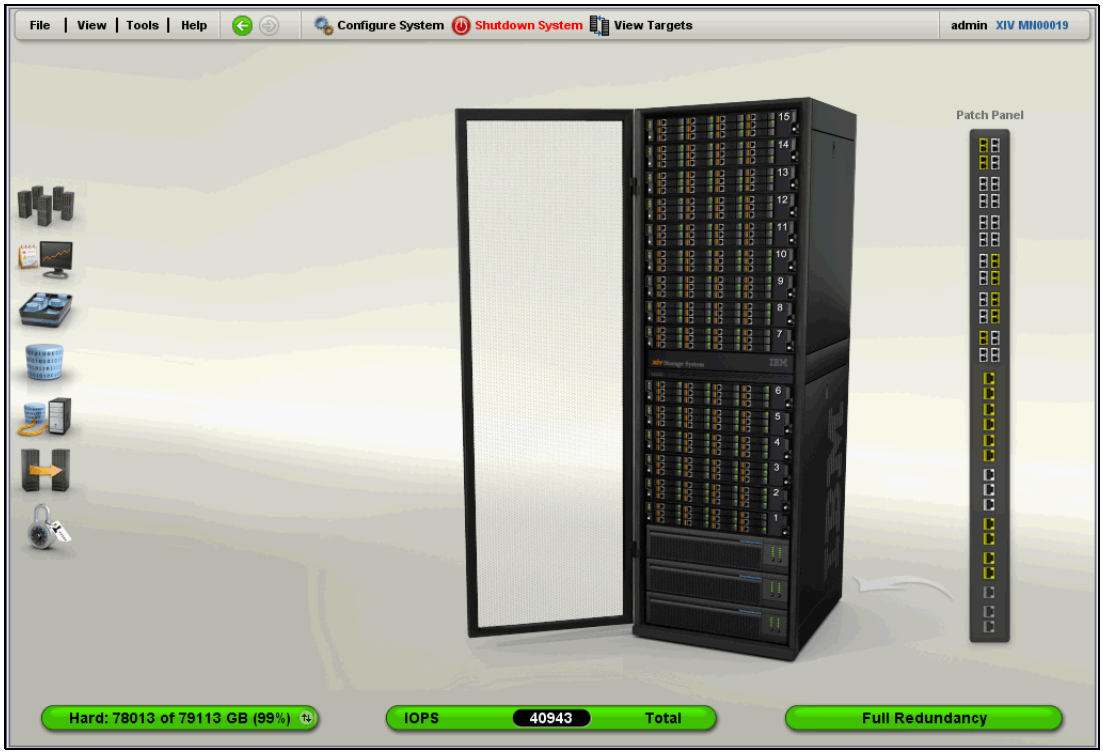


Figure 1-2 The IBM XIV Storage Manager GUI

The GUI is also supported on the following platforms:

- Microsoft Windows® 2000, Windows ME, Windows XP, Windows Server® 2003, Windows Vista®
- Linux® (Red Hat® 5.x or equivalent)
- AIX 5.3, AIX 6
- Solaris™ v9, Solaris v10
- HPUX 11i v2, HPUX 11i v3

The GUI can be downloaded at: `ftp://ftp.software.ibm.com/storage/XIV/GUI/`

It also contains a demo mode. To use the demo mode, log on as user `P10DemoMode` and no password.

Note that GUI and XCLI are packaged together.

IBM XIV Storage System XCLI

The XIV Storage System also offers a comprehensive set of Extended Command Line Interface (XCLI) commands to configure and monitor the system. All the functions available in the GUI are also available in the XCLI. The XCLI can be used in a shell environment to interactively configure the system or as part of a script to perform lengthy and/or complex tasks. Figure 1-3 shows a command being run in the XCLI interactive mode (XCLI session).

```
>> config_get
Name                       Value
dns_primary
dns_secondary
email_reply_to_address
email_sender_address
email_subject_format       {severity}: {description}
iscsi_name                 iqn.2005-10.com.xivstorage:000019
machine_model              A14
machine_serial_number      MN00019
machine_type               2810
ntp_server
snmp_community             XIV
snmp_contact               Unknown
snmp_location              Unknown
snmp_trap_community        XIV
support_center_port_type   Management
system_id                  19
system_name                XIV MN00019
```

Figure 1-3 The XCLI interactive mode

The XCLI is supported on:

- Microsoft Windows 2000, Windows ME, Windows XP, Windows Server 2003, Windows Vista
- Linux (Red Hat 5.x or equivalent)
- AIX 5.3, AIX 6
- Solaris v9, Solaris v10
- HPUX 11i v2, HPUX 11i v3

The XCLI can be downloaded at: ftp://ftp.software.ibm.com/storage/XIV/GUI/

Note that GUI and XCLI are packaged together.

1.5 Host support

The IBM XIV Storage System can be attached to a variety of host operating systems. Table 1-2 lists some of them, as well as the minimum version supported as of the time of writing (June 2009).

Table 1-2 Operating system support

Host Operating System	Minimum Supported Level
AIX	5.3 TL7
ESX	3.0
HP-UX	11iv1
Linux - CentOS	Enterprise Linux 4.6
Linux - RHEL	Enterprise Linux 4.6
Linux - SuSE	SLES 10
Macintosh	OS X 10.4.10
Power VM (Virtual IO Server)	VIOS 2.1.1
Solaris	9
SVC	4.3.0.1
Windows	2003 SP1 (Windows 2000 SP4 with RPQ)

For up-to-date information, refer to the XIV interoperability matrix or the System Storage Interoperability Center (SSIC) at:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

2

XIV logical architecture and concepts

This chapter elaborates on several of the XIV underlying design and architectural concepts that were introduced in the executive overview chapter.

The topics described in this chapter include:

- Architectural elements
- Parallelism
- Virtualization
- Data distribution
- Thin provisioning
- Self-healing and resiliency
- Rebuild redundancy

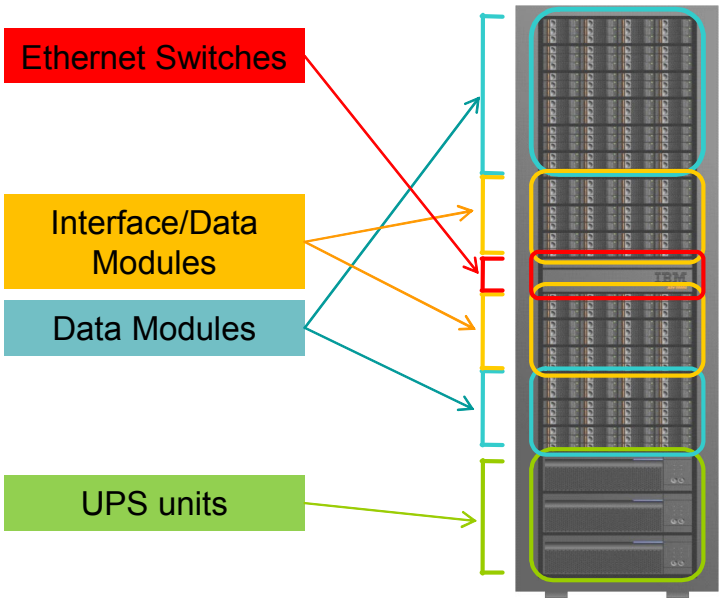
2.1 Architecture overview

The XIV Storage System architecture incorporates a variety of features designed to uniformly distribute data across internal resources. This unique data distribution method fundamentally differentiates the XIV Storage System from conventional storage subsystems, thereby offering numerous availability, performance, and management benefits across both physical and logical elements of the system.

Hardware elements

In order to convey the conceptual principles that comprise the XIV Storage System architecture, it is useful to first provide a glimpse of the physical infrastructure. Further details are covered in Chapter 3, “XIV physical architecture, components, and planning” on page 43.

The primary components of the XIV Storage System are known as *modules*. Modules provide processing, cache, and host interfaces and are based on “off the shelf” Intel® based systems. They are redundantly connected to one another through an internal switched Ethernet network, as shown in Figure 2-1. All of the modules work together concurrently as elements of a grid architecture, and therefore, the system harnesses the powerful parallelism inherent in such a distributed computing environment. We discuss the grid architecture in 2.2, “Parallelism” on page 12.



Although externally similar in appearance, Data and Interface/Data Modules differ in functions, interfaces, and in how they are interconnected.

Figure 2-1 IBM XIV Storage System major hardware elements

Data Modules

At a conceptual level, the Data Modules function as the elementary “building blocks” of the system, providing storage capacity, processing power, and caching, in addition to advanced system-managed services. The Data Module’s ability to share and manage system software and services are key elements of the physical architecture, as depicted in Figure 2-2.

Interface Modules

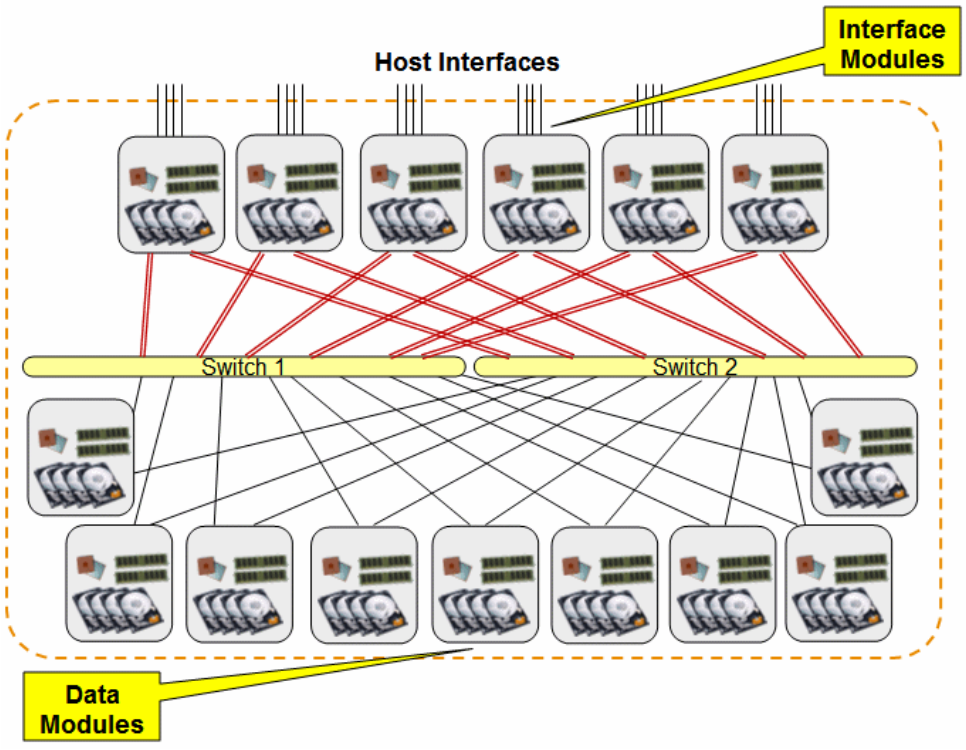
Interface Modules are equivalent to Data Modules in all aspects, with the following exceptions:

- In addition to disk, cache, and processing resources, Interface Modules are designed to include both Fibre Channel and iSCSI interfaces for host system connectivity, Remote Mirroring, and Data Migration activities. Figure 2-2 conceptually illustrates the placement of Interface Modules within the topology of the XIV IBM Storage System architecture.
- The system services and software functionality associated with managing external I/O reside exclusively on the Interface Modules.

Ethernet switches

The XIV Storage System contains a redundant switched Ethernet network that transmits both data and metadata traffic between the modules. Traffic can flow in any of the following ways:

- Between two Interface Modules
- Between two Data Modules
- Between an Interface Module and a Data Module



Interface and Data Modules are connected to each other through an internal IP switched network.

Figure 2-2 Architectural overview

Note: Figure 2-2 depicts the conceptual architecture only. Do not misinterpret the number of connections and such as a precise hardware layout.

2.2 Parallelism

The concept of *parallelism* pervades all aspects of the XIV Storage System architecture by means of a balanced, redundant data distribution scheme in conjunction with a pool of distributed (or grid) computing resources. In order to explain the principle of parallelism further, it is helpful to consider the ramifications of both the hardware and software implementations independently. We subsequently examine virtualization principles in 2.3, “Full storage virtualization” on page 14.

Important: The XIV Storage System exploits parallelism at *both* the hardware and software levels.

2.2.1 Hardware parallelism and grid architecture

The XIV grid design (Figure 2-3) entails the following characteristics:

- Both Interface Modules and Data Modules work together in a distributed computing sense. However, the Interface Modules also have additional functions and features associated with host system connectivity.
- The modules communicate with each other through the internal, redundant Ethernet network.
- The software services and distributed computing algorithms running within the modules collectively manage all aspects of the operating environment.

Design principles

The XIV Storage System grid architecture, by virtue of its distributed topology and “off the shelf” Intel components, ensures that the following design principles are possible:

- Performance:
  - The relative effect of the loss of a module is minimized.
  - All modules are able to participate equally in handling the total workload.

    This design principle is true regardless of access patterns. The system architecture enables excellent load balancing, even if certain applications access certain volumes, or certain parts within a volume, more frequently.
- Compatibility:
  - Modules consist of standard “off the shelf” components.

    Because components are not specifically engineered for the system, the resources and time required for the development of newer hardware technologies are minimized. This benefit, coupled with the efficient integration of computing resources into the grid architecture, enables the system to realize the rapid adoption of the newest hardware technologies available without the need to deploy a whole new subsystem.
- Scalability:
  - Computing resources can be dynamically changed
  - “Scaled out” by adding new modules to accommodate both new capacity and new performance demands
  - “Scaled up” by upgrading modules

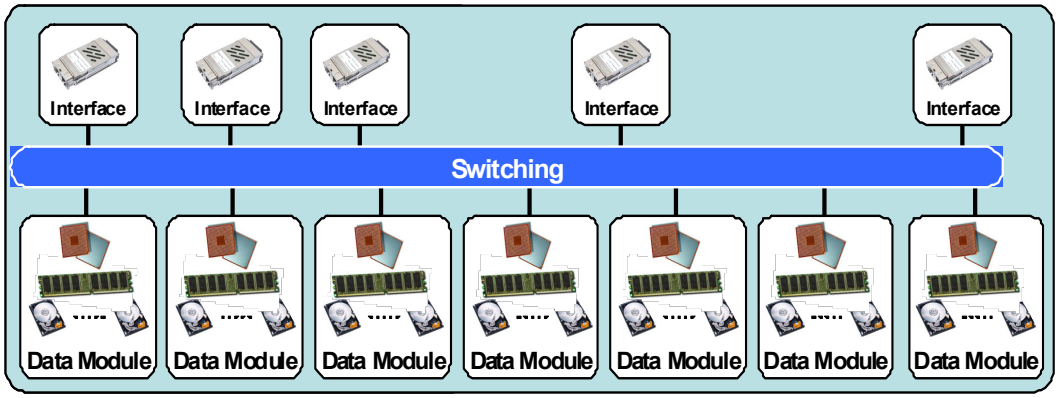


Figure 2-3 IBM XIV Storage System scalable conceptual grid architecture

Proportional scalability

Within the XIV Storage System, each module contains all of the pertinent hardware elements that are necessary for a grid topology (processing, caching, and storage). All modules are connected through a scalable network. This aspect of the grid infrastructure enables the relative proportions of cache, processor, disk, and interconnect bandwidth to remain optimal even in the event that modules are added or removed:

- **Linear cache growth**: The total system cache size and cache bandwidth increase linearly with disk capacity, because every module is a self-contained computing resource that houses its own cache. Note that the cache bandwidth scales linearly in terms of both host-to-cache and cache-to-disk throughput, and the close proximity of cache, processor, and disk is maintained.
- **Proportional interface growth**: Interface Modules house Ethernet and Fibre Channel host interfaces and are able to access not only the local resources within the module, but the entire system. With every Interface Module added, the system proportionally scales both the number of host interfaces and the bandwidth to the internal resources.
- **Constant switching capacity**: The internal switching capacity is designed to scale proportionally as the system grows, preventing bottlenecks regardless of the number of modules. This capability ensures that internal throughput scales proportionally to capacity.
- **Embedded processing power**: Because each module incorporates its own processing power in conjunction with cache and disk components, the ability of the system to perform processor-intensive tasks, such as aggressive prefetch caching, sophisticated cache updates, snapshot management, and data distribution, is always maintained regardless of of the system capacity.

2.2.2 Software parallelism

In addition to the hardware parallelism, the XIV Storage System also employs sophisticated algorithms to achieve optimal parallelism.

Modular software design

The XIV Storage System internal operating environment consists of a set of software functions that are loosely coupled with the hardware modules. These software functions reside on one or more modules and can be redistributed among modules as required, thus ensuring resiliency under changing hardware conditions.

An example of this modular design resides specifically in the interface modules. All six interface modules actively manage system services and software functionality associated with managing external I/O. Also, three of the interface modules deliver the system's management interface service for use with the XIV Storage System.

Data distribution algorithms

Data is distributed across all drives in a *pseudo-random* fashion. The patented algorithms provide a uniform yet random spreading of data across all available disks to maintain data resilience and redundancy. Figure 2-4 on page 17 provides a conceptual representation of the pseudo-random data distribution within the XIV Storage System.

For more details about the topic of data distribution and storage virtualization, refer to 2.3.1, "Logical system concepts" on page 16.

2.3 Full storage virtualization

The data distribution algorithms employed by the XIV Storage System are innovative in that they are deeply integrated into the system architecture itself, instead of at the host or storage area network level. The XIV Storage System is unique in that it is based on an innovative implementation of full storage virtualization within the system itself.

In order to fully appreciate the value inherent to the virtualization design that is used by the XIV Storage System, it is helpful to remember several aspects of the physical and logical relationships that comprise conventional storage subsystems. Specifically, traditional subsystems rely on storage administrators to carefully plan the relationship between logical structures, such as arrays and volumes, and physical resources, such as disk packs and drives, in order to strategically balance workloads, meet capacity demands, eliminate hot-spots, and provide adequate performance.

IBM XIV Storage System virtualization design

The implementation of full storage virtualization employed by the XIV Storage System eliminates many of the potential operational drawbacks that can be present with conventional storage subsystems, while maximizing the overall usefulness of the subsystem.

The XIV Storage System virtualization offers the following benefits:

- Easier volume management:
  - Logical volume placement is driven by the distribution algorithms, freeing the storage administrator from planning and maintaining volume layout. The data distribution algorithms manage all of the data in the system collectively without deference to specific logical volume definitions.
  - Any interaction, whether host or system driven, with a specific logical volume in the system is inherently handled by all resources; it harnesses all storage capacity, all internal bandwidth, and all processing power currently available in the system.
  - Logical volumes are not exclusively associated with a subset of physical resources.
    - Logical volumes can be dynamically resized.
    - Logical volumes can be thinly provisioned, as discussed in 2.3.4, "Capacity allocation and thin provisioning" on page 23.

- Consistent performance and scalability:
  - Hardware resources are always utilized equally, because all logical volumes always span all physical resources and are therefore able to reap the performance potential of the full system and maintain data integrity.
    - Virtualization algorithms automatically redistribute the logical volumes data and workload when new hardware is added, thereby maintaining the system balance while preserving transparency to the attached hosts.
    - In the event of a hardware failure, data is automatically, efficiently, and rapidly rebuilt across all the drives and modules in the system, thereby preserving host transparency, equilibrium, and data redundancy at all times while virtually eliminating any performance penalty associated with traditional RAID rebuilds.
  - There are no “pockets” of capacity, “orphaned” disk space, or resources that are inaccessible due to array mapping constraints or data placement.
- Flexible snapshots:
  - Full storage virtualization incorporates snapshots that are differential in nature; only updated data consumes physical capacity:
    - Many concurrent snapshots (Up to 16 000 volumes and snapshots can be defined.)

      Multiple concurrent snapshots are possible because a snapshot uses physical space only after a change has occurred on the source.
    - Multiple snapshots of a single master volume can exist independently of each other.
    - Snapshots can be cascaded, in effect, creating snapshots of snapshots.
  - Creation and deletion of snapshots do not require data to be copied and hence occur immediately.
  - When updates occur to master volumes, the system’s virtualized logical structure enables it to preserve the original point-in-time data associated with any and all dependent snapshots by redirecting the update to a new physical location on disk. This process, which is referred to as *redirect on write,* occurs transparently from the host perspective and uses the virtualized remapping of the updated data to minimize any performance impact associated with preserving snapshots, regardless of the number of snapshots defined for a given master volume.

    Note: The XIV snapshot process uses “redirect on write,” which is more efficient than the “copy on write” that is used by many other storage subsystems.

- Data migration efficiency:
  - XIV supports thin provisioning. When migrating from a system that only supports regular (or thick) provisioning, XIV allows thick-to-thin provisioning of capacity. Thin-provisioned capacity is discussed in 2.3.4, “Capacity allocation and thin provisioning” on page 23.
  - Due to the XIV pseudo-random distribution of data, the performance impact of data migration on production activity is minimized, because the load is spread evenly over all resources.

2.3.1 Logical system concepts

In this section, we elaborate on the logical system concepts, which form the basis for the system full storage virtualization.

Logical constructs

The XIV Storage System logical architecture incorporates constructs that underlie the storage virtualization and distribution of data, which are integral to its design. The logical structure of the system ensures that there is optimum granularity in the mapping of logical elements to both modules and individual physical disks, thereby guaranteeing an equal distribution of data across all physical resources.

Partitions

The fundamental building block of logical volumes is known as a *partition*. Partitions have the following characteristics on the XIV Storage System:

- All partitions are 1 MB (1024 KB) in size.
- A partition contains either a primary copy or secondary copy of data:
  - Each partition is mapped to a single physical disk:
    - This mapping is dynamically managed by the system through innovative data distribution algorithms in order to preserve data redundancy and equilibrium. For more information about the topic of data distribution, refer to “Logical volume layout on physical disks” on page 18.
    - The storage administrator has no control or knowledge of the specific mapping of partitions to drives.
  - Secondary copy partitions are always placed in a different *Module* than the one containing the primary copy partition.

Important: In the context of the XIV Storage System logical architecture, a partition consists of 1 MB (1024 KB) of data. Do not confuse this definition with other definitions of the term “partition.”

The diagram in Figure 2-4 illustrates that data is uniformly, yet randomly distributed over all disks. Each 1 MB of data is duplicated in a primary and secondary partition. For the same data, the system ensures that the primary partition and its corresponding secondary are not located within the same module.

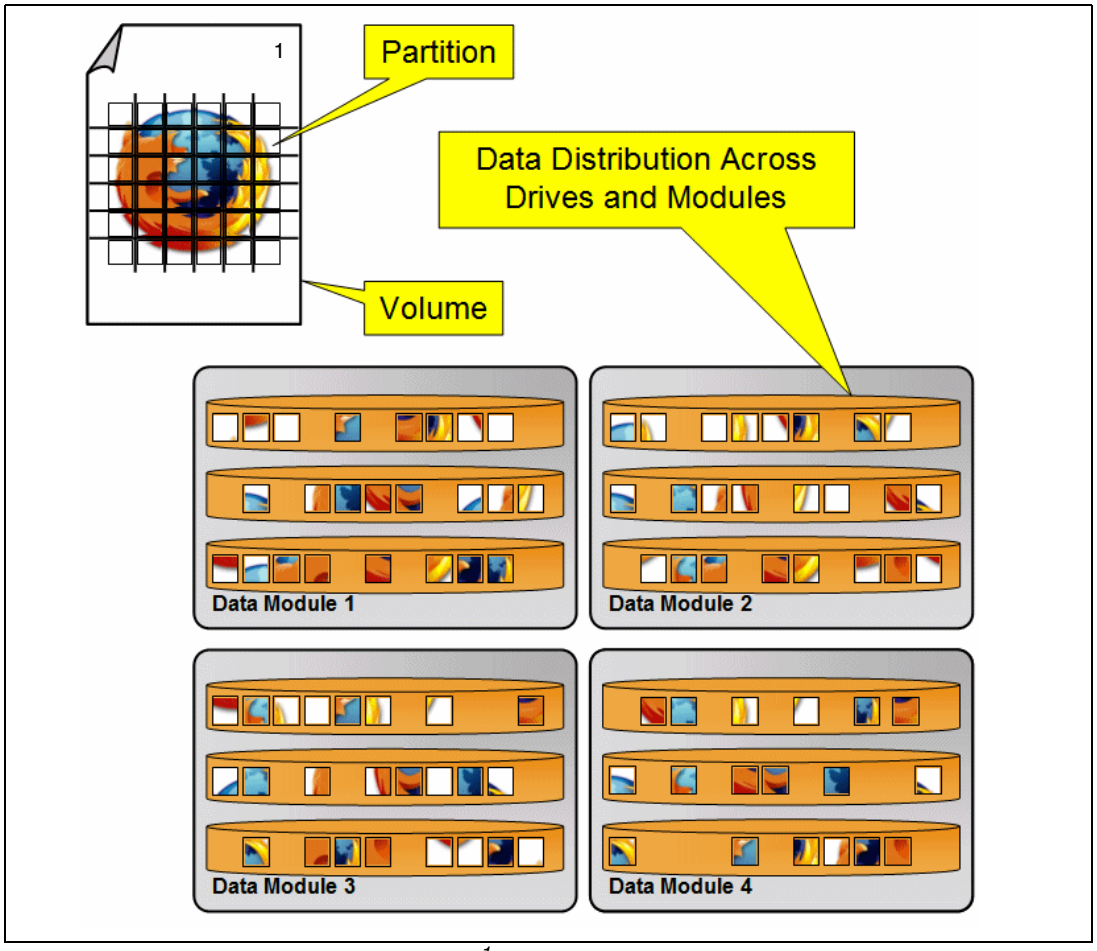


Figure 2-4 Pseudo-random data distribution[1]

Logical volumes

The XIV Storage System presents logical volumes to hosts in the same manner as conventional subsystems; however, both the granularity of logical volumes and the mapping of logical volumes to physical disks differ:

- As discussed previously, every logical volume is comprised of 1 MB (1024 KB) constructs of data known as partitions.
- The physical capacity associated with a logical volume is *always a multiple of 17 GB (decimal).*

  Therefore, while it is possible to present a block-designated logical volume to a host that is *not* a multiple of 17 GB, the *actual physical space* that is allocated for the volume will always be the sum of the minimum number of 17 GB increments needed to meet the block-designated capacity.

> **Note:** Note that the initial physical capacity actually allocated by the system upon volume creation can be less than this amount, as discussed in “Actual and logical volume sizes” on page 23.

[1] Copyright 2005-2008 Mozilla. All Rights Reserved. All rights in the names, trademarks, and logos of the Mozilla Foundation, including without limitation, Mozilla, Firefox, as well as the Firefox logo, are owned exclusively by the Mozilla Foundation.

- The maximum number of volumes that can be concurrently defined on the system is limited by:
  - The logical address space limit:
    - The logical address range of the system permits up to 16 377 volumes, although this constraint is purely logical, and therefore, is not normally a practical consideration.
    - Note that the same address space is used for both volumes and snapshots.
  - The limit imposed by the logical and physical topology of the system for the minimum volume size.

    The physical capacity of the system, based on 180 drives with 1 TB of capacity per drive and assuming the minimum volume size of 17 GB, limits the maximum volume count to 4 605 volumes. Again, a system with active snapshots can have more than 4 605 addresses assigned collectively to both volumes and snapshots, because volumes and snapshots share the same address space.

Important: The logical address limit is ordinarily not a practical consideration during planning, because under most conditions, this limit will not be reached; it is intended to exceed the adequate number of volumes for all conceivable circumstances.

Storage Pools

Storage Pools are administrative boundaries that enable storage administrators to manage relationships between volumes and snapshots and to define separate capacity provisioning and snapshot requirements for such uses as separate applications or departments. Storage Pools are not tied in any way to physical resources, nor are they part of the data distribution scheme. We discuss Storage Pools and their associated concepts in 2.3.3, “Storage Pool concepts” on page 20.

Snapshots

A *snapshot* represents a point-in-time copy of a volume. Snapshots are like volumes except snapshots incorporate dependent relationships with their source volumes, which can be either logical volumes or other snapshots. Because they are not independent entities, a given snapshot does not necessarily wholly consist of partitions that are unique to that snapshot. Conversely, a snapshot image will not share all of its partitions with its source volume if updates to the source occur after the snapshot was created.

Logical volume layout on physical disks

The XIV Storage System manages the distribution of logical volumes over physical disks and modules by means of a dynamic relationship between primary data partitions, secondary data partitions, and physical disks. This virtualization of resources in the XIV Storage System is governed by the data distribution algorithms.

Distribution table

The Distribution table is created at system startup, and contains a mapping of every primary and secondary partition, as well as the Module and physical disk they reside on. When hardware changes occur, a new Distribution table is created and delivered to every module. Each module retains redundant copies of the Distribution table.

Volume layout

At a conceptual level, the data distribution scheme can be thought of as an mixture of mirroring and striping. While it is tempting to think of this scheme in the context of RAID 1+0

(10) or 0+1, the low-level virtualization implementation precludes the usage of traditional RAID algorithms in the architecture.

As discussed previously, the XIV Storage System architecture divides logical volumes into 1 MB partitions. This granularity and the mapping strategy are integral elements of the logical design that enable the system to realize the following features and benefits:

- Partitions that make up a volume are distributed on all disks using what is defined as a *pseudo-random distribution function*, which was introduced in 2.2.2, "Software parallelism" on page 13.
  - The distribution algorithms seek to preserve the equality of access among all physical disks under all conceivable conditions and volume access patterns. Essentially, while not truly random in nature, the distribution algorithms in combination with the system architecture preclude the occurrence of "*hot-spots*":
    - A fully configured XIV Storage System contains 180 disks, and each volume is allocated across at least 17 GB (decimal) of capacity that is distributed evenly across all disks.
    - Each logically adjacent partition on a volume is distributed across a different disk; partitions are *not* combined into groups before they are spread across the disks.
    - The pseudo-random distribution ensures that logically adjacent partitions are never striped sequentially across physically adjacent disks. Refer to 2.2.2, "Software parallelism" on page 13 for a further overview of the partition mapping topology.
  - Each disk has its data mirrored across all other disks, *excluding the disks in the same module.*
  - Each disk holds approximately one percent of any other disk in other modules.
  - Disks have an equal probability of being accessed regardless of aggregate workload access patterns.

Note: When the number of disks or modules changes, the system defines a new data layout that preserves redundancy and equilibrium. This target data distribution is called the *goal distribution* and is discussed in "Goal distribution" on page 35.

- As discussed previously in "IBM XIV Storage System virtualization design" on page 14:
  - The storage system administrator does *not* plan the layout of volumes on the modules.
  - Provided that there is space available, volumes can always be added or resized instantly with negligible impact on performance.
  - There are no unusable pockets of capacity known as "orphaned spaces."
- When the system is scaled out through the addition of modules, a new *goal distribution* is created whereby just a minimum number of partitions are moved to the newly allocated capacity to arrive at the new distribution table.

  The new capacity is fully utilized within several hours and with no need for any administrative intervention. Thus, the system automatically returns to a state of equilibrium among all resources.
- Upon the failure or phase-out of a drive or a module, a new goal distribution is created whereby data in non-redundant partitions is copied and redistributed across the remaining modules and drives.

  The system rapidly returns to a state in which all partitions are again redundant, because all disks and modules participate in achieving the new goal distribution.

2.3.2 System usable capacity

The XIV Storage System reserves physical disk capacity for:

- Global spare capacity
- Metadata, including statistics and traces
- Mirrored copies of data

Global spare capacity

The dynamically balanced distribution of data across all physical resources by definition obviates the inclusion of dedicated spare drives that are necessary with conventional RAID technologies. Instead, the XIV Storage System reserves capacity on each disk in order to provide adequate space for the redistribution or rebuilding of redundant data in the event of a hardware failure.

This global spare capacity approach offers advantages over dedicated hot spare drives, which are used only upon failure and are not used otherwise, therefore reducing the number of spindles that the system can leverage for better performance. Also, those non-operating disks are typically not subject to background scrubbing processes, whereas in XIV, all disks are operating and subject to examination, which helps detect potential reliability issues with drives.

The global reserved space includes sufficient capacity to withstand the failure of a full module and a further three disks, and will still allow the system to execute a new *goal distribution,* and to return to full redundancy.

Important: The system will tolerate multiple hardware failures, including up to an entire module in addition to three subsequent drive failures *outside of the failed module*, provided that a new goal distribution is fully executed before a subsequent failure occurs. If the system is less than 100% full, it can sustain more subsequent failures based on the amount of unused disk space that will be allocated at the event of failure as a spare capacity. For a thorough discussion of how the system uses and manages reserve capacity under specific hardware failure scenarios, refer to 2.4, “Reliability, availability, and serviceability” on page 31.

Note: The XIV Storage System does not manage a global reserved space for snapshots. We explore this topic in the next section.

Metadata and system reserve

The system reserves roughly 4% of the physical capacity for statistics and traces, as well as the distribution table.

Net usable capacity

The calculation of the *net usable capacity* of the system consists of the total disk count, less disk space reserved for sparing (which is the equivalent of one module plus three more disks), multiplied by the amount of capacity on each disk that is dedicated to data (that is 96% because of metadata and system reserve), and finally reduced by a factor of 50% to account for data mirroring achieved via the secondary copy of data.

2.3.3 Storage Pool concepts

While the hardware resources within the XIV Storage System are virtualized in a global sense, the available capacity in the system can be administratively portioned into separate and independent Storage Pools. The concept of *Storage Pools* is purely administrative.

Essentially, Storage Pools function as a means to effectively manage a related group of similarly provisioned logical volumes and their snapshots.

Improved management of storage space

Storage Pools form the basis for controlling the usage of storage space by imposing a capacity quota on specific applications, a group of applications, or departments, enabling isolated management of relationships within the associated group of logical volumes and snapshots.

A *logical volume* is defined within the context of one and only one Storage Pool. As Storage Pools are logical constructs, a volume and any snapshots associated with it can be moved to any other Storage Pool, as long as there is sufficient space.

As a benefit of the system virtualization, there are no limitations on the size of Storage Pools or on the associations between logical volumes and Storage Pools. In fact, manipulation of Storage Pools consists exclusively of metadata transactions and does not trigger any copying of data. Therefore, changes are completed instantly and without any system overhead or performance degradation.

Consistency Groups

A *Consistency Group* is a group of volumes of which a snapshot can be made at the same point in time, thus ensuring a consistent image of all volumes within the group at that time. The concept of a Consistency Group is common among storage subsystems in which it is necessary to perform concurrent operations collectively across a set of volumes, so that the result of the operation preserves the consistency among volumes. For example, effective storage management activities for applications that span multiple volumes, or for creating point-in-time backups, is not possible without first employing Consistency Groups.

This consistency between the volumes in the group is paramount to maintaining data integrity from the application perspective. By first grouping the application volumes into a Consistency Group, it is possible to later capture a consistent state of all volumes within that group at a given point-in-time using a special snapshot command for Consistency Groups.

Issuing this type of a command results in the following process:

1. Complete and destage writes across the constituent volumes.
2. Instantaneously suspend I/O activity simultaneously across all volumes in the Consistency Group.
3. Create the snapshots.
4. Finally, resume normal I/O activity across all volumes.

The XIV Storage System manages these suspend and resume activities for all volumes within the Consistency Group.

Note: Note that additional mechanisms or techniques, such as those provided by the Microsoft Volume Shadow copy Services (VSS) framework, might still be required to maintain full application consistency.

Storage Pool relationships

Storage Pools facilitate the administration of relationships among logical volumes, snapshots, and Consistency Groups.

The following principles govern the relationships between logical entities within the Storage Pool:

- A logical volume can have multiple independent snapshots. This logical volume is also known as a *master volume*.
- A *master volume* and all of its associated snapshots are always a part of only one Storage Pool.
- A volume can only be part of a single Consistency Group.
- All volumes of a Consistency Group must belong to the same Storage Pool.

Storage Pools have the following characteristics:

- The size of a Storage Pool can range from 17 GB (the minimum size that can be assigned to a logical volume) to the capacity of the entire system.
- *Snapshot reserve capacity* is defined within each Storage Pool and is effectively *maintained separately* from logical, or master, volume capacity. The same principles apply for thinly provisioned Storage Pools, which are discussed in “Thinly provisioned storage pools” on page 24, with the exception that space is not *guaranteed* to be available for snapshots due to the potential for hard space depletion, which is discussed in “Depletion of hard capacity” on page 30:
  - Snapshots are structured in the same manner as logical, or master, volumes.

    Note: The snapshot reserve needs to be a minimum of 34 GB. The system preemptively deletes snapshots if the snapshots fully consume the allocated available space.

  - As mentioned before, snapshots will only be automatically deleted when there is inadequate physical capacity available within the context of each Storage Pool. This process is managed by a snapshot deletion priority scheme. Therefore, when the capacity of a Storage Pool is exhausted, only the snapshots that reside in the affected Storage Pool are deleted in order of the deletion priority.
- The space allocated for a Storage Pool can be dynamically changed by the storage administrator:
  - The Storage Pool can be increased in size. It is limited only by the unallocated space on the system.
  - The Storage Pool can be decreased in size. It is limited only by the space that is consumed by the volumes and snapshots that are defined within that Storage Pool.
- The designation of a Storage Pool as a regular pool or a thinly provisioned pool can be dynamically changed even for existing Storage Pools. Thin provisioning is discussed in-depth in 2.3.4, “Capacity allocation and thin provisioning” on page 23.
- The storage administrator can relocate logical volumes between Storage Pools without any limitations, provided there is sufficient free space in the target Storage Pool:
  - If necessary, the target Storage Pool capacity can be dynamically increased prior to volume relocation, assuming there is sufficient unallocated capacity available in the system.
  - When a logical volume is relocated to a target Storage Pool, sufficient space must be available for all of its snapshots to reside in the target Storage Pool as well.

Notes:

- When moving a volume into a Storage Pool, the size of the Storage Pool is not automatically increased by the size of the volume. Likewise, when removing a volume from a Storage Pool, the size of the Storage Pool does not decrease by the size of the volume.
- The system defines capacity using decimal metrics. Using decimal metrics, 1 GB is 1 000 000 000 bytes. Using binary metrics, 1 GB is 1 073 741 824 bytes.

2.3.4 Capacity allocation and thin provisioning

Thin provisioning is a central theme of the virtualized design of the XIV system, because it uncouples the virtual, or apparent, allocation of a resource from the underlying hardware allocation.

The following benefits emerge from the XIV Storage System's implementation of thin provisioning:

- Capacity associated with specific applications or departments can be dynamically increased or decreased per the demand imposed at a given point in time, without necessitating an accurate prediction of future needs. Physical capacity is only committed to the logical volume when the associated applications execute writes, as opposed to when the logical volume is initially allocated.
- Because the total system capacity is architected as a globally available pool, thinly provisioned resources share the same "buffer" of free space, which results in highly efficient aggregate capacity utilization without pockets of inaccessible unused space.

  With the static, inflexible relationship between logical and physical resources commonly imposed by traditional storage subsystems, each application's capacity must be managed and allocated independently. This situation often results in a large percentage of the total system capacity remaining unused, because the capacity is confined within each volume at a highly granular level.
- Capacity acquisition and deployment can be more effectively deferred until actual application and business needs demand additional space, in effect facilitating an on-demand infrastructure.

Actual and logical volume sizes

The physical capacity that is assigned to traditional volumes is equivalent to the logical capacity presented to hosts, which does not have to be the case with the XIV Storage System. For a given logical volume, there are effectively two associated sizes. The physical capacity allocated for the volume is not static, but it increases as host writes fill the volume.

Logical volume size

The *logical volume size* is the size of the logical volume that is observed by the host, as defined upon volume creation or as a result of a resizing command. The storage administrator specifies the volume size in the same manner regardless of whether the Storage Pool will be a thin pool or a regular pool. The volume size is specified in one of two ways, depending on units:

- In terms of GB: The system will allocate the soft volume size as the minimum number of discrete 17 GB increments needed to meet the requested volume size.

- In terms of blocks: The capacity is indicated as a discrete number of 512 byte blocks. The system will still allocate the soft volume size consumed within the Storage Pool as the minimum number of discrete 17 GB increments needed to meet the requested size (specified in 512 byte blocks); however, the size that is reported to hosts is equivalent to the precise number of blocks defined.

Incidentally, the snapshot reserve capacity associated with each Storage Pool is a soft capacity limit, and it is specified by the storage administrator, though it effectively limits the hard capacity consumed collectively by snapshots as well.

Tip: Defining logical volumes in terms of blocks is useful when you must precisely match the size of an existing logical volume residing on another system.

Actual volume size

This reflects the total size of volume areas that were written by hosts. The actual volume size is not controlled directly by the user and depends only on the application behavior. It starts from zero at volume creates or formatting and can reach the logical volume size when the entire volume has been written. Resizing of the volume affects the logical volume size, but does not affect the actual volume size.

The actual volume size reflects the physical space used in the volume, as a result of host writes. It is discretely and dynamically provisioned by the system, not the storage administrator. The discrete additions to actual volume size can be measured in two different ways, by considering the allocated space or the consumed space. The allocated space reflects the physical space used by the volume in 17 GB increments. The consumed space reflects the physical space used by the volume in 1 MB partitions. In both cases, the upper limit of this provisioning is determined by the logical size assigned to the volume.

- Capacity is allocated to volumes by the system in increments of 17 GB due to the underlying logical and physical architecture; there is no smaller degree of granularity than 17 GB. For more details, refer to 2.3.1, “Logical system concepts” on page 16.
- Application write access patterns determine the rate at which the allocated hard volume capacity is consumed and subsequently the rate at which the system allocates additional increments of 17 GB up to the limit defined by the logical volume size. As a result, the storage administrator has no direct control over the actual capacity allocated to the volume by the system at any given point in time.
- During volume creation, or when a volume has been formatted, there is *zero physical capacity assigned to the volume*. As application writes accumulate to new areas of the volume, the physical capacity allocated to the volume will grow in increments of 17 GB and can ultimately reach the full logical volume size.
- Increasing the logical volume size *does not affect* the actual volume size.

Thinly provisioned storage pools

While volumes are effectively thinly provisioned automatically by the system, Storage Pools can be defined by the storage administrator (when using the GUI) as either *regular* or *thinly provisioned.* Note that when using the Extended Command Line Interface (XCLI), there is no specific parameter to indicate thin provisioning for a Storage Pool. You indirectly and implicitly create a Storage Pool as thinly provisioned by specifying a pool soft size greater than its hard size.

With a regular pool, the "host-apparent," capacity is *guaranteed* to be equal to the physical capacity reserved for the pool. The total physical capacity *allocated to* the constituent individual volumes and collective snapshots at any given time within a regular pool will reflect the current usage by hosts, because the capacity is dynamically consumed as required. However, the remaining unallocated space within the pool remains reserved for the pool and cannot be used by other Storage Pools.

In contrast, a thinly provisioned Storage Pool is not fully backed by hard capacity, meaning that the entirety of the logical space within the pool cannot be physically provisioned unless the pool is transformed first into a regular pool. However, benefits can be realized when physical space consumption is less than the logical space assigned, because the amount of logical capacity assigned to the pool that is not covered by physical capacity is available for use by other Storage Pools.

When a Storage Pool is created using thin provisioning, that pool is defined in terms of both a soft size and a hard size independently, as opposed to a regular Storage Pool in which these sizes are by definition equivalent. Hard pool size and soft pool size are defined and used in the following ways.

Hard pool size

Hard pool size is the maximum actual capacity that can be used by all the volumes and snapshots in the pool.

Thin provisioning of the Storage Pool maximizes capacity utilization in the context of a group of volumes, wherein the aggregate "host-apparent," or soft, capacity assigned to all volumes surpasses the underlying physical, or hard, capacity allocated to them. This utilization requires that the aggregate space available to be allocated to hosts within a thinly provisioned Storage Pool must be defined independently of the physical, or hard, space allocated within the system for that pool. Thus, the Storage Pool hard size that is defined by the storage administrator limits the physical capacity that is available collectively to volumes and snapshots within a thinly provisioned Storage Pool, whereas the aggregate space that is assignable to host operating systems is specified by the Storage Pool's soft size.

Regular Storage Pools effectively segregate the hard space reserved for volumes from the hard space consumed by snapshots by limiting the soft space allocated to volumes; however, thinly provisioned Storage Pools permit the totality of the hard space to be consumed by volumes with no guarantee of preserving any hard space for snapshots. Logical volumes take precedence over snapshots and might be allowed to overwrite snapshots if necessary as hard space is consumed. The hard space that is allocated to the Storage Pool that is unused (or in other words, the incremental difference between the aggregate logical and actual *volume* sizes) can, however, be used by snapshots in the same Storage Pool.

Careful management is critical to prevent hard space for both logical volumes and snapshots from being exhausted. Ideally, hard capacity utilization must be maintained under a certain threshold by increasing the pool hard size as needed in advance.

Notes:

- As discussed in "Storage Pool relationships" on page 21, Storage Pools control when and which snapshots are deleted when there is insufficient space assigned within the pool for snapshots.
- The soft snapshot reserve capacity and the hard space allocated to the Storage Pool are consumed only as changes occur to the master volumes or the snapshots themselves, not as snapshots are created.

Soft pool size

Soft pool size is the maximum logical capacity that can be assigned to all the volumes and snapshots in the pool.

Thin provisioning is managed for each Storage Pool independently of all other Storage Pools:

- Regardless of any unused capacity that might reside in other Storage Pools, snapshots within a given Storage Pool will be deleted by the system according to corresponding snapshot pre-set priority if the hard pool size contains insufficient space to create an additional volume or increase the size of an existing volume. (Note that snapshots will actually only be deleted when a write occurs under those conditions, and not when allocating more space).
- As discussed previously, the storage administrator defines both the soft size and the hard size of thinly provisioned Storage Pools and allocates resources to volumes within a given Storage Pool without any limitations imposed by other Storage Pools.

The designation of a Storage Pool as a regular pool or a thinly provisioned pool can be dynamically changed by the storage administrator:

- When a regular pool needs to be converted to a thinly provisioned pool, the soft pool size parameter needs be explicitly set in addition to the hard pool size, which will remain unchanged unless updated.
- When a thinly provisioned pool needs to be converted to a regular pool, the soft pool size is automatically reduced to match the current hard pool size. If the combined allocation of soft capacity for existing volumes in the pool exceeds the pool hard size, the Storage Pool cannot be converted. Of course, this situation can be resolved if individual volumes are selectively resized or deleted to reduce the soft space consumed.

System-level thin provisioning

The definitions of hard size and soft size naturally apply at the subsystem level as well, because by extension, it is necessary to permit the full system to be defined in terms of thin provisioning in order to achieve the full potential benefit previously described: namely, the ability to defer deployment of additional capacity on an as-needed basis.

The XIV Storage System's architecture allows the global system capacity to be defined in terms of both a *hard system size* and a soft system size. When thin provisioning is not activated at the system level, these two sizes are equal to the system's physical capacity.

With thin provisioning, these concepts have the following meanings.

Hard system size

The hard system size represents the physical disk capacity that is available within the XIV Storage System. Obviously, the system's hard capacity is the upper limit of the aggregate hard capacity of all the volumes and snapshots and can only be increased by installing new hardware components in the form of individual modules (and associated disks) or groups of modules.

There are conditions that can *temporarily* reduce the system's hard limit. For further details, refer to 2.4.2, "Rebuild and redistribution" on page 34.

Soft system size

The soft system size is the total, “global,” logical space available for all Storage Pools in the system. When the soft system size exceeds the hard system size, it is possible to logically provision more space than is physically available, thereby allowing the aggregate benefits of thin provisioning of Storage Pools and volumes to be realized at the system level.

The soft system size obviously limits the soft size of all volumes in the system and has the following attributes:

- It is not related to any direct system attribute and can be defined to be larger than the hard system size if thin provisioning is implemented. Note that the storage administrator cannot set the soft system size.

  Note: If the Storage Pools within the system are thinly provisioned, but the soft system size does not exceed the hard system size, the total system hard capacity cannot be filled until all Storage Pools are regularly provisioned. Therefore, we recommend that you define all Storage Pools in a non-thinly provisioned system as regular Storage Pools.

- The soft system size is a purely logical limit; however, you must exercise care when the soft system size is set to a value greater than the maximum potential hard system size. Obviously, it must be possible to upgrade the system’s hard size to be equal to the soft size, so defining an unreasonably high system soft size can result in full capacity depletion. It is for this reason that defining the soft system size is not within the scope of the storage administrator role.

  There are conditions that might *temporarily* reduce the system’s soft limit. For further details, refer to 2.4.2, “Rebuild and redistribution” on page 34.

Thin provisioning conceptual examples

In order to further explain the thin provisioning principles previously discussed, it is helpful to examine the following basic examples, because they incorporate all of the concepts inherent to the XIV Storage System’s implementation of thin provisioning.

System-level thin provisioning conceptual example

Figure 2-5 depicts the incremental allocation of capacity to both a regular Storage Pool and a thinly provisioned Storage Pool within the context of the global system soft and hard sizes. This example assumes that the soft system size has been defined to exceed its hard size. The unallocated capacity shown within the system’s soft and hard space is represented by a discontinuity in order to convey the full scope of both the logical and physical view of the system’s capacity. Each increment in the diagram represents 17 GB of soft or hard capacity.

When a regular Storage Pool is defined, only one capacity is specified, and this amount is allocated to the Storage Pool from *both the hard and soft global capacity within the system*.

When a thinly provisioned Storage Pool is defined, both the soft and hard capacity limits for the Storage Pool must be specified, and these amounts are deducted from the system’s global available soft and hard capacity, respectively.

In the next example we focus on the regular Storage Pool introduced in Figure 2-5.

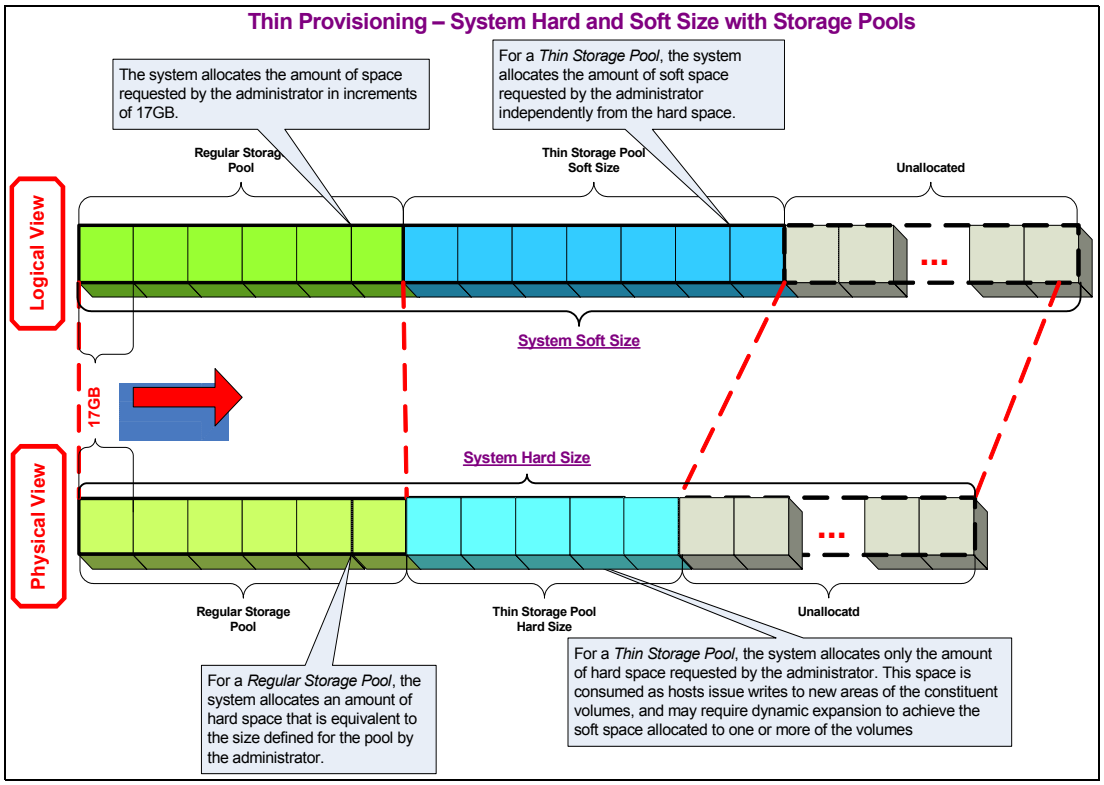


Figure 2-5 Thin provisioning at the system level

Regular Storage Pool conceptual example

Next, Figure 2-6 represents a focused view of the regular Storage Pool that is shown in Figure 2-5 and depicts the division of both soft and hard capacity among volumes within the pool. Note that the regular pool is the same size (102 GB) in both diagrams.

First, consider Volume 1. Although Volume 1 is defined as 19,737,900 blocks (10 GB), the soft capacity allocated will nevertheless be comprised of the minimum number of 17 GB increments needed to meet or exceed the requested size in blocks, which is in this case only a single 17 GB increment of capacity. The host will, however, see exactly 19,737,900 blocks. When Volume 1 is created, the *system does not initially allocate any hard capacity.* At the moment that a host writes to Volume 1, even if it is just to initialize the volume, the system will allocate 17 GB of hard capacity. The hard capacity allocation of 17 GB for Volume 1 is illustrated in Figure 2-6, although clearly this allocation will never be fully utilized as long as the host-defined capacity remains only 10 GB.

Unlike Volume 1, Volume 2 has been defined in terms of gigabytes and has a soft capacity allocation of 34 GB, which is the amount that is reported to any hosts that are mapped to the volume. In addition, the hard capacity consumed by host writes has not yet exceeded the 17 GB threshold, and hence, the system has thus far *only allocated one increment of 17 GB hard capacity*. However, because the hard capacity and the soft capacity allocated to a regular Storage Pool are equal by definition, the remaining 17 GB of soft capacity assigned to Volume 2 is effectively preserved and will remain available within the pool's hard space until it is needed by Volume 2. In other words, because the pool's soft capacity does not exceed its hard capacity, there is no way to allocate soft capacity to effectively "overcommit" the available hard capacity.

The final reserved space within the regular Storage Pool shown in Figure 2-6 is dedicated for the snapshot usage. The diagram illustrates that the specified snapshot reserve capacity of

34 GB is effectively deducted from both the hard and soft space defined for the regular Storage Pool, thus guaranteeing that this space will be available for consumption collectively by the snapshots associated with the pool. Although snapshots consume space granularly at the partition level, as discussed in “Storage Pool relationships” on page 21, the snapshot reserve capacity is still defined in increments of 17 GB.

The remaining 17 GB within the regular Storage Pool have not been allocated to either volumes or snapshots. Note that all soft capacity remaining in the pool is “backed” by hard capacity; the remaining unused soft capacity will always be less than or equal to the remaining unused hard capacity.

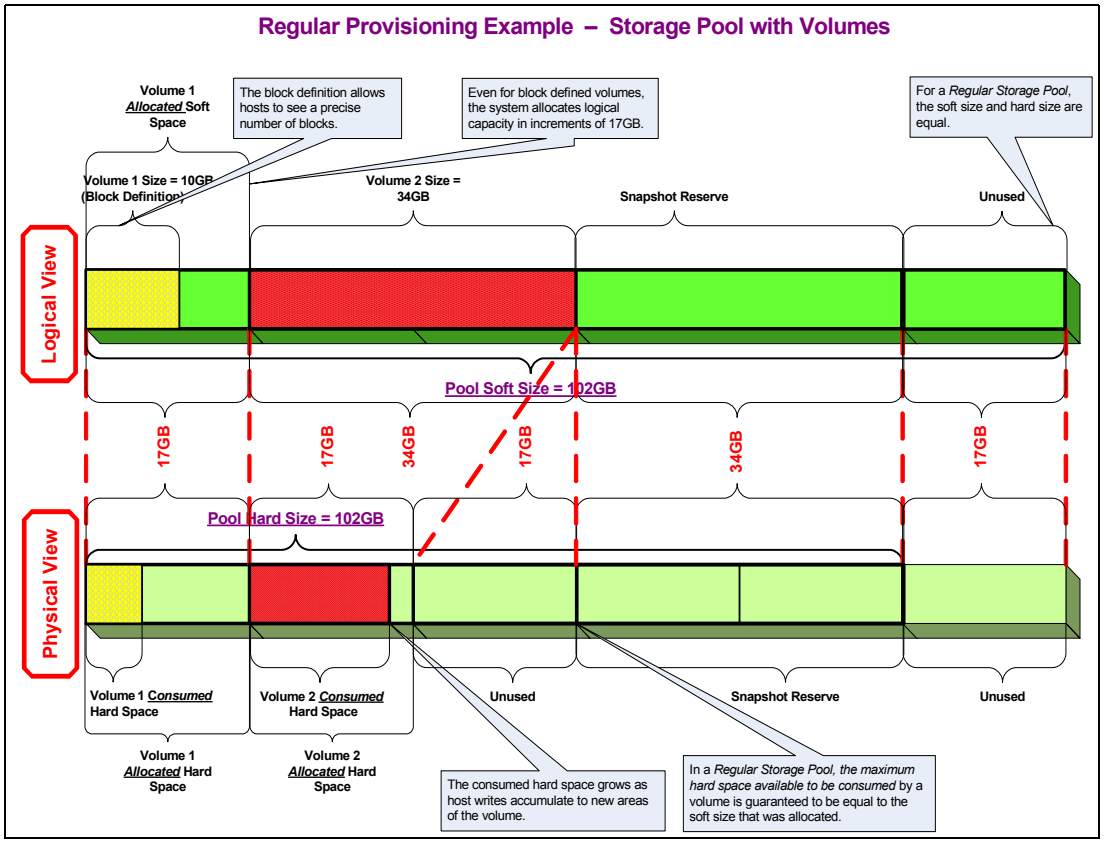


Figure 2-6 Volumes and snapshot reserve space within a regular Storage Pool

Thinly provisioned Storage Pool conceptual example

The thinly provisioned Storage Pool that was introduced in Figure 2-5 on page 28 is explored in detail in Figure 2-7. Note that the hard capacity and the soft capacity allocated to this pool are the same in both diagrams: 136 GB of soft capacity and 85 GB of hard capacity are allocated. Because the available soft capacity exceeds the available hard capacity by 51 GB, it is possible to thinly provision the volumes collectively by *up to* 66.7%, assuming that the snapshots are preserved and the remaining capacity within the pool is allocated to volumes.

Consider Volume 3 in Figure 2-7. The size of the volume is defined as 34 GB; however, less than 17 GB has been consumed by host writes, so only 17 GB of hard capacity have been allocated by the system. In comparison, Volume 4 is defined as 51 GB, but Volume 4 has consumed between 17 GB and 34 GB of hard capacity and therefore has been allocated 34 GB of hard space by the system. It is possible for either of these two volumes to require up to an additional 17 GB of hard capacity to become fully provisioned, and therefore, at least 34 GB of additional hard capacity must be allocated to this pool in anticipation of this requirement.

Finally, consider the 34 GB of snapshot reserve space depicted in Figure 2-7. If a new volume is defined in the unused 17 GB of soft space in the pool, or if either Volume 3 or Volume 4 requires additional capacity, the system will sacrifice the snapshot reserve space in order to give priority to the volume requirements. Normally, this scenario does not occur, because additional hard space must be allocated to the Storage Pool as the hard capacity utilization crosses certain thresholds.

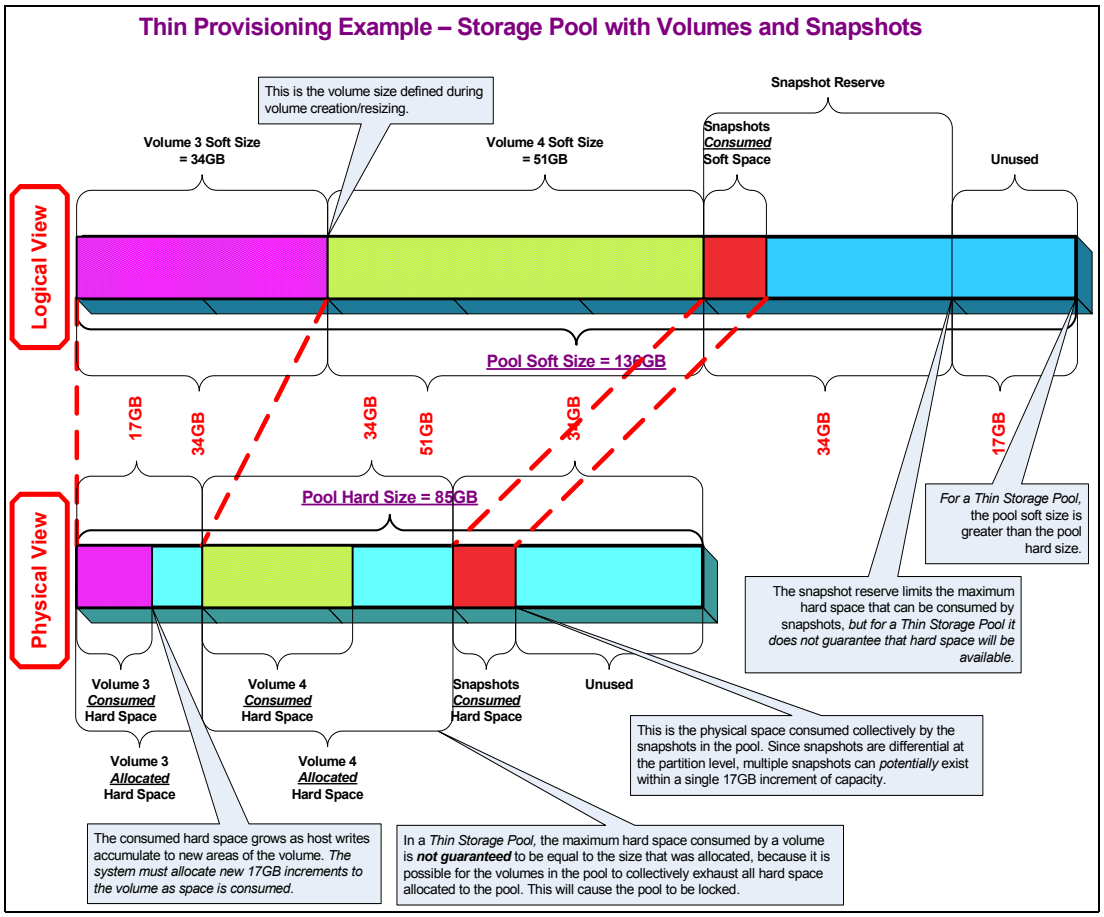


Figure 2-7 Volumes and snapshot reserve space within a thinly provisioned Storage Pool

Depletion of hard capacity

Using thin provisioning creates the inherent danger of exhausting the available physical capacity. If the soft system size exceeds the hard system size, the potential exists for applications to fully deplete the available physical capacity.

Important: Upgrading the system beyond the full 15 modules in a single frame is currently not supported.

Snapshot deletion

As mentioned previously, snapshots in regular Storage Pools can be automatically deleted by the system in order to provide space for newer snapshots, or in the case of thinly provisioned pools, to permit more physical space for volumes.

Volume locking

If more hard capacity is still required after all the snapshots in a thinly provisioned Storage Pool have been deleted, *all the volumes in the Storage Pool are locked, thereby preventing*

any additional consumption of hard capacity. There are two possible behaviors for a locked volume: read only (the default behavior) or no I/O at all.

Important: Volume locking prevents writes to all volumes in the Storage Pool.

It is very important to note that thin provisioning implementation in the XIV Storage System manages space allocation within each Storage Pool, so that hard capacity depletion in one Storage Pool will never affect the hard capacity available to another Storage Pool. There are both advantages and disadvantages:

- Because Storage Pools are independent, thin provisioning volume locking on one Storage Pool never cascades into another Storage Pool.
- Hard capacity cannot be reused across Storage Pools, even if a certain Storage Pool has free hard capacity available, which can lead to a situation where volumes are locked due to the depletion of hard capacity in one Storage Pool, while there is available capacity in another Storage Pool. Of course, it is still possible for the storage administrator to intervene in order to redistribute hard capacity.

2.4 Reliability, availability, and serviceability

The XIV Storage System's unique modular design and logical topology fundamentally differentiate it from traditional monolithic systems, and this architectural divergence extends to the exceptional reliability, availability, and serviceability aspects of the system. In addition, the XIV Storage System incorporates autonomic, proactive monitoring and self-healing features that are capable of not only transparently and automatically restoring the system to full redundancy within minutes of a hardware failure, but also taking preventive measures to preserve data redundancy even before a component malfunction actually occurs.

For further reading about the XIV Storage System's parallel modular architecture, refer to 2.2, "Parallelism" on page 12.

2.4.1 Resilient architecture

As with any enterprise class system, redundancy pervades every aspect of the XIV Storage System, including the hardware, internal operating environment, and the data itself. However, the design elements, including the distribution of volumes across the whole of the system, in combination with the loosely coupled relationship between the underlying hardware and software elements, empower the XIV Storage System to realize unprecedented resiliency. The resiliency of the architecture encompasses not only high availability, but also excellent maintainability, serviceability, and performance under non-ideal conditions resulting from planned or unplanned changes to the internal hardware infrastructure, such as the loss of a module.

Availability

The XIV Storage System maximizes operational availability and minimizes the degradation of performance associated with nondisruptive planned and unplanned events, while providing for the capability to preserve the data to the fullest extent possible in the event of a disaster.

High reliability

The XIV Storage System not only withstands individual component failures by quickly and efficiently reinstating full data redundancy, but also automatically monitors and phases out individual components before data redundancy is compromised. We discuss this topic in

detail in “Proactive phase-out and self-healing mechanisms” on page 40. The collective high reliability provisions incorporated within the system constitute multiple layers of protection from unplanned outages and minimize the possibility of related service actions.

Maintenance freedom

While the potential for unplanned outages and associated corrective service actions are mitigated by the reliability attributes inherent to the system design, the XIV Storage System’s autonomic features also minimize the need for storage administrators to conduct non-preventative maintenance activities that are purely reactive in nature, by adapting to potential issues before they are manifested as a component failure. The continually restored redundancy in conjunction with the self-healing attributes of the system effectively enable maintenance activities to be decoupled from the instigating event (such as a component failure or malfunction) and safely carried out according to a predefined schedule. In addition to the system’s diagnostic monitoring and autonomic maintenance, the proactive and systematic, rather than purely reactive, approach to maintenance is augmented, because the entirety of the logical topology is continually preserved, optimized, and balanced according to the physical state of the system.

The modular system design also expedites the installation of any replacement or upgraded components, while the automatic, transparent data redistribution across all resources eliminates the downtime, even in the context of individual volumes, associated with these critical activities.

High availability

The rapid restoration of redundant data across all available drives and modules in the system during hardware failures, and the equilibrium resulting from the automatic redistribution of data across all newly installed hardware, are fundamental characteristics of the XIV Storage System architecture that minimize exposure to cascading failures and the associated loss of access to data.

Consistent performance

The XIV Storage System is capable of adapting to the loss of an individual drive or module efficiently and with relatively minor impact compared to monolithic architectures. While traditional monolithic systems employ an N+1 hardware redundancy scheme, the XIV Storage System harnesses the resiliency of the grid topology, not only in terms of the ability to sustain a component failure, but also by maximizing consistency and transparency from the perspective of attached hosts. The potential impact of a component failure is vastly reduced, because each module in the system is responsible for a relatively small percentage of the system’s operation. Simply put, a controller failure in a typical N+1 system likely results in a dramatic (up to 50%) reduction of available cache, processing power, and internal bandwidth, whereas the loss of a module in the XIV Storage System translates to only 1/15th of the system resources and does not compromise performance nearly as much as the same failure with a typical architecture.

Additionally, the XIV Storage System incorporates innovative provisions to mitigate isolated disk-level performance anomalies through *redundancy-supported reaction*, which is discussed in “Redundancy-supported reaction” on page 41, and flexible handling of dirty data, which is discussed in “Flexible handling of dirty data” on page 41.

Disaster recovery

Enterprise class environments must account for the possibility of the loss of both the system and all of the data as a result of a disaster. The XIV Storage System includes the provision for Remote Mirror functionality as a fundamental component of the overall disaster recovery strategy.

Write path redundancy

Data arriving from the hosts is temporarily placed in two separate caches before it is permanently written to disk drives located in separate modules. This design guarantees that the data is always protected against possible failure of individual modules, even before the data has been written to the disk drives.

Figure 2-8 on page 33 illustrates the path taken by a write request as it travels through the system. The diagram is intended to be viewed as a conceptual topology, so do not interpret the specific numbers of connections and so forth as literal depictions. Also, for purposes of this discussion, the Interface Modules are depicted on a separate level from the Data Modules. However, in reality the Interface Modules also function as Data Modules. The following numbers correspond to the numbers in Figure 2-8:

1. A host sends a write request to the system. Any of the Interface Modules that are connected to the host can service the request, because the modules work in an active-active capacity. Note that the XIV Storage System does not load balance the requests itself. Load balancing must be implemented by storage administrators to equally distribute the host requests among all Interface Modules.
2. The Interface Module uses the system configuration information to determine the location of the primary module that houses the referenced data, which can be either an Interface Module, including the Interface Module that received the write request, or a Data Module. The data is written only to the local cache of the primary module.
3. The primary module uses the system configuration information to determine the location of the secondary module that houses the copy of the referenced data. Again, this module can be either an Interface Module or a Data Module, but it will not be the same as the primary module. The data is redundantly written to the local cache of the secondary module.

After the data is written to cache in both the primary and secondary modules, the host receives an acknowledgement that the I/O is complete, which occurs independently of copies of either cached, or dirty, data being destaged to physical disk.

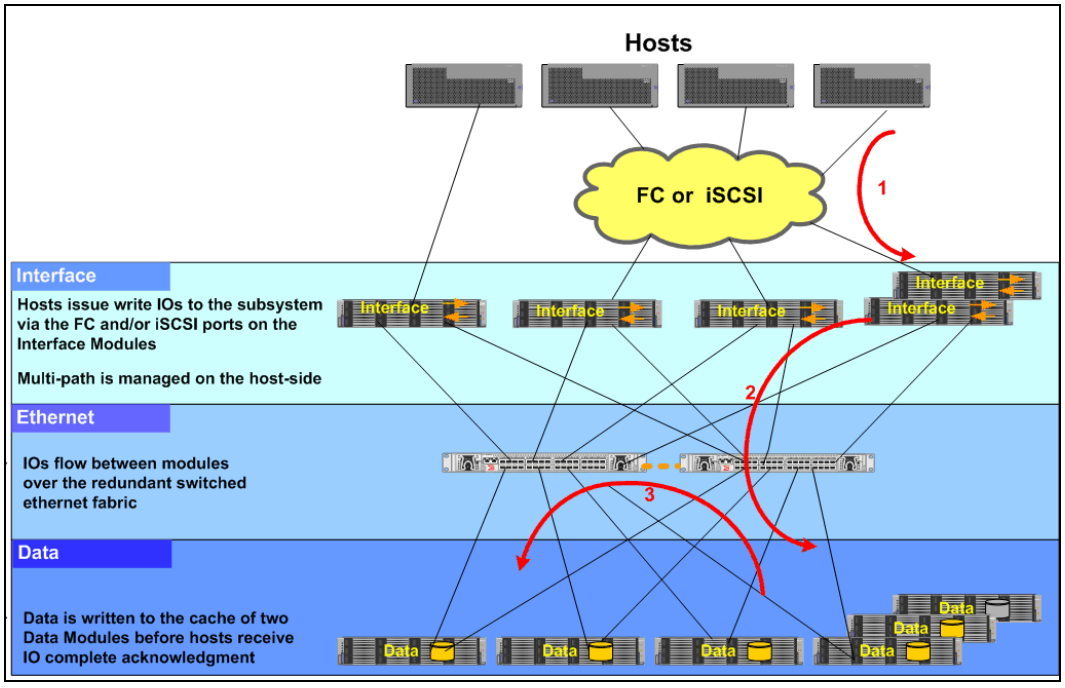


Figure 2-8 Write path

System quiesce and graceful shutdown

When an event occurs that compromises both sources of power to the XIV Storage System's redundant uninterruptible power supplies, the system executes the graceful shutdown sequence. Full battery power is guaranteed during this event, because the system monitors available battery charge at all times and takes proactive measures to prevent the possibility of conducting write operations when battery conditions are non-optimal.

Due to the XIV Storage System's grid topology, a system quiesce event essentially entails the graceful shutdown of all modules within the system. Each module can be thought of as an independent entity that is responsible for managing the destaging of "dirty" data, that is, written data that has not yet been destaged to physical disk. The dirty data within each module consists of equal parts primary and secondary copies of data, *but will never contain both primary and secondary copies of the same data.*

Write cache protection

Each module in the XIV Storage System contains an local, independent space reserved for caching operations within its system memory.

Each module contains 8 GB of high speed volatile *memory* (a total of 120 GB), from which 5.5 GB (and 82.5 GB overall) is dedicated for caching data.

Note: The system does *not* contain non-volatile memory space that is reserved for write operations. However, the close proximity of the cache and the drives, in conjunction with the enforcement of an upper limit for dirty, or non-destaged, data on a per-drive basis, ensures that the full destage will occur while operating under battery power.

Graceful shutdown sequence

The system executes the graceful shutdown sequence under either of these conditions:

- The battery charge remaining in two or more universal power supplies is below a certain threshold, which is conservatively predetermined in order to provide adequate time for the system to fully destage all dirty data from cache.
- The system detects the loss of external power for more than 30 seconds.

Power on sequence

Note: If the battery charge is inadequate, the system will remain fully locked until the battery charge has exceeded the necessary threshold to safely resume I/O activity.

Upon startup, the system will verify that the battery charge levels in all uninterruptible power supplies exceed the threshold necessary to guarantee that a graceful shutdown can occur. If the charge level is inadequate, the system will halt the startup process until the charge level has exceeded the minimum required threshold.

2.4.2 Rebuild and redistribution

As discussed in "Data distribution algorithms" on page 14, the XIV Storage System dynamically maintains the pseudo-random distribution of data across all modules and disks while ensuring that two copies of data exist at all times *when the system reports Full Redundancy.* Obviously, when there is a change to the hardware infrastructure as a result of a failed component, data must be restored to redundancy and distributed, or when a component is added, or *phased-in,* a new data distribution must accommodate the change.

Goal distribution

The process of achieving a new goal distribution while simultaneously restoring data redundancy due to the loss of a disk or module is known as a *rebuild*. Because a rebuild occurs as a result of a component failure that compromises full data redundancy, there is a period during which the *non-redundant data* is both restored to full redundancy and homogeneously redistributed over the remaining disks.

The process of achieving a new goal distribution (only occurring when redundancy exists) is known as a *redistribution*, during which all data in the system (including both primary and secondary copies) is redistributed, when it is a result of the following events:

- The replacement of a failed disk or module following a rebuild, also known as a “phase-in”
- When one or more modules are added to the system, known as a “scale out” upgrade

Following any of these occurrences, the XIV Storage System immediately initiates the following sequence of events:

1. The XIV Storage System distribution algorithms calculate which partitions must be relocated and copied based on the distribution table that is described in 2.2.2, “Software parallelism” on page 13. The resultant distribution table is known as the *goal distribution.*
2. The Data Modules and Interface Modules begin concurrently redistributing and copying (in the case of a rebuild) the partitions according to the goal distribution:
   - This process occurs in a parallel, any-to-any fashion concurrently among all modules and drives in the background, with complete host transparency.
   - The priority associated with achieving the new goal distribution is internally determined by the system. The priority cannot be adjusted by the storage administrator:
     - Rebuilds have the highest priority; however, the transactional load is homogeneously distributed over all the remaining disks in the system resulting in a very low density of system-generated transactions.
     - Phase-outs (caused by the XIV technician removing and replacing a failed module) have lower priority than rebuilds, because at least two copies of all data exist at all times during the phase-out.
     - Redistributions have the lowest priority, because there is neither a lack of data redundancy nor has the system detected the potential for an impending failure.
3. The system reports Full Redundancy after the goal distribution has been met.

   Following the completion of goal distribution resulting from a rebuild or phase-out, a subsequent redistribution must occur when the system hardware is fully restored through a phase-in.

Note: The goal distribution is transparent to storage administrators and cannot be changed. In addition, the goal distribution has many determinants depending on the precise state of the system.

Important: Never perform a phase-in to replace a failed disk or module until after the rebuild process has completed. These operations must be performed by the IBM XIV technician anyway.

Preserving data redundancy

Whereas conventional storage systems maintain a static relationship between RAID arrays and logical volumes by preserving data redundancy only across a subset of disks that are defined in the context of a particular RAID array, the XIV Storage System dynamically and fluidly restores redundancy and equilibrium across all disks and modules in the system during the rebuild and phase-out operations. Refer to “Logical volume layout on physical disks” on page 18 for a detailed discussion of the low-level virtualization of logical volumes within the XIV Storage System. The proactive phase-out of non-optimal hardware through autonomic monitoring and the modules’ cognizance of the virtualization between the logical volumes and physical disks yield unprecedented efficiency, transparency, and reliability of data preservation actions, encompassing both rebuilds and phase-outs:

- The rebuild of data is many times faster than conventional RAID array rebuilds and can complete in a short period of time for a fully provisioned system, because the redistribution workload spans all drives in the system resulting in very low transactional density:
  - Statistically, the chance of exposure to data loss or a cascading hardware failure, which occurs when corrective actions in response to the original failure result in a subsequent failure, is minimized due to both the brevity of the rebuild action and the low density of access on any given disk.

    Rebuilding conventional RAID arrays can take many hours to complete, depending on the type of the array, the number of drives, and the ongoing host-generated transactions to the array.
  - The rebuild process can complete 25% to 50% more quickly for systems that are not fully provisioned, which equates to a rebuild completion in as little as 15 minutes.
- The system relocates only real data, as opposed to rebuilding the entire array, which consists of complete disk images that often include unused space, vastly reducing the potential number of transactions that must occur.

  Conventional RAID array rebuilds can place many times the normal transactional load on the disks and substantially reduce effective host performance.
- The number of drives participating in the rebuild is about 20 times greater than in most average-sized conventional RAID arrays, and by comparison, the array rebuild workload is greatly dissipated, greatly reducing the relative impact on host performance.
- Whereas standard dedicated spare disks utilized during a conventional RAID array rebuild might not be globally accessible to all arrays in the system, the XIV Storage System maintains universally accessible reserve space on all disks in the system, as discussed in “Global spare capacity” on page 20.
- Because the system maintains access density equilibrium, hot-spots are statistically eliminated, which reduces the chances of isolated workload-induced failures.
- The system-wide goal distribution alleviates localized drive stress and associated additional heat generation, which can significantly increase the probability of a double drive failure during the rebuild of a RAID array in conventional subsystems.
- Modules intelligently send information to each other directly. There is no need for a centralized supervising controller to read information from one disk module and write to another disk module.
- All disks are monitored for errors, poor performance, or other signs that might indicate that a full or partial failure is impending.

  Dedicated spare disks in conventional RAID arrays are inactive, and therefore, unproven and potentially unmonitored, increasing the possibility for a second failure during an array rebuild.

Rebuild examples

When the full redundancy of data is compromised due to a module failure, as depicted in Figure 2-9, the system immediately identifies the non-redundant partitions and begins the rebuild process. Because none of the disks within a given module contain the secondary copies of data residing on any of the disks in the module, the secondary copies are read from the remaining modules in the system. Therefore, during a rebuild resulting from a module failure, there will be concurrently 168 disks (180 disks in the system minus 12 disks in a module) reading, and 168 disks writing, as is conceptually illustrated in Figure 2-9.

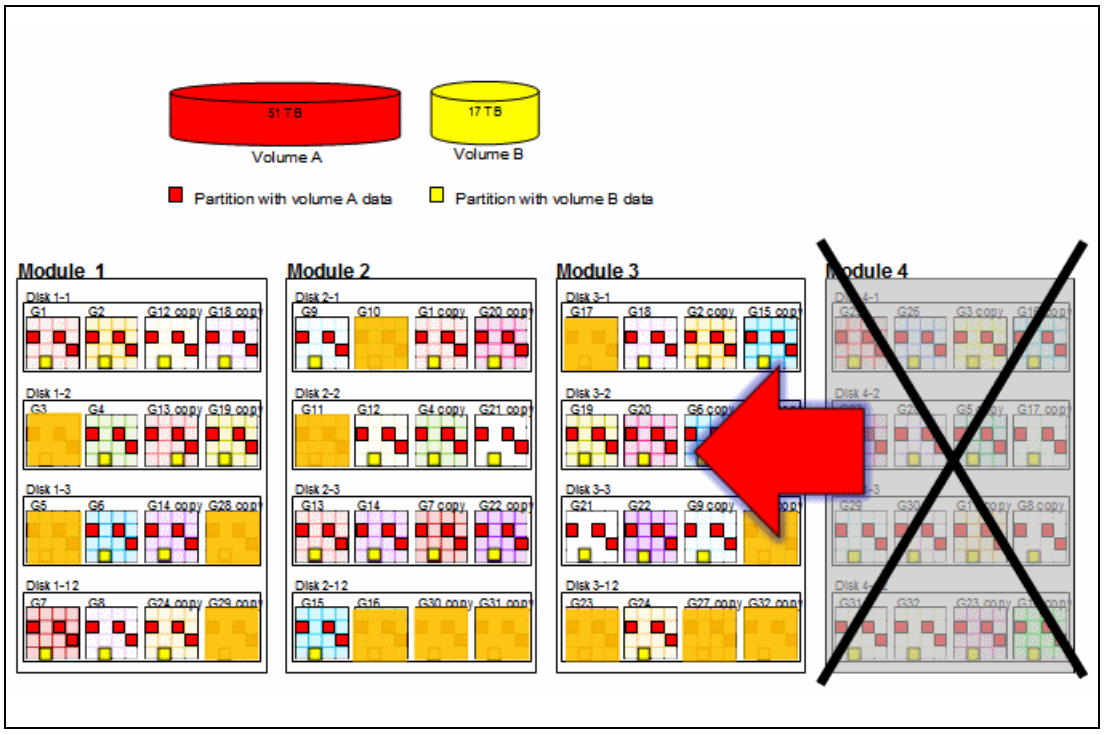


Figure 2-9 Non-redundant group of partitions following module failure

Figure 2-10 depicts a denser population of redundant partitions for both volumes A and B, thus representing the completion of a new goal distribution, as compared to Figure 2-9, which contains the same number of redundant partitions for both volumes distributed less densely over the original number of modules and drives.

Finally, consider the case of a single disk failure occurring in an otherwise healthy system (no existing phased-out or failed hardware). During the subsequent rebuild, there will be only 168 disks reading, because there is no non-redundant data residing on the other disks within the same module as the failed disk. Concurrently, there will be 179 disks writing in order to preserve full data distribution.

Note: Figure 2-9 and Figure 2-10 conceptually illustrate the rebuild process resulting from a failed module. *The diagrams are not intended to depict in any way the specific placement of partitions within a real system, nor do they literally depict the number of modules in a real system.*

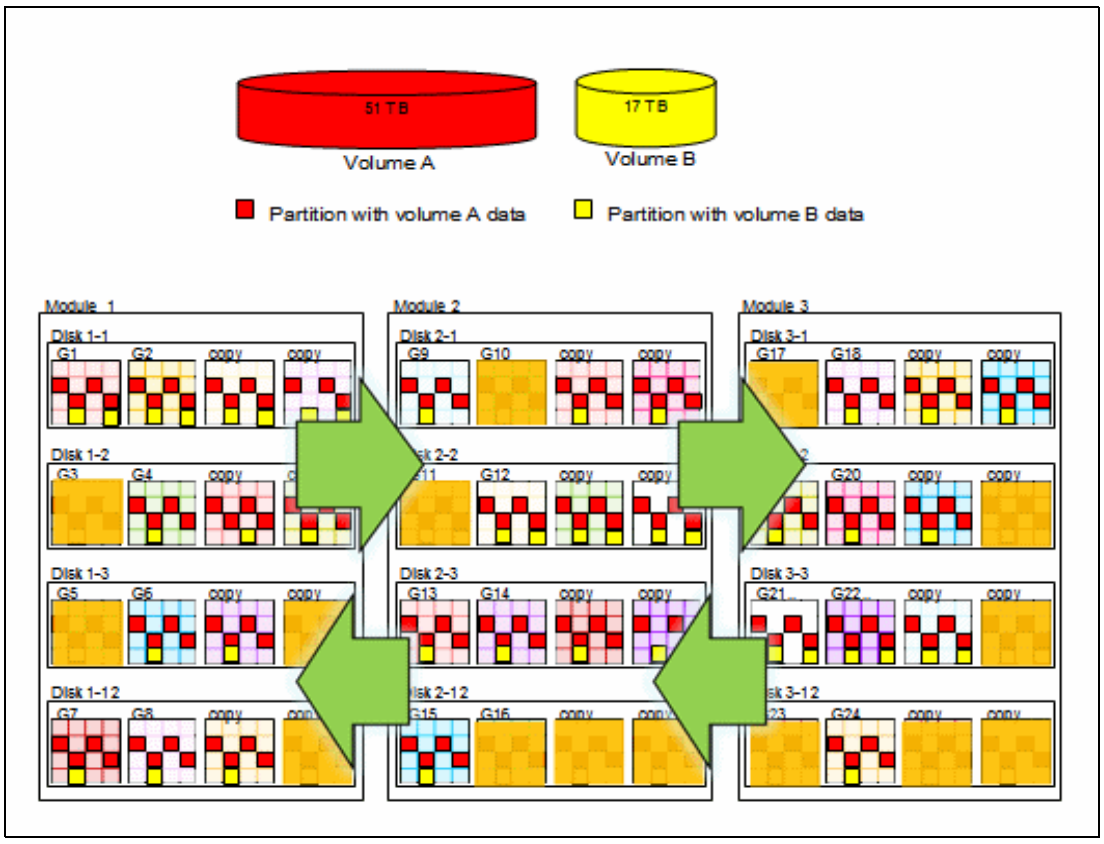


Figure 2-10 Performing a new goal distribution following module failure

Transient soft and hard system size

The capacity allocation that is consumed for purposes of either restoring non-redundant data during a rebuild, or creating a tertiary copy during a phase-out, will be sourced based on availability, with the following precedence:

1. *Unallocated system hard capacity*: The system might consume hard capacity that was not assigned to any Storage Pools at the time of the failure.
2. *Unallocated Storage Pool hard capacity: The hard capacity of Storage Pools that is not assigned to any existing volumes or consumed by snapshots, as measured before the failure, is unallocated hard capacity. For details about the topic of Storage Pool sizes, refer to "Thinly provisioned storage pools" on page 24.* Do not confuse this unallocated Storage Pool hard capacity with "unconsumed" capacity, which is unwritten hard space allocated to volumes.
3. *Reserve spare capacity*: As discussed previously, the system reserves enough capacity to sustain the consecutive, non-concurrent failure of three drives and an entire module before replacement hardware must be phased in to ensure that data redundancy can be restored during subsequent hardware failures.

In the event that sufficient unallocated hard capacity is available, the system will withhold allocating reserve spare space to complete the rebuild or phase-out process in order to provide additional protection. As a result, it is possible for the system to report a maximum soft size that is temporarily less than the allocated soft capacity. The soft and hard system sizes will not revert to the original values until a replacement disk or module is phased-in, and the resultant redistribution completes.

Important: While it is possible to resize or create volumes, snapshots, or Storage Pools while a rebuild is underway, we strongly discourage these activities until the system has completed the rebuild process and restored full data redundancy.

Redistribution

The XIV Storage System homogeneously redistributes all data across all disks whenever new disks or modules are introduced or phased in to the system. This redistribution process is not equivalent to the “striping volumes on all disks” employed in traditional systems:

- Both conventional RAID striping, as well as the data distribution, fully incorporate all spindles when the hardware configuration remains static; however, when new capacity is added and new volumes are allocated, ordinary RAID striping algorithms do not intelligently redistribute data to preserve equilibrium for all volumes through the pseudo-random distribution of data, which is described in 2.2.2, “Software parallelism” on page 13.
- Thus, the XIV Storage System employs dynamic volume-level virtualization, obviating the need for ongoing manual volume layout planning.

The redistribution process is triggered by the “phase-in” of a new drive or module and differs from a rebuild or phase-out in that:

- The system does not need to create secondary copies of data to reinstate or preserve full data redundancy.
- The distribution density, or the concentration of data on each physical disk, decreases instead of increasing.
- The redistribution of data performs differently, because the concentration of write activity on the new hardware resource is the bottleneck:
  - When a replacement module is phased-in, there will be concurrently 168 disks reading and 12 disks writing, and thus the time to completion is limited by the throughput of the replacement module. Also, the read access density on the existing disks will be extremely low, guaranteeing extremely low impact on host performance during the process.
  - When a replacement disk is phased-in, there will be concurrently 179 disks reading and only one disk writing. In this case, the replacement drive obviously limits the achievable throughput of the redistribution. Again, the impact on host transactions is extremely small, or insignificant.

2.4.3 Minimized exposure

This section describes other features that contribute to the XIV Storage System reliability and availability.

Disaster recovery

All high availability SAN implementations must account for the contingency of data recovery and business continuance following a disaster, as defined by the organization’s recovery point and recovery time objectives. The provision within the XIV Storage System to efficiently and flexibly create nearly unlimited snapshots, coupled with the ability to define Consistency Groups of logical volumes, constitutes integral elements of the data preservation strategy. In addition, the XIV Storage System’s synchronous data mirroring functionality facilitates excellent potential recovery point and recovery time objectives as a central element of the full disaster recovery plan.

Proactive phase-out and self-healing mechanisms

The XIV Storage System can seamlessly restore data redundancy with minimal data migration and overhead. A further enhancement to the level of reliability standards attained by the XIV Storage System entails self-diagnosis and early detection mechanisms that autonomically phase out components before the probability of a failure increases beyond a certain point. In real systems, the failure rate is not constant with time, but rather increases with service life and duty cycle. By actively gathering component statistics to monitor this trend, the system ensures that components will not operate under conditions beyond an acceptable threshold of reliability and performance. Thus, the XIV Storage System's self-healing mechanisms dramatically increase the already exceptional level of availability of the system, because they virtually preclude the possibility of data redundancy from ever being compromised along with the associated danger, however unlikely, of subsequent failures during the rebuild process.

The autonomic attributes of the XIV Storage System cumulatively impart an enormous benefit to not only the reliability of the system, but also the overall availability, by augmenting the maintainability and serviceability aspects of the system. Both the monetary and time demands associated with maintenance activities, or in other words, the total cost of ownership (TCO), are effectively minimized by reducing reactive service actions and enhancing the potential scope of proactive maintenance policies.

Disk scrubbing

The XIV Storage System maintains a series of scrubbing algorithms that run as background processes concurrently and independently scanning multiple media locations within the system in order to maintain the integrity of the redundantly stored data. This continuous checking enables the early detection of possible data corruption, alerting the system to take corrective action to restore the data integrity before errors can manifest themselves from the host perspective. Thus, redundancy is not only implemented as part of the basic architecture of the system, but it is also continually monitored and restored as required. In summary, the data scrubbing process has the following attributes:

- Verifies the integrity and redundancy of stored data
- Enables early detection of errors and early recovery of redundancy
- Runs as a set of background processes on all disks in parallel
- Checks whether data can be read from partitions and verifies data integrity by employing checksums
- Examines a partition approximately every two seconds

Enhanced monitoring and disk diagnostics

The XIV Storage System continuously monitors the performance level and reliability standards of each disk drive within the system, using an enhanced implementation of Self-Monitoring, Analysis and Reporting Technology (SMART) tools. As typically implemented in the storage industry, SMART tools simply indicate whether certain thresholds have been exceeded, thereby alerting that a disk is at risk for failure and thus needs to be replaced.

However, as implemented in XIV Storage System, the SMART diagnostic tools, coupled with intelligent analysis and low tolerance thresholds, provide an even greater level of refinement of the disk behavior diagnostics and the performance and reliability driven reaction. For instance, the XIV Storage System measures the specific values of parameters including, but not limited to:

- *Reallocated sector count*: If the disk encounters a read or write verification error, it designates the affected sector as "reallocated" and relocates the data to a reserved area of spare space on the disk. Note that this spare space is a parameter of the drive itself and is *not* related in any way to the system reserve spare capacity that is described in "Global spare capacity" on page 20. The XIV Storage System initiates phase-out at a much lower count than the manufacturer recommends.
- *Disk temperature*: The disk temperature is a critical factor that contributes to premature drive failure and is constantly monitored by the system.
- *Raw read error*: The raw read error count provides an indication of the condition of the magnetic surface of the disk platters and is carefully monitored by the system to ensure the integrity of the magnetic media itself.
- *Spin-up time*: The spin-up time is a measure of the average time that is required for a spindle to accelerate from zero to 7 200 rpm. The XIV Storage System recognizes abnormal spin-up time as a potential indicator of an impending mechanical failure.

Likewise, for additional early warning signs, the XIV Storage System continually monitors other aspects of disk-initiated behavior, such as spontaneous reset or unusually long latencies. The system intelligently analyzes this information in order to reach crucial decisions concerning disk deactivation and phase-out. The parameters involved in these decisions allow for a very sensitive analysis of the disk health and performance.

Redundancy-supported reaction

The XIV Storage System incorporates *redundancy-supported reaction*, which is the provision to exploit the distributed redundant data scheme by intelligently redirecting reads to the secondary copies of data, thereby extending the system's tolerance of above average disk service time when accessing primary data locations. The system will reinstate reads from the primary data copy when the transient degradation of the disk service time has subsided. Of course, a redundancy-supported reaction itself might be triggered by an underlying potential disk error that will ultimately be managed autonomically by the system according to the severity of the exposure, as determined by ongoing disk monitoring.

Flexible handling of dirty data

In a similar manner to the redundancy-supported reaction for read activity, the XIV Storage System can also make convenient use of its redundant architecture in order to consistently maintain write performance. Because intensive write activity directed to any given volume is distributed across all modules and drives in the system, and the cache is independently managed within each module, the system is able to tolerate sustained write activity to an under-performing drive by effectively maintaining a considerable amount of "dirty," or unwritten, data in cache, thus potentially circumventing any performance degradation resulting from the transient, anomalous service time of a given disk drive.

Non-disruptive code load

Non-disruptive code load (NDCL) enables upgrades to the IBM XIV Storage System software from a current version (starting with Version 10.1) to a later version without disrupting the application service.

The code upgrade is run on all modules in parallel and the process is fast enough to minimize impact on hosts applications.

No data migration or rebuild process is allowed during the upgrade. Mirroring, if any, will be suspended during the upgrade and automatically reactivated upon completion.

Storage management operations are also not allowed during the upgrade, although the status of the system and upgrade progress can be queried. It is also possible to cancel the upgrade process up to a point of no return.

Note that the NDCL does not apply to specific components firmware upgrades (for instance, module BIOS, HBA firmware). Those require a phase in / phase out process of the impacted modules.

3

XIV physical architecture, components, and planning

This chapter describes the hardware architecture of the XIV Storage System. We present the physical components that make up the XIV Storage System, such as the system rack, Interface Modules, Data Modules, Management Module, disks, switches, and power distribution devices.

Included as well is an overview of the planning aspects required before and after deployment of an XIV Storage System.

3.1 IBM XIV Storage System models 2810-A14 and 2812-A14

The XIV Storage System seen in Figure 3-1 is designed to be a scalable enterprise storage system based upon a grid array of hardware components. The architecture offers the highest performance through maximized utilization of all disks, true distributed cache implementation, coupled with more effective bandwidth. It also offers superior reliability through distributed architecture, redundant components, self-monitoring, and self-healing.

Figure 3-1 IBM XIV Storage System front and rear views

Note: Figure 3-1does not depict the new rack door now available as shown in Figure 3-4 on page 47,

Hardware characteristics

The XIV Storage System family consists of two machine types, the 2810-A14 and the 2812-A14. The 2812 machine type comes standard with a 3 year manufacturer warranty. The 2810 machine type is delivered with a 1 year standard warranty. Most of the hardware features are the same for both machine types; The major differences are listed in Table 3-1.

Table 3-1 Machine type comparisons

Machine type	2810-A14	2812-A14
Warranty	1 year	3 years
CPUs per Interface Module	1 or 2	2
CPUs per Data Module	1	1

Figure 3-2 summarizes the main hardware characteristics of the IBM XIV Storage System 2810-A14 and 2812-A14.

All XIV hardware components come pre-installed in a standard 19” data center class rack. At the bottom of the rack, an Uninterruptible Power Supply (UPS) module complex, which is made up of three redundant UPS units, is installed and provides power to the various system components.

Fully populated configurations

A fully populated rack contains 9 Data Modules and 6 Interface Modules for a total of 15 modules. Each module is equipped with the following connectivity adapters:

- USB ports
- Serial ports
- Ethernet adapters
- Fibre Channel adapters (Interface Modules only)

Each module also contains twelve 1TB Serial Advanced Technology Attachment (SATA) disk drives. This design translates into a total usable capacity of 79 TB (180 TB raw) for the complete system. For information about *usable capacity*, refer to 2.3, “Full storage virtualization” on page 14.

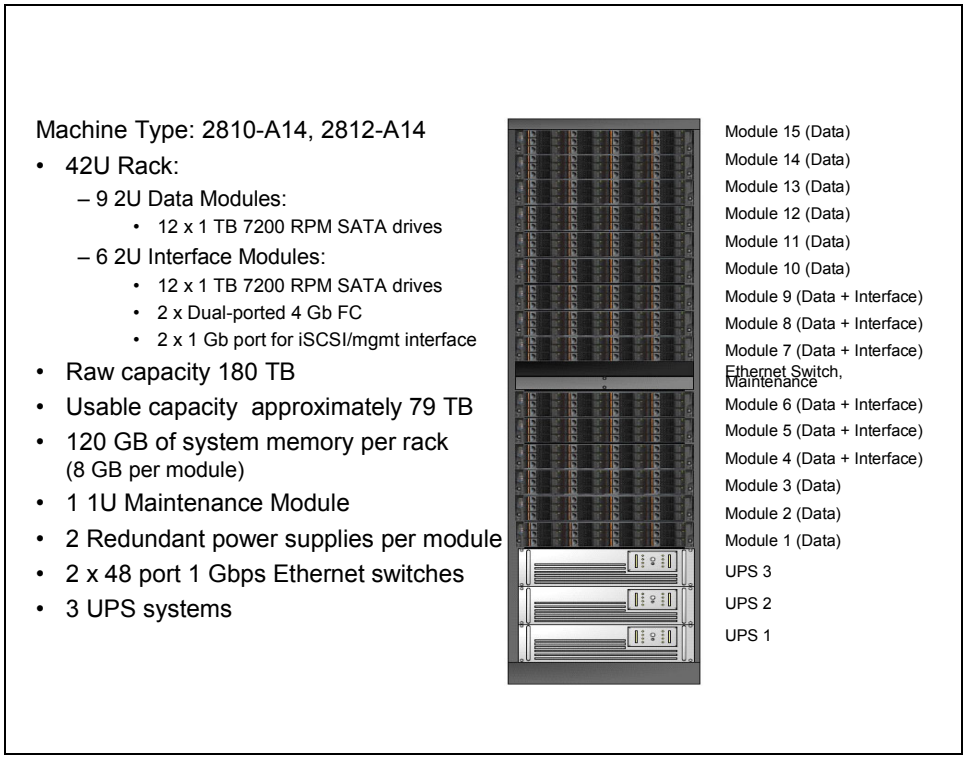


Figure 3-2 Hardware overview - Machine type 2810_2812 - model A14

Partially populated configurations

The IBM 2810-A14 and IBM 2812-A14 are also available in partially configured racks allowing for more granular capacity configurations. These partially configured racks are available in usable capacities ranging from 27 to 79 TB.

Details on these configuration options and the various capacities, drives, ports, and memory are provided in Figure 3-3.

Total Modules	6	9	10	11	12	13	14	15
Useable Capacity (TB)	27	43	50	54	61	66	73	79
Interface Modules (Feature #1100)	3	6	6	6	6	6	6	6
Data Modules (Feature #1105)	3	3	4	5	6	7	8	9
Disk Drives	72	108	120	132	144	156	168	180
Fibre Channel Ports	8	16	16	20	20	24	24	24
iSCSI Ports	0	4	4	6	6	6	6	6
Memory (GB)	48	72	80	88	96	104	112	120

Figure 3-3 XIV partial configurations

You can order from manufacturing any partial configuration of 6 or 9, 10, 11, 12, 13, 14, 15 modules. You also have the option of upgrading already deployed partial configurations to achieve configurations with a total of nine, ten, eleven, twelve, thirteen, fourteen, or fifteen modules.

Module interconnections

The system includes two Ethernet switches (1 Gbps, 48 ports). They form the basis of an internal redundant Gigabit Ethernet network that links all the modules in the system. The switches are installed in the middle of the rack just above module 6.

The connections between the modules and switches, including the internal power connections, are all fully redundant with a second set of cables. For power connections, standard power cables and plugs are used. Additionally, standard Ethernet cables are used for interconnection between the modules and switches.

All 15 modules (6 Interface Modules and 9 Data Modules) have redundant connections through the two 48-port 1Gbps Ethernet switches. This grid network ensures communication between all modules even if one of the switches or a cable connection fails. Furthermore, this grid network provides the capabilities for parallelism and the execution of a data distribution algorithm that contributes to the excellent performance of the XIV Storage System.

3.2 IBM XIV hardware components

The system architecture of the XIV Storage Subsystem is designed specifically upon off-the-shelf components[1] that are not dependent upon specifically designed hardware or proprietary technology. This design in architecture is optimized for ease of use such that, as newer and higher performing components are made available in the marketplace, development is able to incorporate this newer technology into the base system design at a faster pace than was traditionally possible. In the sections that follow, we explore these base components in further detail.

[1] With the exception of the Automatic Transfer System (ATS)

At a minimum, for all configurations, the following components are supplied with the system:

- 3 UPS Units
- 2 Ethernet Switches
- 1 Ethernet Switch Redundant Power Supply
- 1 Maintenance Module
- 1 Automatic Transfer Switch (ATS)
- 1 Modem
- 8-24 Fibre Channel Ports
- 0-6 iSCSI Ports
- Complete set of internal cabling

Note: For the same reason that the system is not dependent on specially developed parts, there might be differences in the actual hardware components that are used in your particular system compared with those components described next.

3.2.1 Rack and UPS modules

This section describes the hardware rack and UPS modules.

Rack

The IBM XIV hardware components are installed in a standard 482.6 mm (19 inches) rack with a newly redesigned door with the release of the 2009 2810-A14 and 2812-A14 hardware (Figure 3-4).

Figure 3-4 XIV Model 2810 & 2812 redesigned door

The rack is 1070 mm (42 inches) deep (not including the doors) to accommodate deeper size modules and to provide more space for cables and connectors. Adequate space is provided to house all components and to properly route all cables. The rack door and side panels are locked with a key to prevent unauthorized access to the installed components. For detailed

dimensions, clearances, and the weight of the rack and its components, refer to 3.3.2, “Physical site planning” on page 63.

UPS module complex

The Uninterruptible Power Supply (UPS) module complex consists of three UPS units. Each unit maintains an internal 30 seconds storage of system power, for use in the event of any temporary failure of the external power supply to protect the system from failure. In case of an extended external power failure or outage, the UPS module complex maintains battery power long enough to allow a safe and orderly shutdown of the XIV Storage System. The complex can sustain the failure of one UPS unit, while protecting against external power outages.

Figure 3-5 shows an illustration of one UPS module.

Figure 3-5 UPS

The three UPS modules are located at the bottom of the rack. Each of the modules has an output of 6 kVA to supply power to all other components in the rack and is 3U in height. The UPS module complex design allows proactive detection of temporary power problems and can correct them before the system goes down. In the case of a complete power outage, integrated batteries continue to supply power to the entire system. The batteries are designed to last long enough for a safe and ordered shutdown of the IBM XIV Storage System.

Important: Do not power off the XIV using the UPS power button because this can result in the loss of data and system configuration information. We recommend that you use the GUI to power off the system.

Automatic Transfer System (ATS)

The Automatic Transfer System (ATS) seen in Figure 3-6 supplies power to all three Uninterruptible Power Supplies (UPS) and to the Maintenance Module. In case of a power problem on one line, the ATS reorganizes the power and switches to the other line.

Note: Although the system is protected by an uninterruptible power supply for internal usage, you can reduce the risk of a power outage if you connect the system to an external uninterruptible power supply, a backup generator, or both.

The operational components take over the load from the failing power source or power supply. This rearrangement is performed by the ATS in a seamless manner such that the system operation continues without any application impact.

The ATS is available as a single-phase power or three-phase power. Depending on the ATS and your geography, the XIV Storage System is available in multiple line cord configurations. For the appropriate line cord selection, refer to the *IBM XIV Storage System (Types 2810 and 2812) Model A14 (Gen2) Introduction and Planning Guide for Customer Configuration*, GA52-1327-07.

Single-phase power ATS

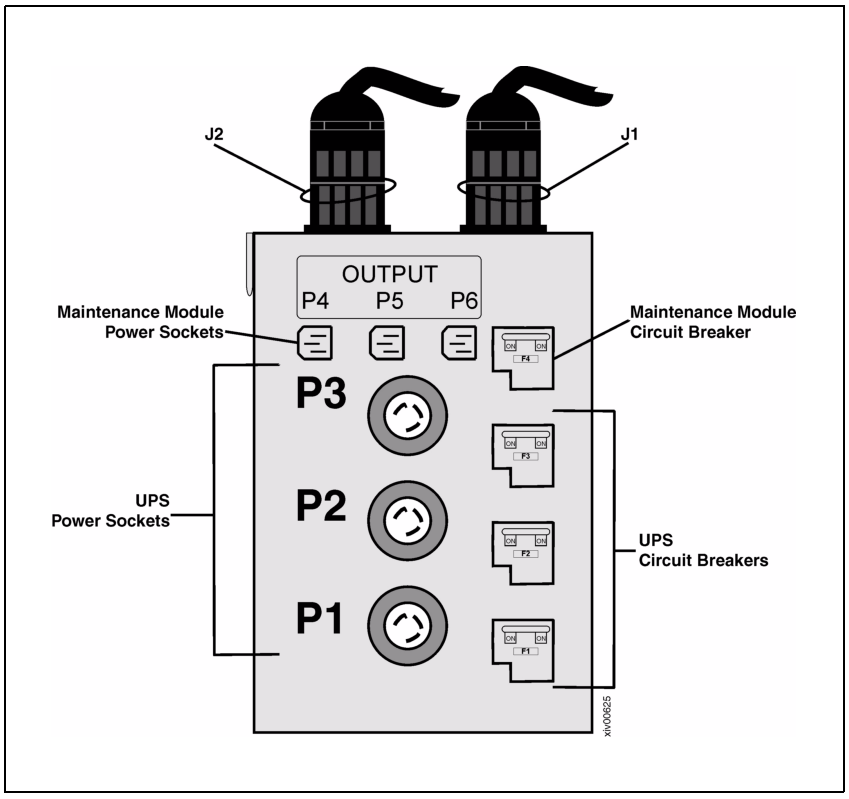


Figure 3-6 Automatic Transfer System ATS

Two separate external main power sources supply power to the ATS. The following power options are available:

- Two 60 A, 200-240 V ac, single-phase, two-pole, line-line ground female receptacles, each connected to a different power source
- Four 30 A, 200-240 V ac, single-phase, two-pole, line-line ground female receptacles, connected to two (2) independent power sources

Note that if you do not have the two 60 amp power feeds normally required and use instead four 30 amp power feeds, two of the lines will go to the ATS, which is then only connected to UPS unit 2. One of the other two lines goes to UPS unit 1 and the other line goes to UPS unit 2, as seen in Figure 3-7.

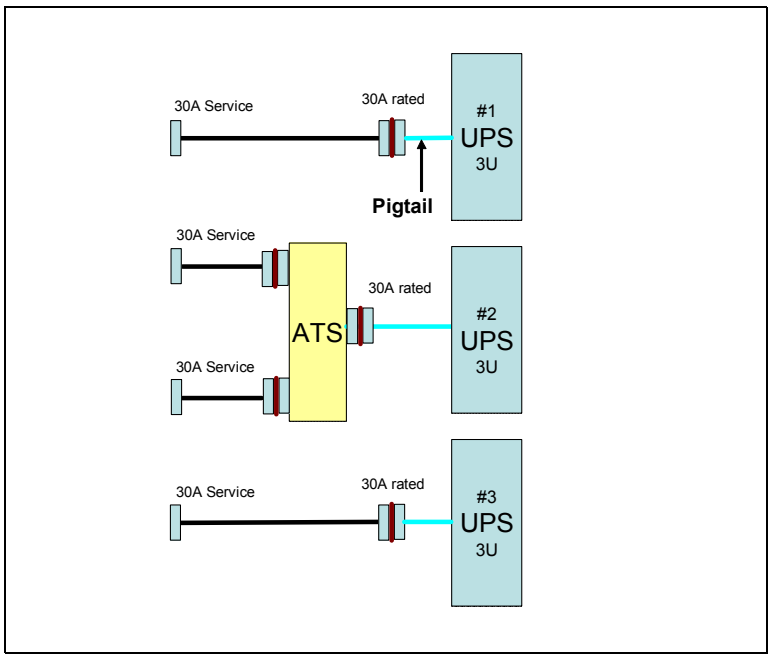


Figure 3-7 Single-phase power ATS with 30 amp power feeds

Three-phase power ATS

A newer, three-phase power ATS provides additional options to power the IBM XIV Storage System in your data center. Single-phase power remains available.

Two separate external main power sources supply power to the ATS. The following power options are available:

- Two 30 A, 200-240 V ac, three-phase receptacles, each connected to a differentpower source
- Two 60 A, 200-240 V ac, three-phase receptacles, each connected to a different power source

3.2.2 Data Modules and Interface Modules

The hardware of the Interface Modules and Data Modules is based on an Intel server platform optimized for data storage services. A module is 87.9 mm (3.46 inches) (2U) tall, 483 mm (19 inches) wide, and 707 mm (27.8 inches) deep. The weight depends on configuration and type (Data Module or Interface Module) and is a maximum of 30 kg (66.14 lbs). Figure 3-8 shows a representation of a module in perspective.

New Interface Modules now contain a dual-CPU (Interface Module feature numbers 1101 and 1111 (with capacity on demand)). These features are used to provide additional CPU bandwidth to the Interface Modules installed in the XIV system. The new dual-CPU is also a low voltage CPU that reduces power consumption. Note, however, that a feature conversion from single-CPU Interface Modules (1100) to dual-CPU Interface Modules (1101) is not offered.

New Data Module feature numbers 1106 and 1116 (capacity on demand) can also include a new low voltage CPU and are a like-for-like replacement of previous Data Module feature numbers 1105 and 1115 (capacity on demand) using newer components.

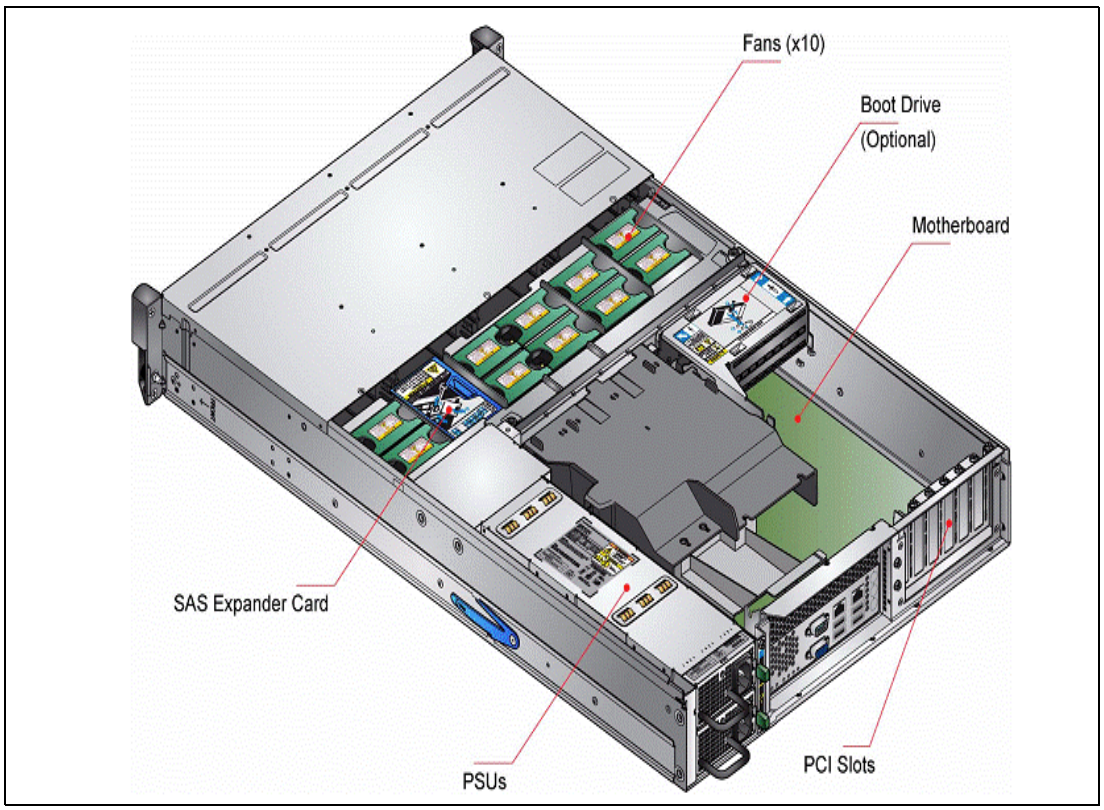


Figure 3-8 Data Module/Interface Module

Data Module

The fully populated rack hosts 9 Data Modules (Module 1-3 and Module 10-15). The only difference between Data Modules and Interface Modules (refer to “Interface Module” on page 54) are the additional host adapters and GigE adapters in the Interface Modules as well as the option of a dual CPU configuration for the new Interface Modules. The main components of the modules, in addition to the 12 disk drives, are:

- System Planar (Motherboard)
- Processor
- Memory/cache
- Enclosure Management Card
- Cooling devices (fans)
- Memory Flash Card
- Redundant power supplies

In addition, each Data Module contain four redundant Gigabit Ethernet ports. These ports together with the two switches form the internal network, which is the communication path for data and metadata between all modules. One Dual GigE adapter is integrated in the System Planar (port 1 and 2). The remaining two ports (3 and 4) are on an additional Dual GigE adapter installed in a PCIe slot as seen in Figure 3-9.

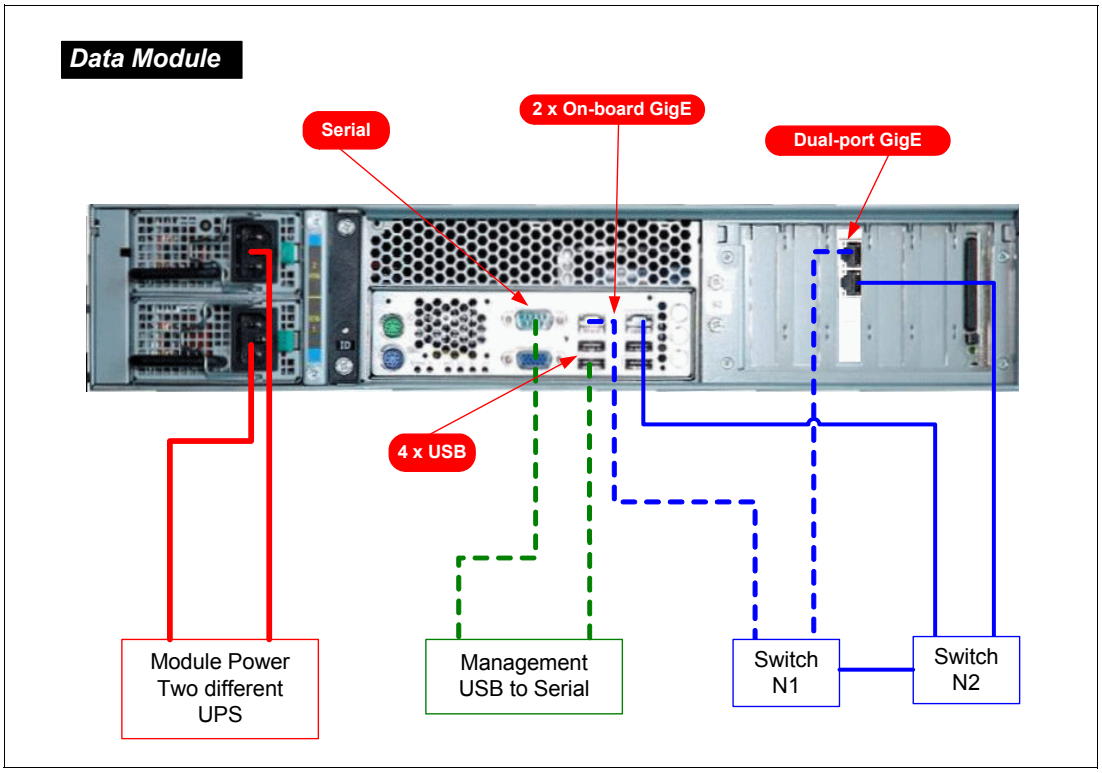


Figure 3-9 Data Module connections

System planar

The system planar used in the Data Modules and the Interface Modules is a standard ATX board from Intel. This high-performance server board with a built-in Serial-Attached SCSI (SAS) adapter supports:

- Single or Dual (for Interface Modules only) 64-bit quad-core Intel Xeon® processors:
  - The dual-CPU Interface Modules ship with two low-voltage Central Processing Units (CPUs). These modules show an improvement in performance on sequential read and write operations over the single-CPU predecessor. The single CPU Interface Module uses the same low voltage CPU.
  - Dual-CPU Interface Modules (feature number 1101) can be used to complete the Interface Module portion of a six-module configuration that already has three single-CPU Interface Modules (feature number 1100).
  - A feature conversion from single-CPU Interface Modules (1100) to dual-CPU Interface Modules (1101) is not allowed.
- 8 x 1 GB or 4 x 2 GB fully buffered 533/667 MHz Dual Inline Memory Module (DIMMs) to increase capacity and performance
- Dual Gb Ethernet with Intel I/O Acceleration Technology to improve application and network responsiveness by moving data to and from applications faster
- Four PCI Express slots to provide the I/O bandwidth needed by servers
- SAS adapter

Processor

The processor is either one or two Intel Xeon Quad Core Processors. This 64-bit processor has the following characteristics:

- 2.33 GHz clock
- 12 MB cache
- 1.33 GHz Front Serial Bus
- Low voltage power profile

For systems already deployed, there are no options to replace the current CPU in the Data and Interface Modules with the newer low voltage processor. For partially populated configurations, it is possible to expand the system with new Modules utilizing the new CPU design.

Memory/Cache

Every module has 8 GB of memory installed (either 4x2GB or 8 x 1GB) Fully Buffered DIMM (FBDIMM). FBDIMM memory technology increases reliability, speed, and density of memory for use with Xeon Quad Core Processor platforms. This processor memory configuration can provide three times higher memory throughput, enable increased capacity and speed to balance capabilities of quad core processors, perform reads and writes simultaneously, and eliminate the previous read to write blocking latency.

Part of the memory is used as module system memory, while the rest is used as cache memory for caching data previously read, pre-fetching of data from disk, and for delayed destaging of previously written data. For a description of the cache algorithm, refer to “Write cache protection” on page 34.

Cooling fans

To provide enough cooling for the disks, processor, and board, the system includes 10 fans located between the disk drives and the board. The cool air is aspirated from the front of the module through the disk drives. An air duct leads the air around the processor before it leaves the module through the back. The air flow and the alignment of the fans assure proper cooling of the entire module, even if a fan is failing.

Enclosure management card

The enclosure management card is located between the disk drives and the system planar. In addition to the internal module connectivity between the drive backplane and the system planar, this card is the backplane for the 10 fans. Furthermore, it includes fan control and the logic to generate hardware alarms in the case of problems in the module.

Compact Flash Card

Each module contains a Compact Flash Card (1 GB) in the right-most rear slot. Refer to Figure 3-10.

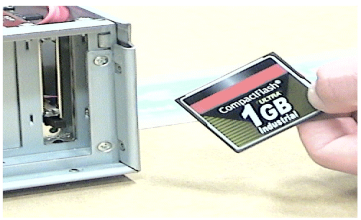


Figure 3-10 Compact Flash Card

This card is the boot device of the module and contains the software and module configuration files.

Important: Due to the configuration files, the Compact Flash Card is not interchangeable between modules.

Power supplies

Figure 3-11 shows the redundant power supply units.

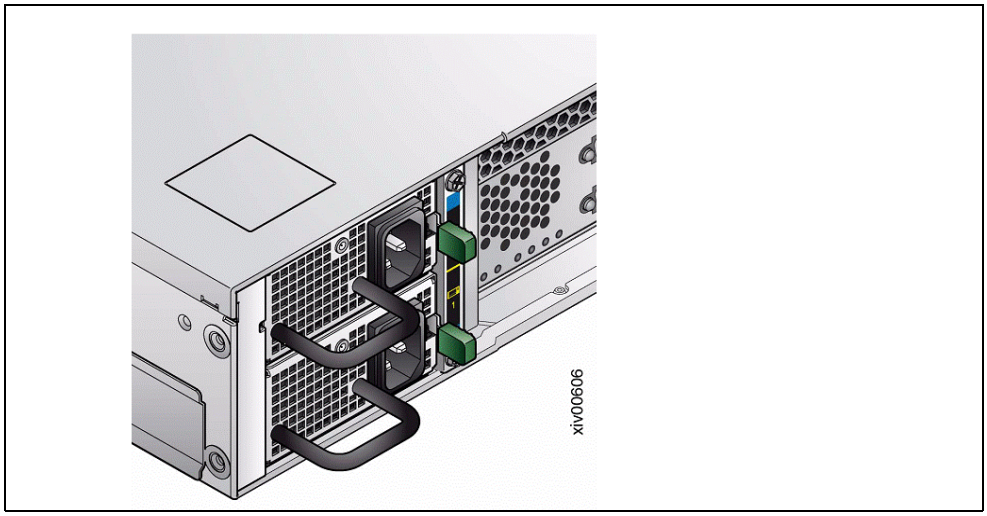


Figure 3-11 Redundant module power supply units

The modules are powered by a redundant Power Supply Unit (PSU) cage with a dual 850W PSU assembly as seen in Figure 3-11. These power supplies are redundant and can be individually replaced with no need to stop the stop the system. The power supply is a Field-Replaceable Unit (FRU).

Interface Module

Figure 3-12 shows an Interface Module with iSCSI ports.

The Interface Module is similar to the Data Module. The only differences are as follows:

- Interface Modules contain iSCSI and Fibre Channel ports, through which hosts can attach to the XIV Storage System. These ports can also be used to establish Remote Mirror links and data migration paths with another remote XIV Storage System.
- There are two 4-port GigE PCIe adapters installed for additional internal network connections as well as for iSCSI host connections.

All Fibre Channel ports, iSCSI ports, and Ethernet ports used for external connections are internally connected to a patch panel where the external cable connections are made. Refer to 3.2.4, “Patch panel” on page 58.

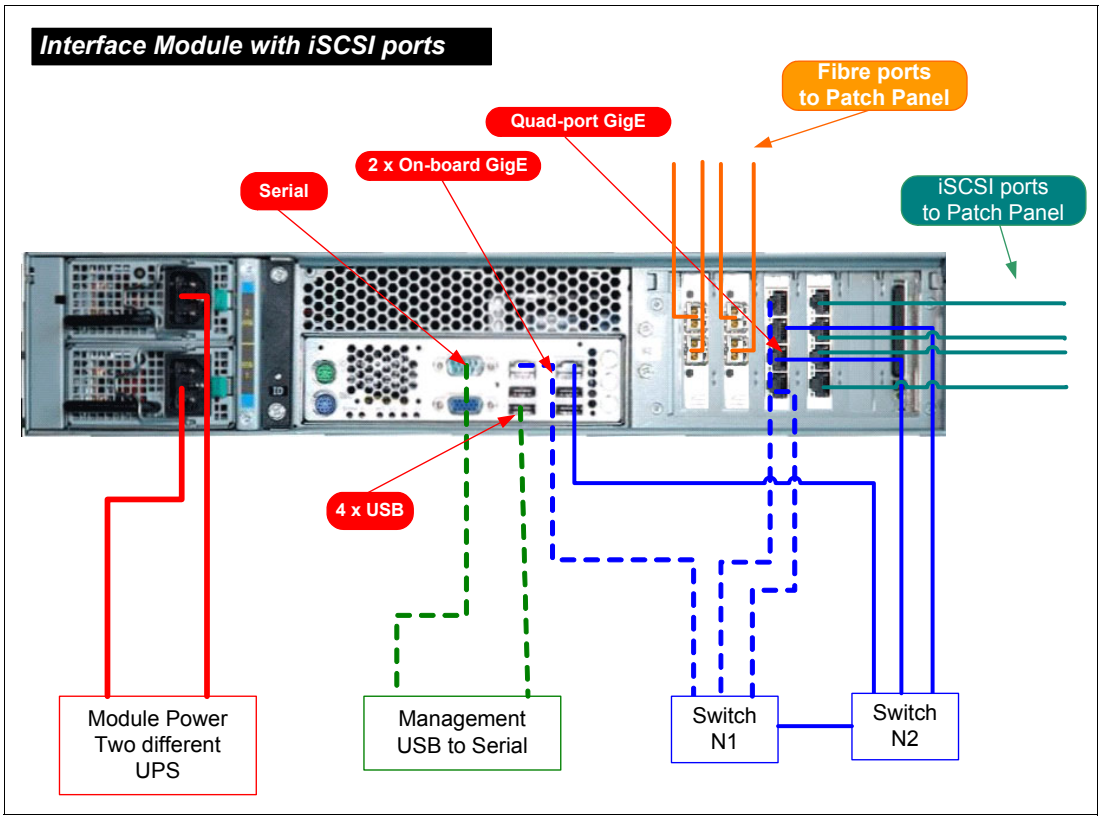


Figure 3-12 Interface Module with iSCSI ports

Fibre Channel connectivity

There are four Fibre Channel ports available in each Interface Module for a total of 24 Fibre Channel ports. They support 1, 2, and 4 Gbps full-duplex data transfer over short wave fibre links, using 50 micron multi-mode cable and support new end-to-end error detection through a Cyclic Redundancy Check (CRC) for improved data integrity during reads and writes.

In each module, the ports are allocated in the following manner:

- Ports 1 and 3 are allocated for host connectivity.
- Ports 2 and 4 are allocated for additional host connectivity or remote mirror and data migration connectivity.

Note: Utilizing more than 12 Fibre Channel ports for host connectivity will not necessarily provide more bandwidth. Best practice is to utilize enough ports to support multipathing, without overburdening the host with too many paths to manage.

iSCSI connectivity

There are 6 iSCSI ports (two ports per Interface Modules 7 through 9) available for iSCSI over IP/Ethernet services. These ports support 1Gbps Ethernet network connection. These ports connect to the user's IP network through the Patch Panel and provide connectivity to the iSCSI hosts. Refer to Figure 3-14 on page 58 for additional details on the cabling of these ports.

You can operate iSCSI connections for various functionalities:

- As an iSCSI target that the server hosts through the iSCSI protocol
- As an iSCSI initiator for Remote Mirroring when connected to another iSCSI port

- As an iSCSI initiator for data migration when connected to a third-party iSCSI storage system

For each iSCSI IP interface, you can define these configuration options:

- IP address (mandatory)
- Network mask (mandatory)
- Default gateway (optional)
- MTU; Default: 1 536; Maximum: 8 192 MTU

Note: iSCSI has been tested and approved with software based initiators only.

3.2.3 SATA disk drives

The SATA disk drives, which are shown in Figure 3-13 and used in the IBM XIV, are 1 TB, 7200 rpm hard drives designed for high-capacity storage in enterprise environments. These drives are manufactured to a higher set of standards than typical off the shelf SATA disk drives to ensure a longer life and increased mean time between failure (MTBF).

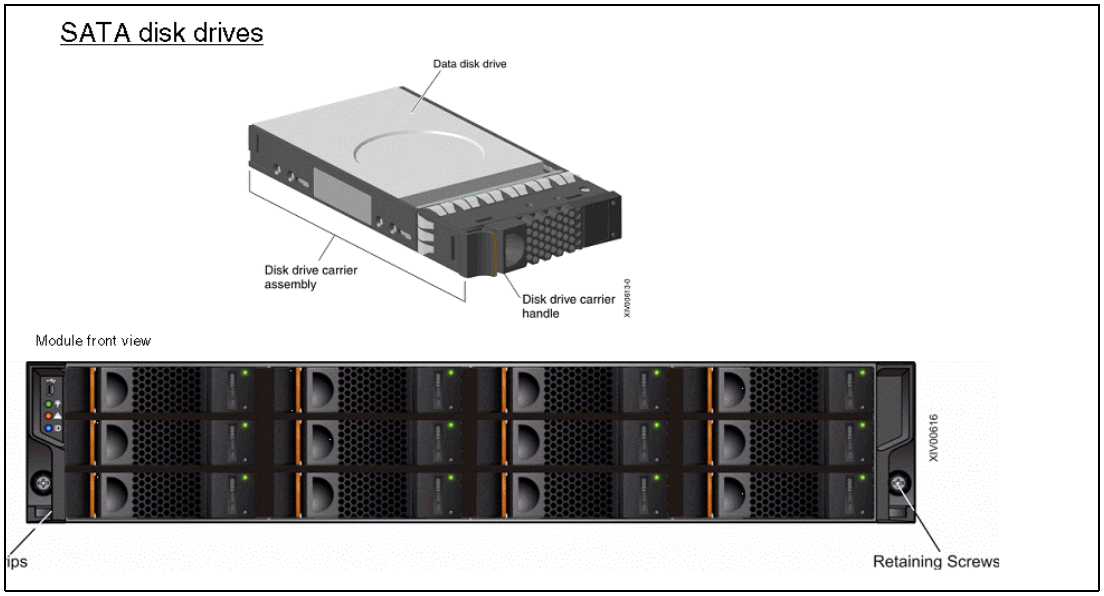


Figure 3-13 SATA disks

The IBM XIV Storage System was engineered with substantial protection against data corruption and data loss. Several features and functions implemented in the disk drive also increase reliability and performance. We describe the highlights next.

Performance features and benefits

Performance features and benefits include:

- SAS interface:

  The disk drive features a 3 Gbps SAS interface supporting key features in the SATA specification, including Native Command Queuing (NCQ) and staggered spin-up and hot-swap capability.

- 32 MB cache buffer:

  The internal 32 MB cache buffer enhances the data transfer performance.

- Rotation Vibration Safeguard (RVS):

In multi-drive environments, rotational vibration, which results from the vibration of neighboring drives in a system, can degrade hard drive performance. To aid in maintaining high performance, the disk drive incorporates the enhanced Rotation Vibration Safeguard (RVS) technology, providing up to a 50% improvement over the previous generation against performance degradation, and therefore, leading the industry.

Reliability features and benefits

Reliability features and benefits include:

- Advanced magnetic recording heads and media:

  There is an excellent soft error rate for improved reliability and performance.

- Self-Protection Throttling (SPT):

  SPT monitors and manages I/O to maximize reliability and performance.

- Thermal Fly-height Control (TFC):

  TFC provides a better soft error rate for improved reliability and performance.

- Fluid Dynamic Bearing (FDB) Motor:

  The FDB Motor improves acoustics and positional accuracy.

- Load/unload ramp

  The R/W heads are placed outside the data area to protect user data when the power is removed.

All IBM XIV disks are installed in the front of the modules, twelve disks per module. Each single SATA disk is installed in a disk tray, which connects the disk to the backplane and includes the disk indicators on the front. If a disk is failing, it can be replaced easily from the front of the rack. The complete disk tray is one FRU, which is latched in its position by a mechanical handle.

Important: SATA disks in the IBM XIV Storage System must never be swapped within a module or placed in another module because of internal tracing and logging data that they maintain.

3.2.4 Patch panel

The patch panel is located at the rear of the rack. Interface Modules are connected to the patch panel using 50 micron cables. All external connections must be made through the patch panel. In addition to the host connections and to the network connections, more ports are available on the patch panel for service connections. Figure 3-14 shows the details for the patch panel and the ports.

The patch panel has had several re-designs with respect to the labelling based on production date and is much easier to read in the latest configurations.

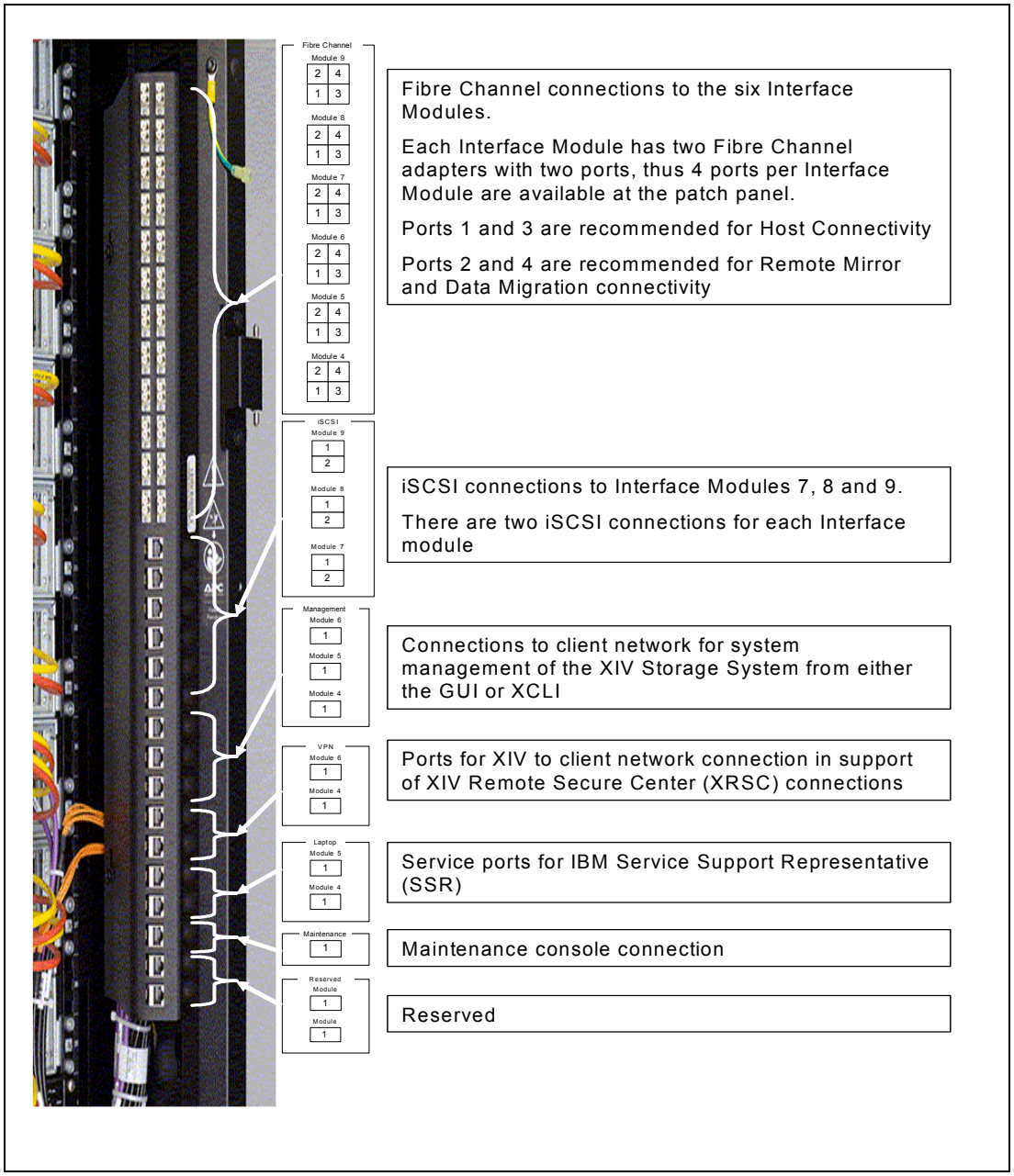


Figure 3-14 Connection details of Patch Panel ports

3.2.5 Interconnection and switches

The internal network is based on two redundant 48-port Gigabit Ethernet switches. Each of the modules (Data or Interface) is directly attached to each of the switches with multiple connections (refer to Figure 3-9 on page 52 and Figure 3-12 on page 55), and the switches are also linked to each other. This network topology enables maximum bandwidth utilization, as the switches are used in an active-active configuration, while being tolerant to any failure of the following individual network components:

- Ports
- Links
- Switches

Figure 3-15 shows the two ethernet switches and the cabling to them.

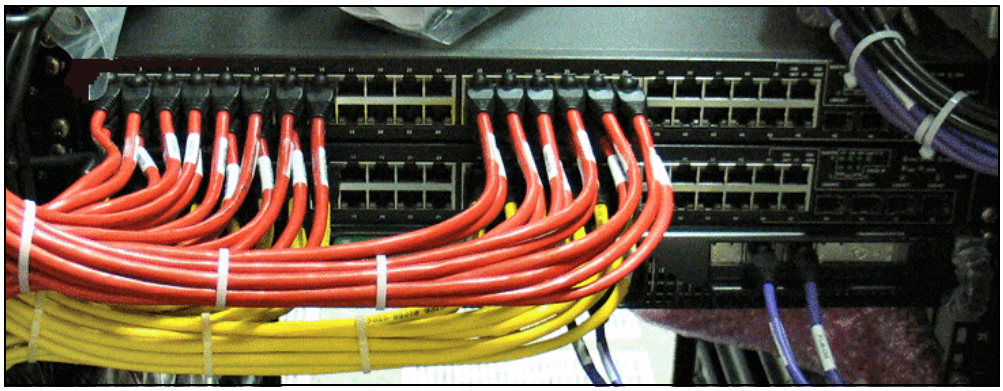


Figure 3-15 48 Port Gigabit Ethernet Switch

The Gigabit Ethernet Layer 3-Switch contains 48 copper and 4 fiber ports (small form-factor plugable (SFP) capable of one of 3 speeds, 10/100/10000 Mbps), robust stacking, and 10 Gigabit-Ethernet uplink capability. The switches are powered by redundant power supplies to eliminate any single point of failure.

3.2.6 Support hardware

This section covers important features of the XIV Storage System used by internal functions and/or IBM maintenance, should a problem arise with the system.

Module USB to Serial connections

The Module USB to Serial connections are used by internal system processes to keep the communication to the modules alive in the event that the network connection is not operational. Modules are linked together with USB to Serial cables in groups of three modules. This emergency link is needed to communicate between the modules for internal processes and are used by IBM Maintenance in the event of repair to the internal network.

The USB to Serial connection always connects a group of three Modules:

- USB Module 1 is connected to Serial Module 3.
- USB Module 3 is connected to Serial Module 2.
- USB Module 2 is connected to Serial Module 1.

This connection sequence is repeated for the modules 4-6, 7-9, 10-12, and 13-15.

For partially configured systems—such as 10 or 11 modules, for example—the USB to Serial connections follow the same pattern as applicable.

In the case of a system with 10 modules, modules 1-3 would be connected together, as would 4-6 and 7-9, while Module 10 would be unconnected via this method.

This connection sequence for a fully configured system (15 modules) is depicted in the diagram shown in Figure 3-16.

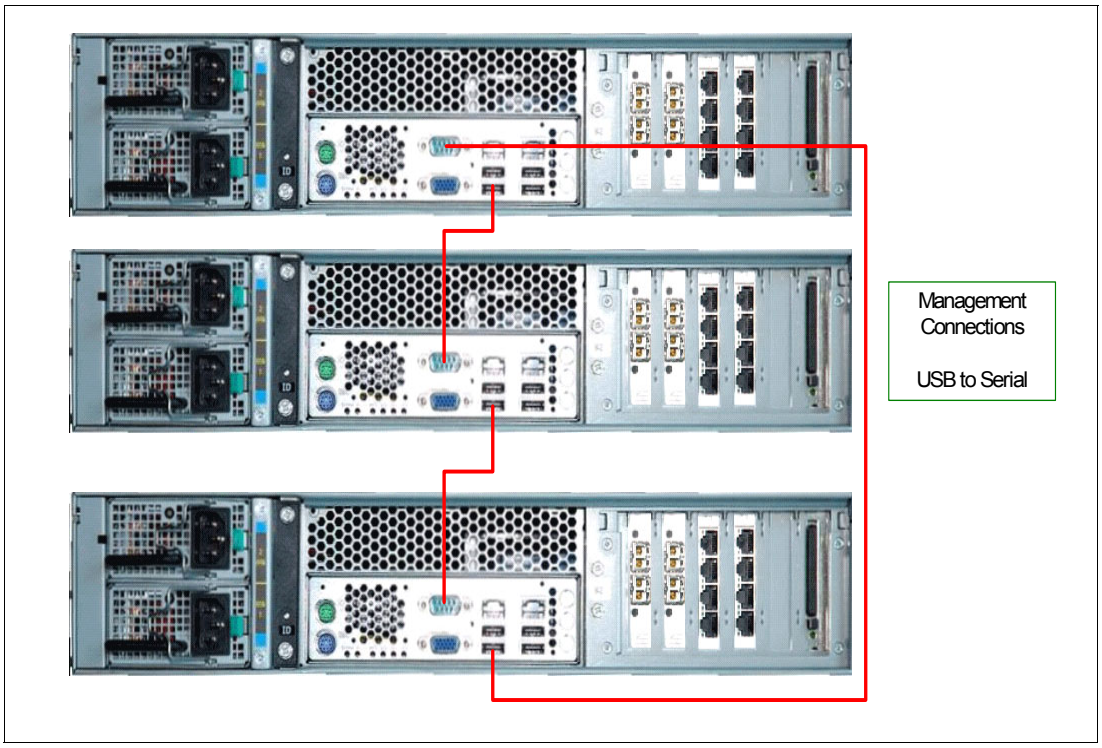


Figure 3-16 Module: USB to serial

Modem

The modem installed in the rack is optionally used for remote support if the preferred choice of XIV Secure Remote Support is not selected. It enables the IBM XIV Support Center specialists and, if necessary, a higher level of support to connect to the XIV Storage System. Problem analysis and repair actions without a remote connection can be complicated and time-consuming.

Note: The modem is not available in all countries.

Maintenance module

A 1U remote support server is also provided for the full functionality and supportability of the IBM XIV Storage System. This device has fairly generic requirements because it is only used to gain remote access to the XIV Storage System through the Secure Remote Support connection or modem for support personnel.

The maintenance module and the modem, which are installed in the middle of the rack, are used for IBM XIV Support and the IBM service support representative (SSR) to maintain and repair the machine. When there is a software or hardware problem that needs the attention of the IBM XIV Support Center, a remote connection will be required and used to analyze and possibly repair the faulty system. This connection is always initiated by the customer and is done either through the XIV Secure Remote Support (XSRC) facility or through a phone line and modem. For further information about remote connections refer to “XIV Remote Support Center (XRSC)” on page 72.

3.2.7 Hardware redundancy

The IBM XIV hardware is redundant to prevent machine outage when any single hardware component is failing. The combination of hardware redundancy with the logical architecture that is described in Chapter 2, “XIV logical architecture and concepts” on page 9 makes the XIV Storage System extremely resilient to outages.

Power redundancy

To prevent the complete rack or single components from failing due to power problems, all power components in the IBM XIV are redundant:

- To ensure redundant power availability at the rack level, a device must be present to enable switching from one power source to another available power source, which is realized by an Automatic Transfer Switch (ATS). In the case of a failing UPS, this switch transfers the load to the remaining two UPS’s without interrupting the system.
- Each module has two independent power supplies. During normal operation, both power supplies operate on half of the maximal load. If one power supply fails, the operational power supply can take over, and the module continues its operation without any noticeable impact. After the failing power supply is replaced, the power load balancing is restored.
- The two switches are powered by the PowerConnect RPS-600 Redundant Power Bank to eliminate the power supply as a single point of failure.

Switch/Interconnect redundancy

The IBM XIV internal network is built around two Ethernet switches which are interconnected for redundancy. Each module (Data and Interface) also has multiple connections to both switches to eliminate any failing hardware component within the network from becoming a single point of failure.

3.3 Hardware planning overview

This section provides an overview of planning considerations for the XIV Storage System, including a reference listing of the information required for the setup. The information in this chapter includes requirements for:

- Physical installation
- Delivery requirements
- Site requirements
- Cabling requirements

For more detailed planning information, refer to the *IBM XIV Storage System Installation and Planning Guide for Customer Configuration, GC52-1327*, and to the IBM *XIV Storage System Pre-Installation Network Planning Guide for Customer Configuration,* GC52-1328*.*

Additional documentation is available from the XIV InfoCenter at:

`http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp`

For a smooth and efficient installation of the XIV Storage System, planning and preparation tasks must take place before the system is scheduled for delivery and installation in the data center.

There are four major areas involved in installation planning:

- Ordering the BM XIV hardware:
  - Selecting the appropriate and required features
- Physical site planning for:
  - Space, dimensions, and weight
  - Raised floor
  - Power, cooling, cabling, and additional equipment
- Configuration planning:
  - Basic configurations
  - Network connections
  - Management connections
- Installation:
  - Physical installation

3.3.1 Ordering IBM XIV hardware

This part of the planning describes ordering the appropriate XIV hardware configuration. At this point, consider actual requirements, but also consider potential future requirements.

Feature codes and hardware configuration

The XIV Storage System hardware is mostly pre-configured and consists of two machine types, namely the 2810-A14 and the 2812-A14, which as previously mentioned, only differ in their initial warranty coverage and are equipped with dual CPU Interface Modules. All features are identical on each system.

There are only a few optional or specific hardware features that you can select as part of the initial order. Refer to the *IBM XIV Storage System Installation and Planning Guide for Customer Configuration*, GC52-1327 for details.

3.3.2 Physical site planning

Physical planning considers the size, weight, and the environment on which you will install the IBM XIV Storage System.

Site requirements

The physical requirements for the room where the XIV Storage System is going to be installed must be checked well ahead of the arrival of the machine:

- The floor must be able to withstand the weight of the XIV Storage System to be installed.

  Consider also possible future machine upgrades. The XIV Storage System can be installed on a non-raised floor, but we highly recommend that you use a raised floor for increased air circulation and cooling.

- Enough clearance around the system must be left for cooling and service.

  The airflow enters the system rack on the front and is expelled to the rear. Also, you must ensure that there is enough service clearance because space must be available to fully open the front and back doors.

- Consider also the building particularities, such as any ramps, elevators, and floor characteristics according to the height and weight of the machine. Remember that the system is housed in a tall (42U) rack.

Figure 3-17 gives a general overview of the clearance needed for airflow and service around the rack.

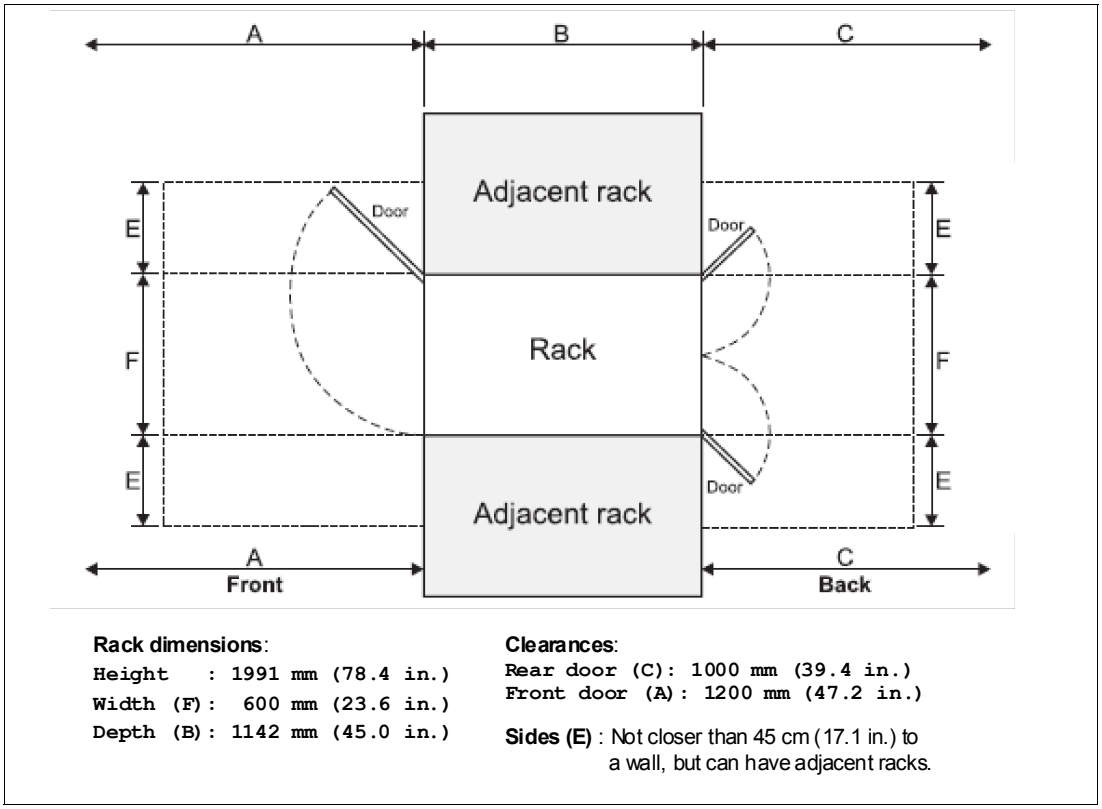


Figure 3-17 Dimension and weight

For detailed information and further requirements, refer to the *IBM XIV Installation Planning Guide.*

Weight and raised floor requirements

The system is normally shipped as one unit. There is a weight reduction feature available (FC 0200) in case the access to the site cannot accommodate the weight of the fully populated rack during its movement to the final install destination.

The following measurements are provided for your convenience. For the latest and most accurate information, refer to the *IBM XIV Storage System Installation and Planning Guide for Customer Configuration*, GC52-1327.

- IBM Storage System XIV, 2810-A14 or 2812-A14 weight:
  - 884 kg (1949 lb.)
- Raised floor requirements:
  - Reinforcement is needed to support a weight of 800 kg (1760 lb) on an area of 60 cm x 109 cm.
  - Provide enough ventilation tiles in front of the rack.
  - Provide a cutout (opening) for the cables according to the template in the *IBM XIV Storage System Model 2810 Installation Planning Guide.*

The installation and planning guide lists the following requirements:

- Recommended power requirements:
- Cooling requirements:
- Delivery requirements:

3.3.3 Basic configuration planning

You must complete the configuration planning first to allow the IBM SSR to physically install and configure the system.

In addition, you must provide the IBM SSR with the information required to attach the system to your network for operations and management, as well as enabling remote connectivity for IBM support and maintenance. Figure 3-18 summarizes the required information.

Customer Network	Customer IP Address	Netmask	Default Gateway
Interface Module 4			
Interface Module 5			
Interface Module 6			
Primary DNS Server			
Secondary DNS Server			
SMTP Gateway(s)			
NTP (Time Server)			
SNMP Server			
Time Zone			
Email sender address			
Remote Access / VPN	**IP Address**	**Netmask**	**Default Gateway**
Remote Support Server Customer Interface			
External IP address (Address exposed to Internet (NAT?)):			
VPN software required at XIV's site			
Modem Phone Number			

Figure 3-18 Network and remote connectivity

Fill in all information to prevent further inquiry and delays during the installation (refer to 3.3.4, “IBM XIV physical installation” on page 73):

- Interface Module:

  Interface Modules (4,5 and 6) need an IP address, Netmask, and Gateway. This address is needed to manage and monitor the IBM XIV with either the GUI or Extended Command Line Interface (XCLI). Each Interface Module needs a separate IP address in case a module is failing.

- DNS server:

  If Domain Name System (DNS) is used in your environment, the IBM XIV needs to have the IP address, Netmask, and Gateway from the primary DNS server and, if available, also from the secondary server.

- SMTP Gateway:

  The Simple Mail Transfer Protocol (SMTP) Gateway is needed for event notification through e-mail. IBM XIV can initiate an e-mail notification, which will be sent out through the configured SMTP Gateway (IP Address or server name, Netmask, and Gateway)

- NTP (Time server):

  IBM XIV can be used with a Network Time Protocol (NTP) time server to synchronize the system time with other systems. To use this time server, IP Address, or server name, Netmask and Gateway need to be configured.

- Time zone:

  Usually the time zone depends on the location where the system is installed. But, exceptions can occur for remote locations where the time zone equals the time of the host system location.

- E-mail sender address;

  This is the e-mail address that is shown in the e-mail notification as the sender.

- Remote access:

  The modem number or a client side IP Address needs to be configured for remote support. This network connection must have outbound connectivity to the Internet.

This basic configuration data will be entered in the system by the IBM SSR following the physical installation. Refer to “Basic configuration” on page 74.

Other configuration tasks, such defining storage pools, volumes, and hosts, are the responsibility of the storage administrator and are described in Chapter 4, “Configuration” on page 79.

Network connection considerations

Network connection planning is also essential to prepare to install the XIV Storage System. To deploy and operate the system in your environment, a number of network connections are required:

- Fibre Channel connections for host I/O over Fibre Channel
- Gigabit Ethernet connections for host I/O over iSCSI
- Gigabit Ethernet connections for management
- Gigabit Ethernet connections for IBM XIV remote support
- Gigabit Ethernet connections for the IBM SSR (field technician ports)

All external IBM XIV connections are hooked up through the patch panel as explained in 3.2.4, “Patch panel” on page 58.

For details about the host connections, refer to Chapter 6, “Host connectivity” on page 183.

Fibre Channel connections

When shipped, the XIV Storage System is by default equipped with 24 Fibre Channel ports (assuming a fully populated 15 Module rack). The IBM XIV supports 50 micron fiber cabling. If you have other requirements or special considerations, contact your IBM Representative.

The 24 FC ports are available from the six Interface Modules, four in each module, and they are internally connected to the patch panel. Of the 24 ports, 12 are provided for connectivity to the switch network for host access and the remaining 12 are for use in remote mirroring or data migration scenarios (however, they can be reconfigured for host connectivity). We recommend that you adhere to this guidance on Fibre Channel connectivity. The external (client-provided) cables are plugged into the patch panel. For planning purposes, Table 3-2 highlights the maximum values for various Fibre Channel parameters for your consideration. These values are correct, at the time of writing this book, for Release 10.1 of the IBM XIV Storage System software.

Refer to Chapter 6, “Host connectivity” on page 183 for details on Fibre Channel configuration and connections.

Table 3-2 Maximum values in context of FC connectivity

FC Parameters	Maximum values
Maximum number of Interface Modules	6
Maximum number of 4 GB FC ports per Interface Module	4
Maximum queue depth per FC host port	1400
Maximum queue depth per mapped volume per (host port, target port, volume) tuple	256
Maximum FC ports for host connections (default configuration)	12
Maximum FC ports for migration/mirroring (default config)	12
Maximum volumes mapped per host	512
Maximum number of clusters	100
Maximum number of hosts (defined WWPNs/iSCSI qualified names (IQNs))	4000
Maximum number of mirroring coupling (number of mirrors)	>16000
Maximum number of mirrors on remote machine	>16000
Maximum number of remote targets	4

iSCSI connections

When shipped, the XIV Storage System is by default equipped with 6 iSCSI ports (assuming a fully populated 15 Module rack with Interface Modules 7-9 providing the 6 ports).

The external (client-provided) ethernet cables are plugged into the patch panel. For planning purposes, Table 3-3 highlights the maximum values for various iSCSI parameters for your consideration. These values are correct at the time of writing this book for Release 10.1 of the IBM XIV Storage System software.

As with Fibre Channel, it is important to plan your connectivity based on these maximums.

Table 3-3 Maximum values in context of iSCSI connectivity

iSCSI Parameters	Maximum values
Maximum number of Interface Modules with iSCSI ports	3
Maximum number of 1GB iSCSI ports per Interface Module	2
Maximum queue depth per iSCSI host port	1400
Maximum queue depth per mapped volume per host port/ target port/volume ordered set	256
Maximum iSCSI ports for any connection (host or XDRP)	6
Maximum number of hosts (iSCSI qualified names (IQNs))	4000
Maximum number of mirroring coupling (number of mirrors)	>16000
Maximum number of mirrors on remote machine	>16000
Maximum number of remote targets	4

Redundant configuration

To configure the Fibre Channel connections (SAN) for high availability, refer to the configuration illustrated in Figure 3-19. This configuration is highly recommended for all production systems to maintain system access and operations following a single hardware element or SAN component failure.

Note: The number of connections depicted was minimized for picture clarity. In principle, you should have connections to all Interface Modules in the configuration (see Figure 6-1 on page 185).

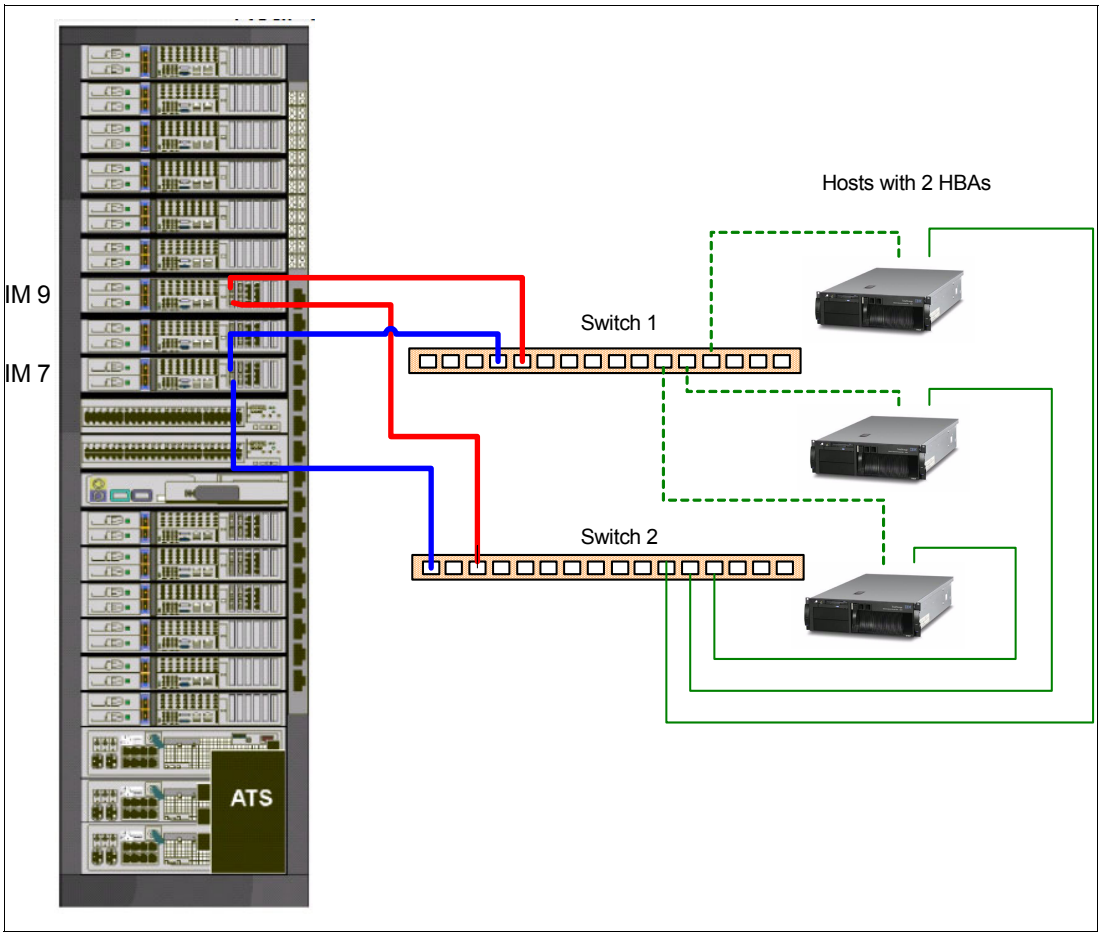


Figure 3-19 Redundant configuration

For a redundant Fibre Channel configuration, use the following guidelines:

- Each XIV Storage System Interface Module is connected to two Fibre Channel switches, using two ports of the module. One patch panel connection to each Fibre Channel switch.
- Each host is connected to two switches using two host bus adapters or a host bus adapter (HBA) with two ports.

This configuration assures full connectivity and no single point of failure:

- Switch failure: Each host remains connected to all modules through the second switch.
- Module failure: Each host remains connected to the other modules.
- Cable failure: Each host remains connected through the second physical link.
- Host HBA failure: Each host remains connected through the second physical link.

Single switch solution

This configuration is resilient to the failures of a single Interface Module, host bus adapter, and cables, however in this configuration the switch represents a single point of failure. If the switch goes down due to a hardware failure or simply because of a software update, the connected hosts will lose all data access. Figure 3-20 depicts this configuration option.

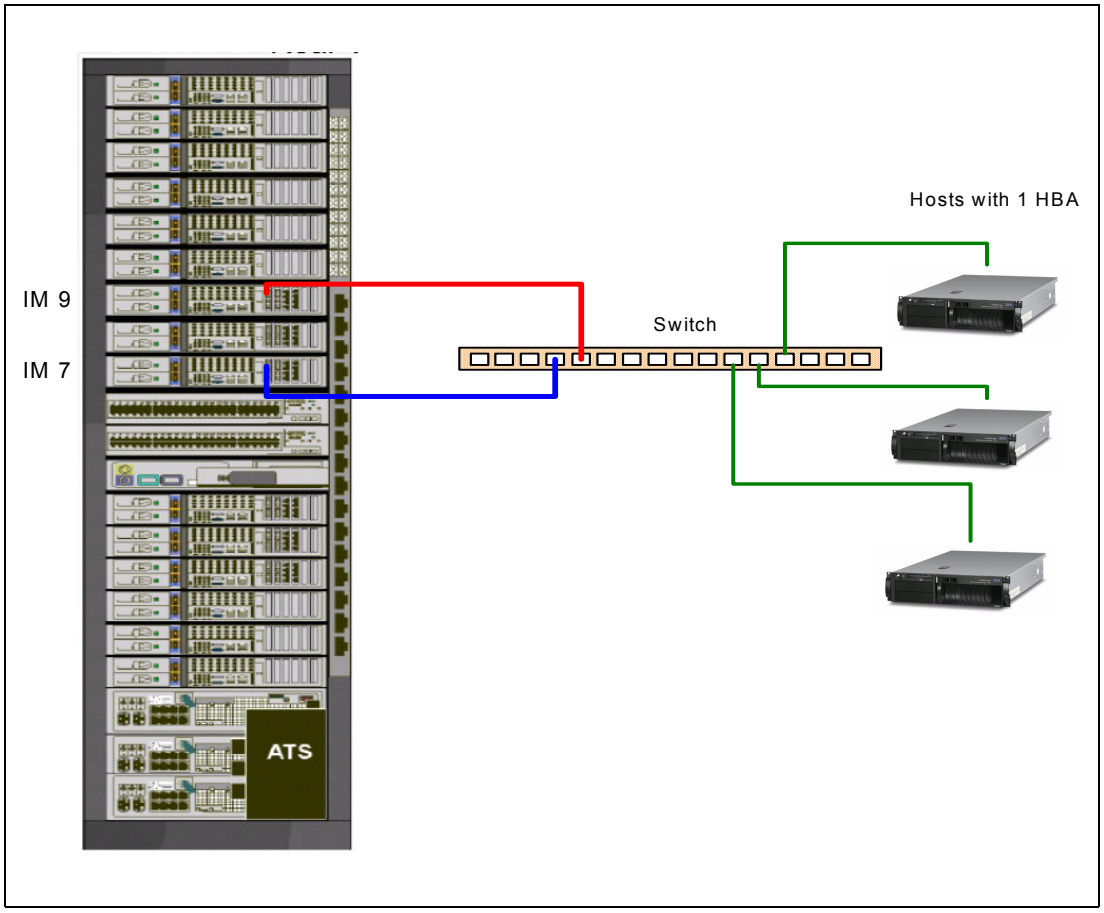


Figure 3-20 Non-redundant configuration

Only use a single switch solution when no second switch is available and/or for test environments only.

Single HBA host connectivity

Hosts that are equipped with a single Fibre Channel port can only access one switch. Therefore, this configuration is resilient to the failure of an individual Interface Module, but there are several possible points of failure (switch, cable, and HBA), which can cause access loss from the host to the IBM XIV. This configuration, which is not recommended for any production system, must be used if there is no way of adding a second Fibre Channel port to the host.

Restriction: Direct host to XIV connectivity is ***not*** permitted. Implementation must make use of a SAN fabric (either single or dual SAN switches), and dual fabric configurations are recommended.

Fibre Channel cabling and configuration

Fibre Channel cabling must be prepared based on the required fibre length and depending on the selected configuration.

When installing an XIV Storage System, perform the following Fibre Channel configuration procedures:

- You must configure Fibre Channel switches zoned correctly allowing access between the hosts and the XIV Storage System. The specific configuration to follow depends on the specific Fibre Channel switch.
- Hosts need to be set up and configured with the appropriate multipathing software to balance the load over several paths. For multipathing software and setup, refer to the specific operating system section in Chapter 6, “Host connectivity” on page 183.

iSCSI network configurations

Logical network configurations for iSCSI are equivalent to the logical configurations that are suggested for Fibre Channel networks. Four options are available:

- Redundant Configuration: Each module connects through two ports to two Ethernet switches, and each host is connected to the two switches. This design provides a network architecture resilient to a failure of any individual network switch or module.
- Single switch configuration: A single switch interconnects all modules and hosts.
- Single port host solution: Each host connects to a single switch, and a switch is connected to two modules.

IP configuration

The configuration of the XIV Storage System iSCSI connection is highly dependent on your network. In the high availability configuration, the two client-provided Ethernet switches used for redundancy can be configured as either two IP subnets or as part of the same subnet. The XIV Storage System iSCSI configuration must match the client’s network. You must provide the following configuration information for each Ethernet port:

- IP address
- Net mask
- MTU (optional):

  Maximum Transmission Unit (MTU) configuration is required if your network supports an MTU, which is larger than the standard one. The largest possible MTU must be specified (we advise you to use up to 9 000 bytes, if supported by the switches and routers). If the iSCSI hosts reside on a different subnet than the XIV Storage System, a default IP gateway per port must be specified.
- Default gateway (optional):

  Because XIV Storage System always acts as a TCP server for iSCSI connections, packets are always routed through the Ethernet port from which the iSCSI connection was initiated. The default gateways are required only if the hosts do not reside on the same layer-2 subnet as the XIV Storage System.

The IP network configuration must be ready to ensure connectivity between the XIV Storage System and the host prior to the physical system installation:

- Ethernet Virtual Local Area Networks (VLANs), if required, must be configured correctly to enable access between hosts and the XIV Storage System.
- IP routers (if present) must be configured correctly to enable access between hosts and the XIV Storage System.

Mixed iSCSI and Fibre Channel host access

IBM XIV Storage System supports mixed concurrent access from the same host to the same volumes through FC and iSCSI. When building this type of a topology, you must plan carefully to properly ensure redundancy and load balancing.

Note: Not all hosts support multi-path configurations between the two protocols, FCP and iSCSI. We highly recommend that you contact the your IBM Representative for help in planning configurations that include mixed iSCSI and Fibre Channel host access.

Management connectivity

IBM XIV Storage System is managed through three IP addresses over Ethernet interfaces on the patch panel in order to be resilient to two hardware failures. Thus, you must have three Ethernet ports available for management. If you require management to be resilient to a single network failure, we recommend that you connect these ports to two switches.

Make sure as well that the networking equipment providing the management communication is protected by an Uninterruptible Power Supply (UPS).

Management IP configurations

For each of the three management ports, you must provide the following configuration information to the IBM SSR upon installation (refer also to 3.3.3, “Basic configuration planning” on page 64):

- IP address of the port
- Subnet mask
- Default IP gateway (if required)

The following system-level IP information should be provided (not port-specific):

- IP address of the primary and secondary DNS servers
- IP address or DNS names of the SNMP manager, if required
- IP address or DNS names of the Simple Mail Transfer Protocol (SMTP) servers

Protocols

The XIV Storage System is managed through dedicated management ports running TCP/IP over Ethernet. Management is carried out through the following protocols (consider this design when configuring firewalls, other security protocols, and SMTP relaying):

- Proprietary XIV protocols are used to manage XIV Storage System from the GUI and the XCLI. This management communication is performed over TCP port 7778, where the GUI/XCLI, as the client, always initiates the connection, and the XIV Storage System performs as the server.
- XIV Storage System sends and responds to SNMP management packets.
- XIV Storage System initiates SNMP packets when sending traps to SNMP managers.
- XIV Storage System initiates SMTP traffic when sending e-mails (for either event notification through e-mail or for e-mail-to-SMS gateways).
- XIV Storage System communicates with remote SSH connections over standard TCP port 22.

SMTP server

For proper operation of the XIV Call Home function, the SMTP server must:

- Be reachable on port 25 for the XIV customer specified management IP addresses.
- Permit relaying from the XIV customer specified management IP addresses

- Permit the XIV system to send e-mails using the fromxiv@il.ibm.com
- Permit recipient addresses of xiv-callhome-western-hemisphere@vnet.ibm.com and xiv-callhome-eastern-hemisphere@vnet.ibm.com

Mobile computer ports

The XIV Storage System has two Ethernet mobile computer ports. A single mobile computer, or other computer, can be connected to these ports. When connected, the system will serve as a Dynamic Host Configuration Protocol (DHCP) server and will automatically configure the mobile computer.

Restriction: Do not connect these ports to the user (client) network.

XIV Remote Support Center (XRSC)

To facilitate remote support by IBM XIV personnel, we recommend that you configure a dedicated Ethernet port for remote access. This port must be connected through the organizational firewall so that IBM XIV personnel can access the XIV Storage System, if required.

The XIV Remote Support Center comprises XIV internal functionality together with a set of globally deployed supporting servers to provide secure IBM support access to the XIV system when necessary and when authorized by the customer personnel. Figure 3-21 provides an representation of the data flow of the XIV to IBM Support.

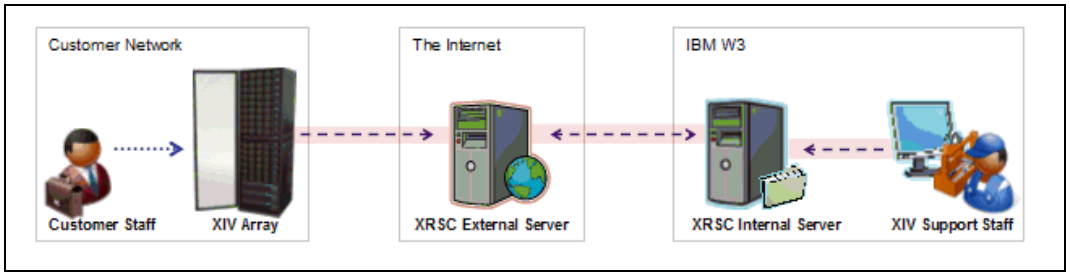


Figure 3-21 XIV Remote Support Center

To initiate the remote connection process, the following steps are performed:

1. Customer initiates an Internet based SSH connection to XRSC either via the GUI or XCLI.
2. XRSC identifies the XIV Storage System and marks it as “connected”.
3. Support personnel connects to XRSC using SSH over the IBM Intranet.
4. XRSC authenticates the support person against the IBM Intranet.
5. XRSC then displays the connected customer system available to the support personnel.
6. The IBM Support person then chooses which system to support and connect to:
   - Only permitted XIV systems are shown
   - IBM Support personnel log their intended activity
7. A fully recorded support session starts.
8. When complete, the support person terminates the session and the XRSC disconnects the XIV array from the remote support system.

The XRSC Internet servers are hard coded in XIV Software and no further configuration is required by the customer to enable this function aside from turning this feature on in the GUI

or XCLI. This provides an expedient manner for IBM support to gather required information from the system in the most timely fashion and with the least impact to the customer.

Remote mirroring connectivity

Planning the physical connections also includes considerations when the IBM XIV is installed in a Remote Copy environment. We recommend that you contact advanced IBM XIV support for assistance for planning Remote Mirroring connectivity to assure the maximum resilience to hardware failures and connection failures.

Remote Copy links, which connect the direct primary system and secondary system, need to also be planned for prior to the physical installation. The physical Remote Copy links can be Fibre Channel links, direct or through a SAN, or iSCSI port connections using ethernet however iSCSI is not recommended for this use.

Planning for growth

Ensure consideration for growth and for the future IO demands of your business. Most applications and databases grow quickly and the need for greater storage capacity increases rapidly. Planning for growth prior to the implementation of the first IBM XIV in the environment can save time and re-configuring effort in the future.

There is also a statement of general direction that the IBM XIV will be available in the future as a multiple frame machine. Therefore, install the first IBM XIV in a place with sufficient space next to the rack for the possibility of expanding the XIV footprint.

3.3.4 IBM XIV physical installation

After all previous planning steps are completed and the machine is delivered to its final location, the physical installation can begin. A IBM SSR will perform all the necessary tasks and perform the first logical configuration steps up to the point where you can connect the IBM XIV through the GUI and the XCLI. Configuring Storage Pools, logical unit numbers (LUNs), and attaching the IBM XIV to the host are storage administrator responsibilities. Refer to Chapter 4, “Configuration” on page 79.

Physical installation

It is the responsibility of the customer or moving contractor to unpack and move the IBM XIV Storage System as close as possible to its final destination before an IBM SSR can start the physical installation. Carefully check and inspect the delivered crate and hardware for any visible damage. If there is no visible damage, and the tilt and shock indicators show no problem, sign for the delivery.

Also arrange that with the start, or during the physical installation, an electrician is available who is able to handle the power requirements in the environment up to the IBM XIV power connectors.

The physical installation steps are as follows:

1. Place and adjust the rack in its final position in the data centre.
2. Check the IBM XIV hardware. When the machine is delivered with the weight reduction feature (FC 0200), the IBM SSR will install the removed modules and components into the rack.
3. Connect the IBM XIV line cords to the client-provided power source and advise an electrician to switch on the power connections.

4. Perform the initial power-on of the machine and perform necessary checks according to the given power-on procedure.
5. To complete the physical steps of the installation, the IBM SSR will perform several final checks of the hardware before continuing with the basic configuration.

Basic configuration

After the completion of the physical installation steps, the IBM SSR establishes a connection to the IBM XIV through the patch panel (refer to 3.2.4, “Patch panel” on page 58) and completes the initial setup. You must provide the required completed information sheet that is referenced in 3.3.3, “Basic configuration planning” on page 64.

The basic configuration steps are as follows:

1. Set the Management IP Addresses, (Client Network) Gateway, and Netmask.
2. Set the system name.
3. Set the e-mail sender address and SMTP server address.
4. Set the primary DNS and the secondary DNS.
5. Set the SNMP management server address.
6. Set the time zone.
7. Set the NTP server address
8. Configure the system to send events to IBM (Call Home)

Complete the physical installation

After the IBM SSR completes the physical installation and initial setup, the IBM SSR performs the final checks for the IBM XIV:

- Power off and power on the machine by using the XIV Storage Management (GUI) and XCLI.
- Check the Events carefully for problems.
- Verify that all settings are correct and persistent.

At this point, the installation is complete, and the IBM XIV Storage System is ready to be handed over to the customer to configure and use. Refer to Chapter 4, “Configuration” on page 79.

3.3.5 System power-on and power-off

Strictly follow these procedures to power on and power off your XIV system.

Power-on

To power on the system:

1. On each UPS, look for the Test button located on the Control Panel (front of the UPS) as illustrated in Figure 3-22.

Important: Do not confuse the Test button with the Power Off button, which is normally protected by a transparent cover. The Test button is the one circled in red in Figure 3-22.

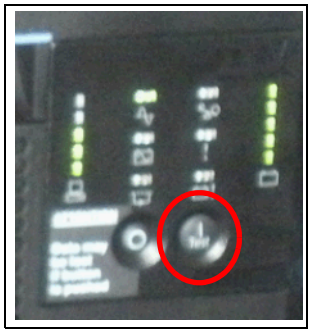


Figure 3-22 Locate Test button

2. Use both hands as shown in Figure 3-23 to press each of the three Test buttons simultaneously.

Figure 3-23 Use both hands to hit the three Test buttons simultaneously

This will start applying power to the rack, and all the modules and initiate the boot process for the interface modules and data modules

Power-off

Powering the system off must be done solely from either the XIV GUI or the XCLI. You must be logged on as Storage Administrator (storageadmin role).

Warning: Do not power off the XIV using the UPS power button because this can result in the loss of data and system configuration information.

Using the GUI

From the XIV GUI:

1. Simply click the Shutdown icon available from the system main window toolbar, as illustrated in Figure 3-24.

Figure 3-24 System shutdown

2. You will have to confirm twice as shown in Figure 3-25.

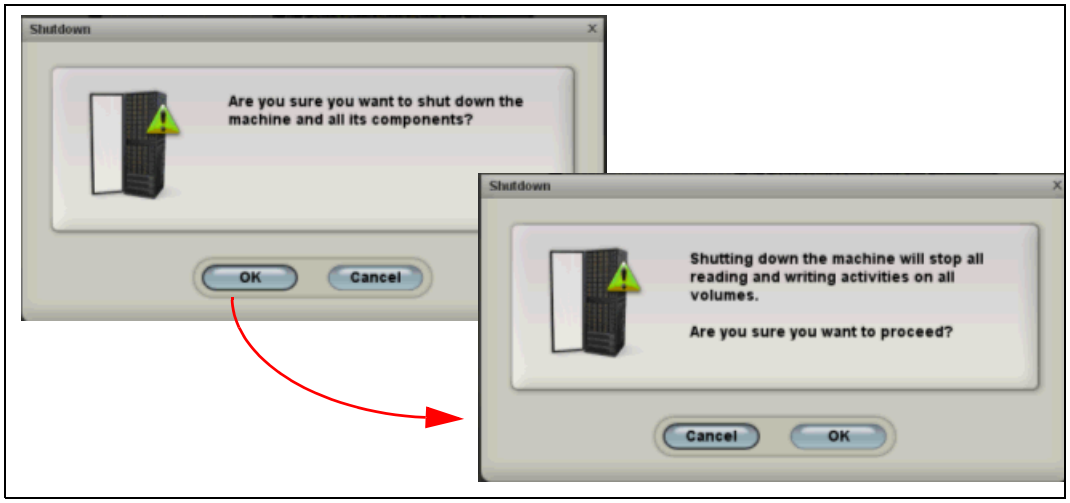


Figure 3-25 Confirm system shutdown

The shutdown takes 2-3 minutes. When done all fans and front lights on modules are off, while the UPS lights stay on.

Tip: Using the GUI is the most convenient and recommended way to power off the system.

Using the XCLI

From the command prompt, issue the command:

```
xcli -c "XIV MN00035" shutdown

where XIV MN00035 is the system name
```

Or, if using the XCLI session, simply enter the **shutdown** command as shown in Example 3-1.

Example 3-1 Executing a shutdown from the XCLI session

```
User Name: admin
Password: **********
Machine IP/Hostname: 9.11.237.119
connecting..
>> shutdown
Warning:   ARE_YOU_SURE_YOU_WANT_TO_SHUT_DOWN Y/N:
Command executed successfully
```

The shutdown takes 2-3 minutes. When done, all fans and front lights on modules are off, while the UPS lights stay on.

Warning: Do not power off the XIV using the UPS power button because this can result in the loss of data and system configuration information.

4

Configuration

This chapter discusses the tasks to be performed by the storage administrator to configure the XIV Storage System using the XIV Management Software.

We provide step-by-step instructions covering the following topics, in this order:

- Install and customize the XIV Management Software
- Connect to and manage XIV using graphical and command line interfaces
- Organize system capacity by Storage Pools
- Create and manage volumes in the system
- Host definitions and mappings
- XIV XCLI scripts

4.1 IBM XIV Storage Management software

The XIV Storage System software supports the functions of the XIV Storage System. The software provides the functional capabilities of the system. It is preloaded on each module (Data and Interface Modules) within the XIV Storage System. The functions and nature of this software are equivalent to what is usually referred to as microcode or firmware on other storage systems.

The XIV Storage Management software is used to communicate with the XIV Storage System Software, which in turn interacts with the XIV Storage hardware.

The XIV Storage Manager can be installed on a Windows, Linux, AIX, HPUX, or Solaris workstation that will then act as the management console for the XIV Storage System. The Storage Manager software is provided at the time of installation, or is optionally downloadable from the following Web site:

http://www.ibm.com/systems/support/storage/XIV

For detailed information about the XIV Storage Management software compatibility, refer to the XIV interoperability matrix or the System Storage Interoperability Center (SSIC) at:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

4.1.1 XIV Storage Management user interfaces

The IBM XIV Storage Manager includes a user-friendly and intuitive Graphical User Interface (GUI) application, as well as an Extended Command Line Interface (XCLI) component offering a comprehensive set of commands to configure and monitor the system.

Graphical User Interface (GUI)

A simple and intuitive GUI lets you perform most administrative and technical operations (depending upon the user role) quickly and easily, with minimal training and knowledge.

The main motivation behind the XIV management and GUI design is the desire to eliminate the complexities of system management. The most important operational challenges, such as overall configuration changes, volume creation or deletion, snapshot definitions, and many more, are achieved with a few clicks.

This chapter contains descriptions and illustrations of tasks performed by a storage administrator when using the XIV graphical user interface (GUI) to interact with the system.

Extended Command Line Interface (XCLI)

The XIV Extended Command Line Interface (XCLI) is a powerful text-based, command line-based tool that enables an administrator to issue simple commands to configure, manage, or maintain the system, including the definitions required to connect to hosts and applications. The XCLI tool can be used in an XCLI Session environment to interactively configure the system or as part of a script to perform lengthy and more complex tasks.

Tip: Any operation that can be performed via the XIV GUI is also supported by the XIV Extended Command Line Interface (XCLI).

This chapter presents the most common XCLI commands and tasks normally used by the administrator to interact with the system.

4.1.2 XIV Storage Management software installation

This section illustrates the step-by-step installation of the XIV Storage Management software under Microsoft Windows XP.

The GUI is also supported on the following platforms:

- Microsoft Windows 2000, Windows ME, Windows XP, Windows Server 2003, and Windows Vista
- Linux (Red Hat 5.x or equivalent)
- AIX 5.3, AIX 6
- Solaris v9, Solaris v10
- HPUX 11i v2, HPUX 11i v3

The GUI can be downloaded at: `ftp://ftp.software.ibm.com/storage/XIV/GUI/`

It also contains a demo mode. To use the demo mode, log on as user `P10DemoMode` and no password.

Important: Minimum requirements for installation in Windows XP:

CPU:	Double Core or equivalent
Memory:	1024 MB
Graphic Memory:	128 MB
Disk capacity:	100 MB Free
Supported OS:	Windows 2000 Server, ME, XP, Windows Server 2003, Windows Vista
Screen resolution:	1024x768 (recommended) to 1600x1200
Graphics:	24/32 True Color Recommended

At the time of writing, the XIV Storage Manager Version 2.4 was available, and later GUI releases might slightly differ in appearance.

Perform the following steps to install the XIV Storage Management software:

1. Locate the XIV Storage Manager installation file (either on the installation CD or a copy you downloaded from the Internet). Running the installation file first shows the welcome window displayed in Figure 4-1. Click **Next**.

Figure 4-1 Installation: Welcome window

2. A Setup dialog window is displayed (Figure 4-2) where you can specify the installation directory. Keep the default installation folder or change it accordingly to your needs. When done, click **Next**.

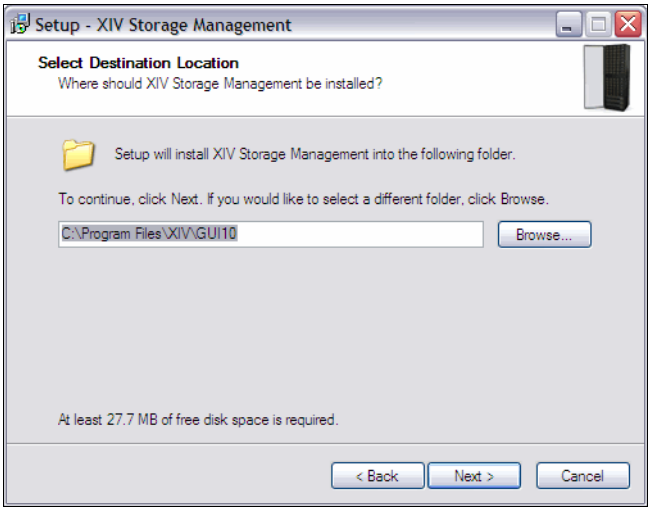


Figure 4-2 Specify the installation directory

3. The next installation dialog is displayed. You can choose between a FULL installation method or just a command line interface installation method. We recommend that you choose **FULL installation** as shown in Figure 4-3. In this case, the Graphical User Interface and the Command Line Interface as well will be installed. Click **Next**.

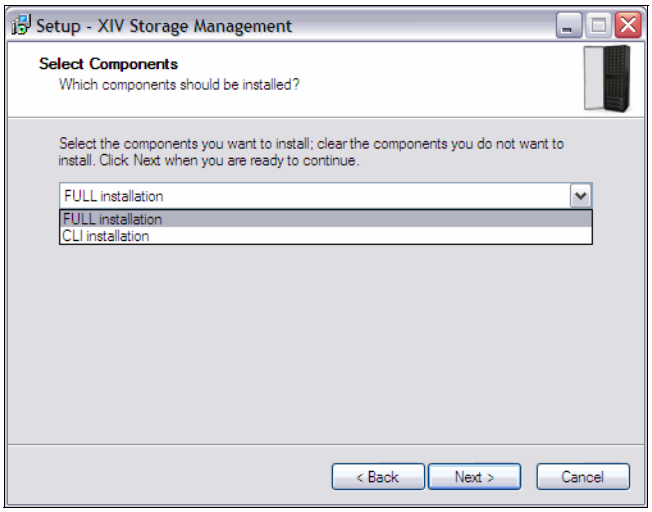


Figure 4-3 Choose the installation type

4. The next step is to specify the Start Menu Folder as shown in Figure 4-4. When done, click **Next**.

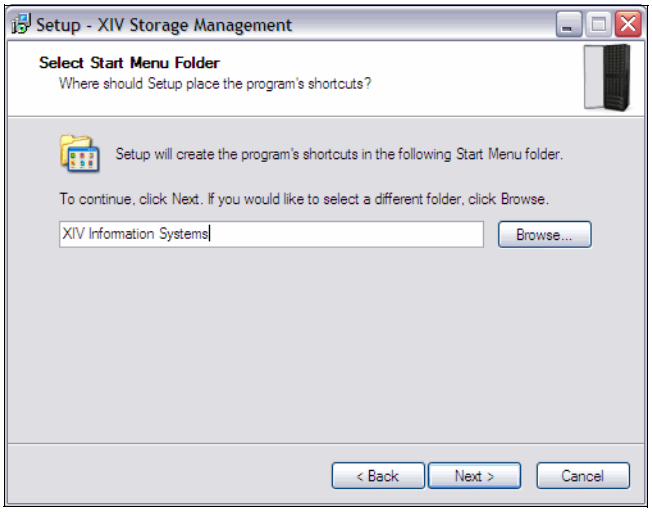


Figure 4-4 Select Start Menu Folder

5. The dialog shown in Figure 4-5 is displayed. Select the desktop icon placement and click **Next**.

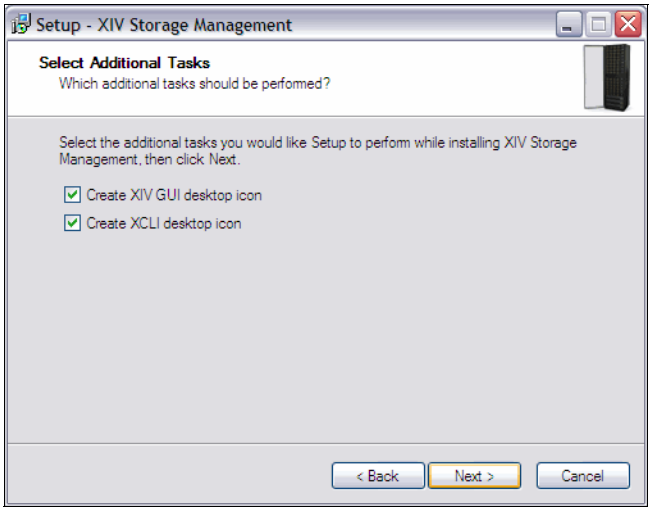


Figure 4-5 Select additional tasks

6. The dialog window shown in Figure 4-6 is displayed. The XIV Storage Manager requires the Java Runtime Environment Version 6, which will be installed during the setup if needed. Click **Finish**.

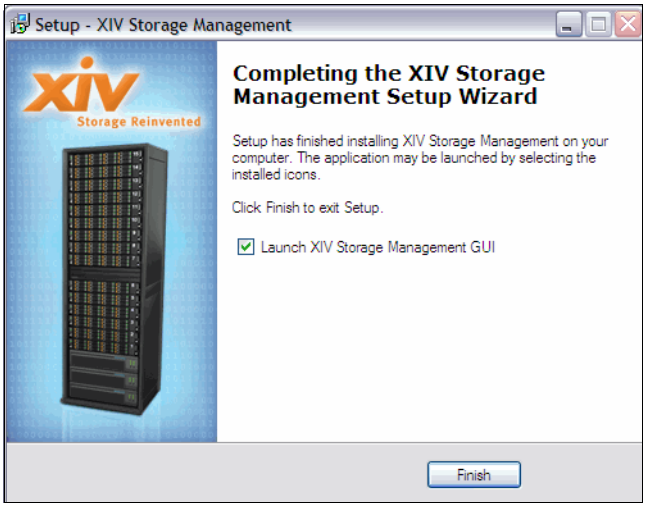


Figure 4-6 Completing setup

If the computer on which the XIV GUI is installed is connected to the Internet, a window might appear to inform you that a new software upgrade is available. Click **OK** to download and install the new upgrade, which normally only requires a few minutes and will not interfere with your current settings or events.

4.2 Managing the XIV Storage System

The basic storage system configuration sequence followed in this chapter includes the initial installation steps, followed by disk space definition and management.

Additional configuration or advanced management tasks are cross-referenced to specific chapters where they are discussed in more detail. For example, allocating and mapping volumes to hosts is covered in Chapter 6, “Host connectivity” on page 183.

Figure 4-7 presents an overview of the configuration flow. Note that the XIV GUI is extremely intuitive, and you can easily and quickly achieve most configuration tasks.

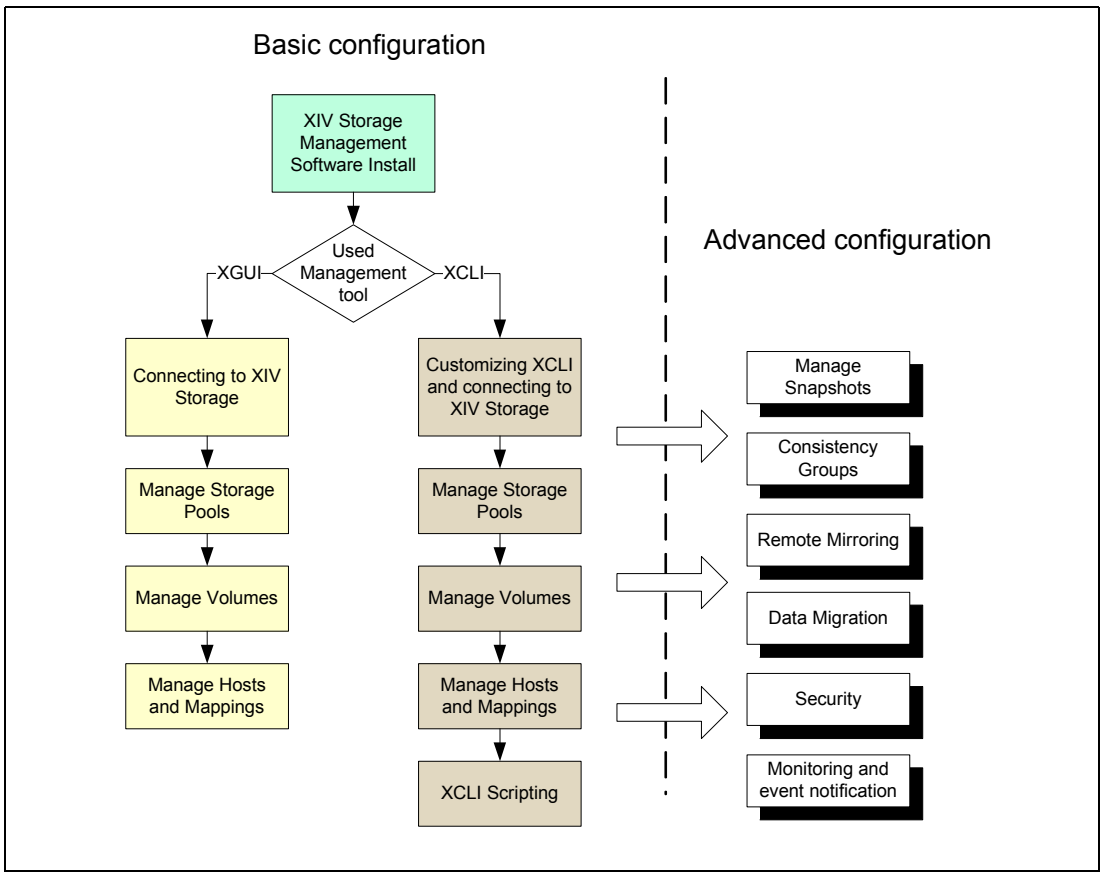


Figure 4-7 Basic configuration sequence and advanced tasks

After the installation and customization of the XIV Management Software on a Windows, Linux, AIX, HPUX, or Solaris workstation, a physical Ethernet connection must be established to the XIV Storage System itself.

The Management workstation is used to:

- Execute commands through the XCLI interface (see 4.2.2, “Log on to the system with XCLI” on page 93).
- Control the XIV Storage System through the GUI.
- Configure e-mail notification messages and Simple Network Management Protocol (SNMP) traps upon occurrence of specific events or alerts. See Chapter 14, “Monitoring” on page 313.

To ensure management redundancy in case of Interface Module failure, the XIV Storage System management functionality is accessible from three IP addresses. Each of the three IP addresses is linked to a different (hardware) Interface Module. The various IP addresses are transparent to the user, and management functions can be performed through any of the IP addresses. These addresses can also be used simultaneously for access by multiple management clients. Users only need to configure the GUI or XCLI for the set of IP addresses that are defined for the specific system.

Notes: All management IP interfaces must be on the same subnet and use the same:

- Network mask
- Gateway
- Maximum Transmission Unit (MTU)

Both XCLI and GUI management run over TCP port 7778 with all traffic encrypted through the Secure Sockets Layer (SSL).

4.2.1 The XIV Storage Management GUI

This section reviews the XIV Storage Management GUI.

Launching the XIV Storage Management GUI

Upon launching the XIV GUI application, a login window prompts you for a user name and its corresponding password before granting access to the XIV Storage System. The default user is "*admin*" and the default corresponding password is "*adminadmin*", as shown in Figure 4-8.

Important: Remember to change the default passwords to properly secure your system.

The default admin user comes with storage administrator (storageadmin) rights. The XIV Storage System offers role-based user access management.

For more information about user security and roles, refer to Chapter 5, "Security" on page 121.

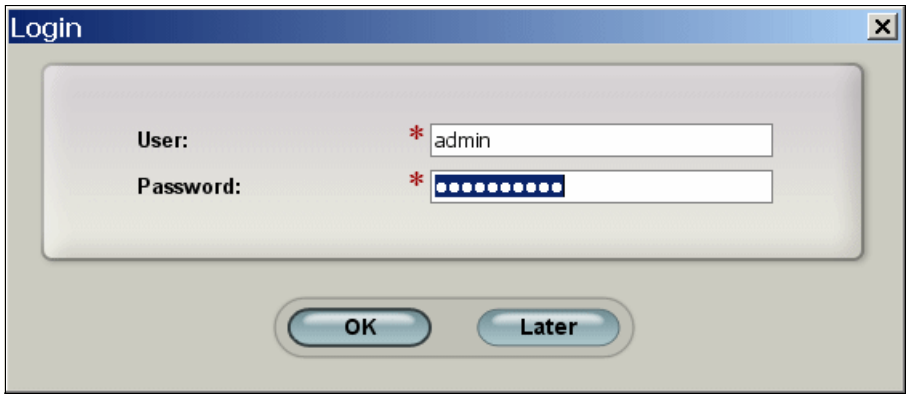


Figure 4-8 Login window with default access

To connect to an XIV Storage System, you must initially add the system to make it visible in the GUI by specifying its IP addresses.

To add the system:

1. Make sure that the management workstation is set up to have access to the LAN subnet where the XIV Storage System resides. Verify the connection by pinging the IP address of the XIV Storage System.

   If this is the first time you start the GUI on this management workstation and no XIV Storage System had been previously defined to the GUI, the Add System Management dialog window is automatically displayed:

   – If the default IP address of the XIV Storage System was not changed, check **Connect Directly**, which populates the IP/DNS Address1 field with the default IP address. Click **Add** to effectively add the system to the GUI.

   – If the default IP address had already been changed to a client-specified IP address (or set of IP addresses, for redundancy), you must enter those addresses in the IP/DNS Address fields. Click **Add** to effectively add the system to the GUI. Refer to Figure 4-9.

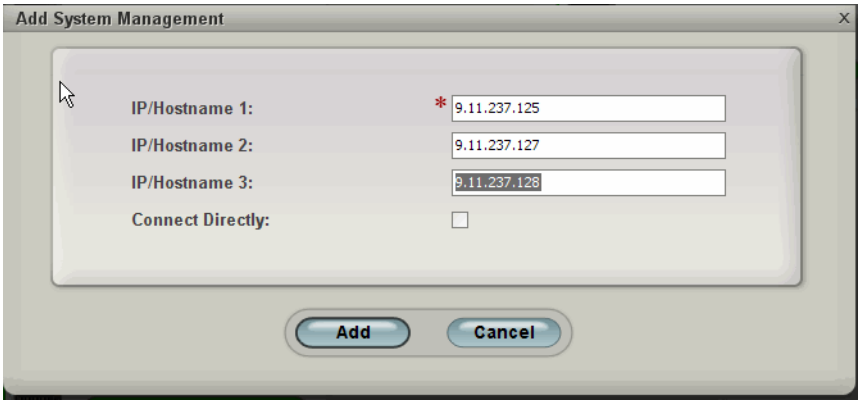


Figure 4-9 Add System Management

2. You are now returned to the main XIV Management window. Wait until the system is displayed and shows as enabled. Under normal circumstances, the system will show a status of Full Redundancy displayed in a green label box.
3. Move the mouse cursor over the image of the XIV Storage System and click to open the XIV Storage System Management main window as shown in Figure 4-10.

XIV Storage Management GUI Main Menu

The XIV Storage Management GUI is mostly self-explanatory with a well-organized structure and simple navigation.

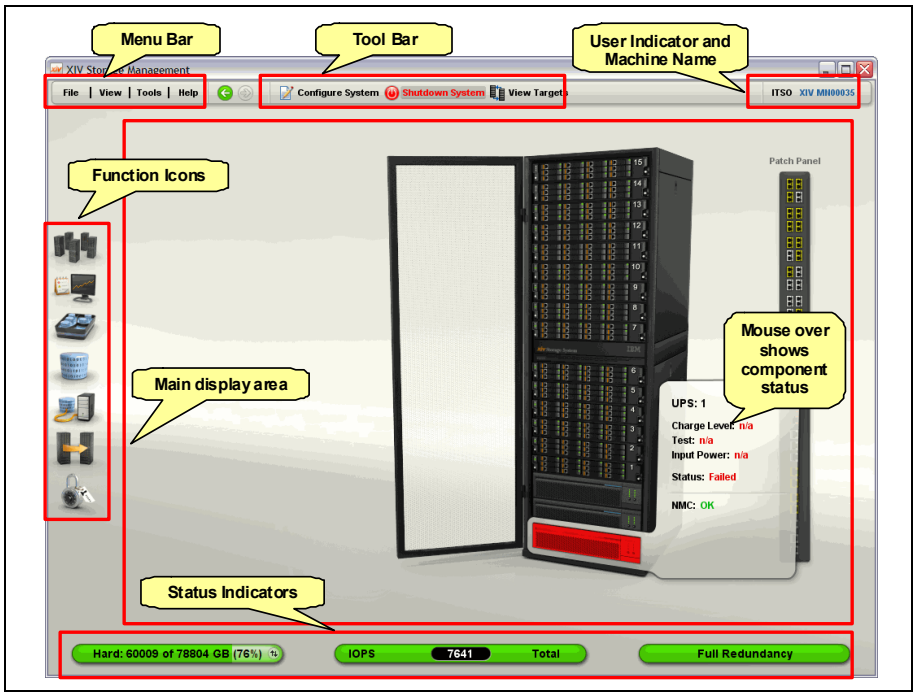


Figure 4-10 XIV Storage Manager main window: System view

The main window is divided into the following areas:

- Function icons: Located on the left side of the main window, you find a set of vertically stacked icons that are used to navigate between the functions of the GUI, according to the icon selected.

  Moving the mouse cursor over an icon brings up a corresponding option menu. The various menu options available from the function icons are presented in Figure 4-11 on page 89.

- Main display: This area occupies the major part of the window and provides graphical representation of the XIV Storage System. Moving the mouse cursor over the graphical representation of a specific hardware component—module, disk, and Uninterruptible Power Supply (UPS) unit— brings up a status callout.

  When a specific function is selected, the main display shows a tabular representation of that function.

- Menu bar: This area is used for configuring the system and as an alternative to the Function icons for accessing the various functions of the XIV Storage System.

- Toolbar: It is used to access a range of specific actions linked to the individual functions of the system.

- Status bar: These indicators are located at the bottom of the window. This area indicates the overall operational status of the XIV Storage System:

  – The first indicator on the left shows the amount of soft or hard storage capacity currently allocated to Storage Pools and provides alerts when certain capacity thresholds are reached. As the physical, or hard, capacity consumed by volumes within a Storage Pool passes certain thresholds, the color of this meter indicates that additional hard capacity might need to be added to one or more Storage Pools.

– The second indicator (in the middle) displays the number of I/O operations per second (IOPS).
– The third indicator on the far right shows the general system status and will, for example, indicate when a redistribution is underway.

Additionally, an Uncleared Event indicator is visible when events occur for which a repetitive notification was defined that has not yet been cleared in the GUI (these notifications are called *Alerting Events*).

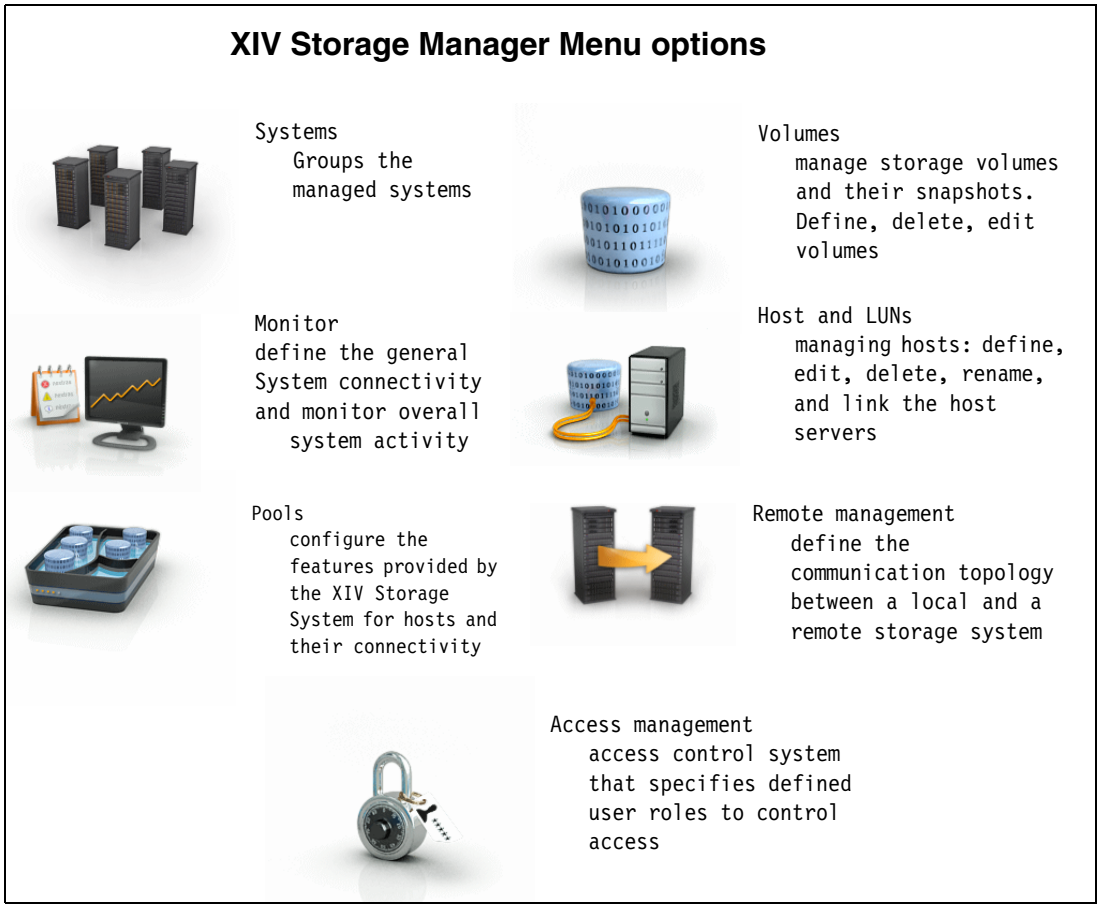


Figure 4-11 Menu items in XIV Storage Management software

Tip: The configuration information regarding the connected systems and the GUI itself is stored in various files under the user's home directory.

As a useful and convenient feature, all the commands issued from the GUI are saved in a log in the format of XCLI syntax. The syntax includes quoted strings; however, the quotes are only needed if the value specified contains blanks.

The default location is in the Documents and Setting folder of the Microsoft Windows current user, for example:

```
%HOMEDRIVE%%HOMEPATH%\Application Data\XIV\GUI10\logs\guiCommands*.log
```

XIV Storage Management GUI Features

There are several features that enhance the usability of the GUI menus. These features enable the user to quickly and easily execute tasks.

As shown in Figure 4-12, commands can be executed against multiple selected objects.

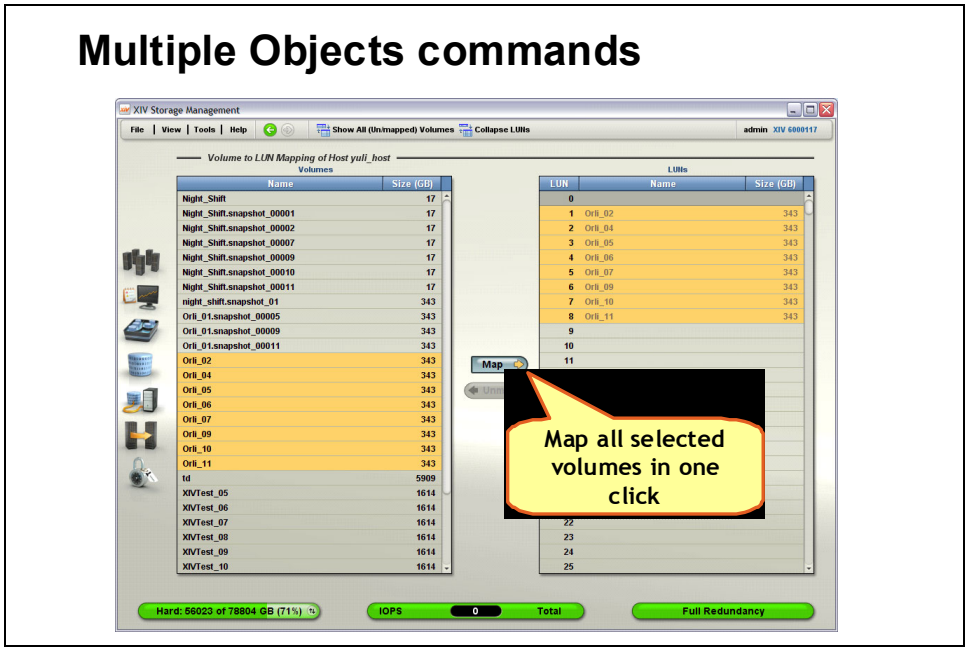


Figure 4-12 Multiple Objects Commands

As shown in Figure 4-13, menu tips are now displayed when placing the mouse cursor over greyed out menu items, explaining why they are not selectable in a given context.

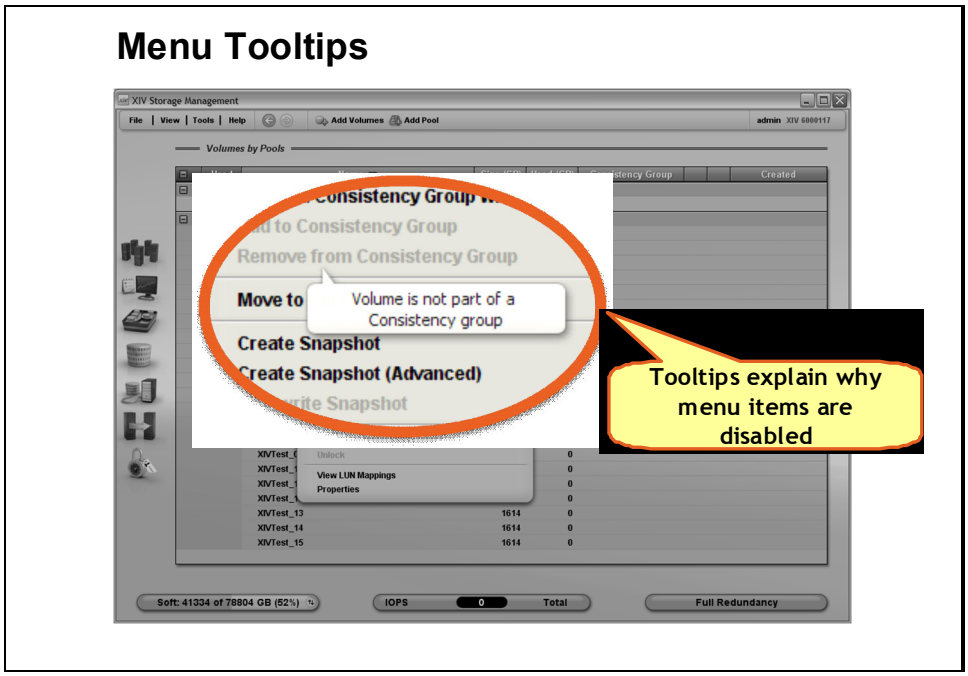


Figure 4-13 Menu Tool Tips

Figure 4-14 illustrates keyboard navigation shortcuts.

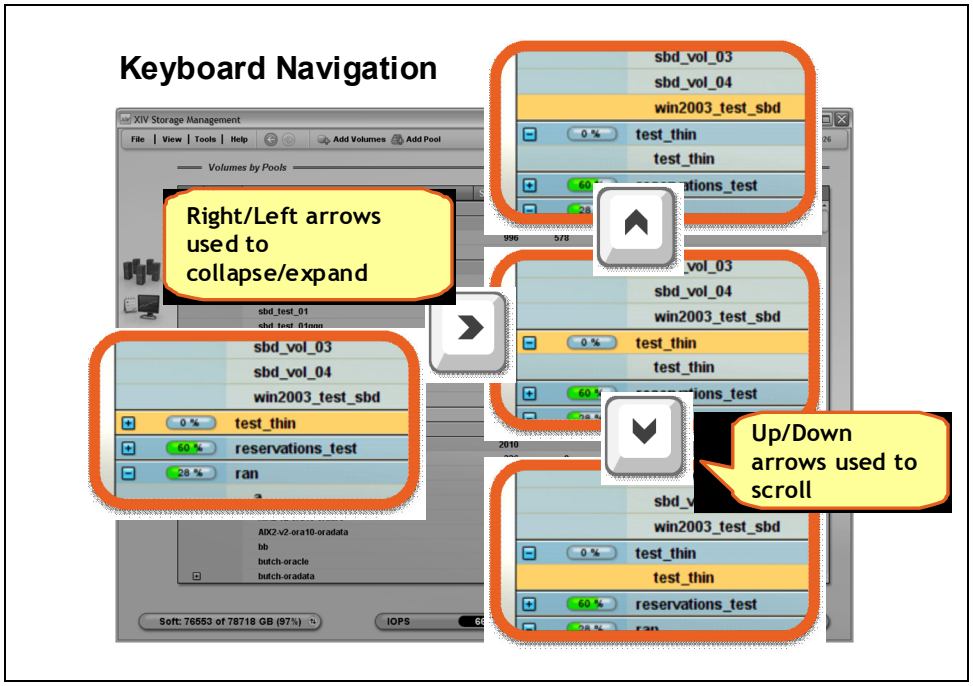


Figure 4-14 Keyboard Navigation

Figure 4-15 illustrates the use of - and + icons to collapse or expand tree views.

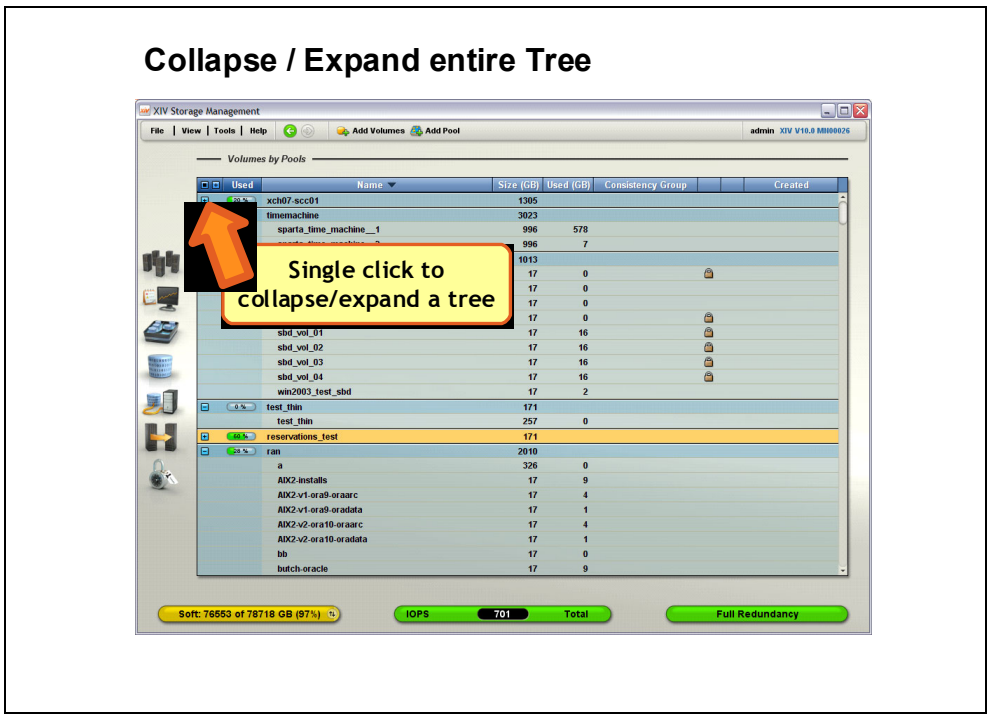


Figure 4-15 Collapse/Expand Entire Tree

In multiple system environments, the main GUI can now register and display up to 15 XIV systems. They are displayed in a matrix arrangement as seen in Figure 4-16.

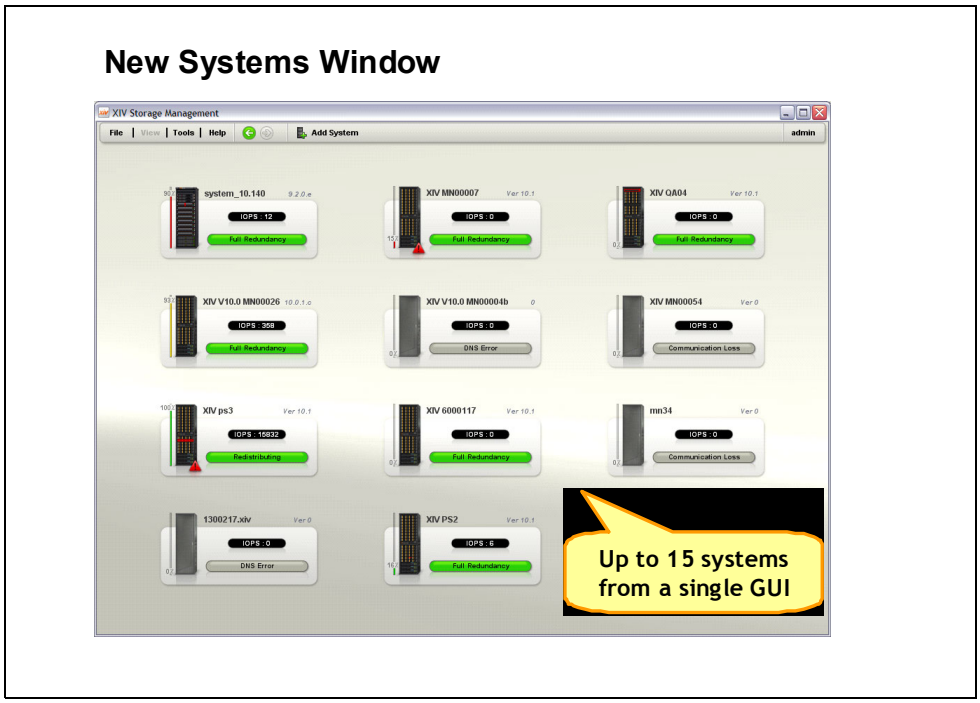


Figure 4-16 Systems window

Another convenient improvement is also the ability of the GUI to autodetect target connectivity and let the user switch between the connected systems, as illustrated in Figure 4-17.

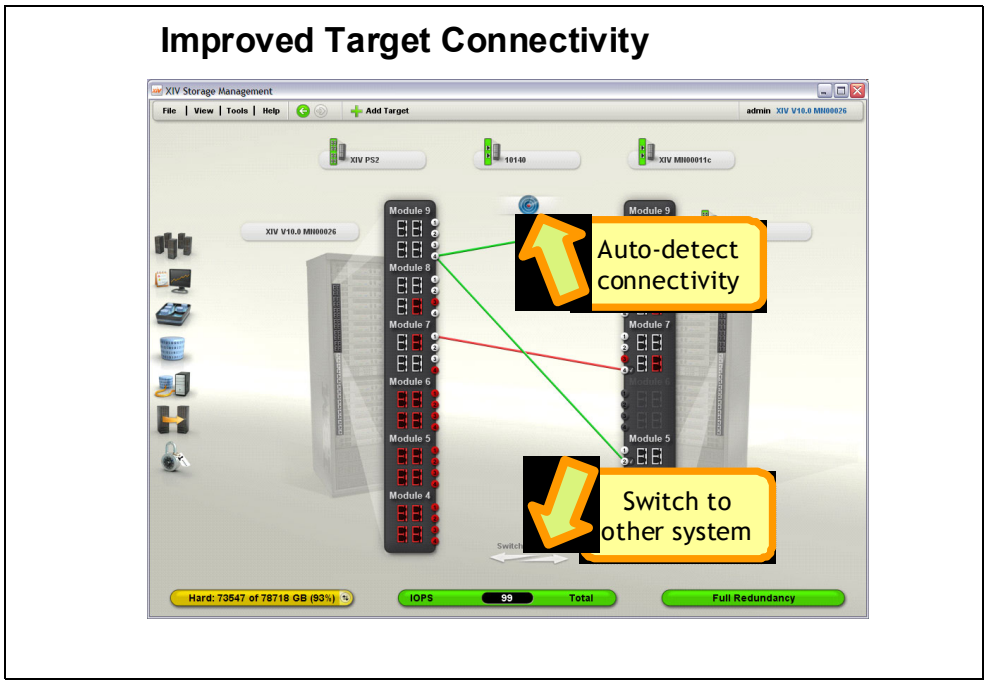


Figure 4-17 Improved target connectivity

4.2.2 Log on to the system with XCLI

After the installation of the XIV Storage Management software, you will find the XCLI Session Link on the desktop. Usage of the XCLI Session environment is simple and has an interactive help for command syntax. XCLI command examples are given in this section.

There are several methods of invoking the XCLI functions:

- *XCLI Session:* Click the desktop link or use the drop-down menu from the Systems Menu in the *GUI* (as shown in Figure 4-18). Starting from the GUI will automatically provide the current userid and password and connect to the system selected. Otherwise you will be prompted for user information and the IP address of the system.

  Tip: The XCLI Session is the easiest way to issue XCLI commands against *XIV systems*, and we recommend its use.

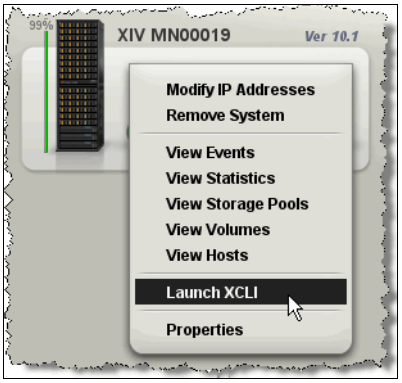


Figure 4-18 Launch XCLI from GUI

- *Invoking the XCLI in order to define configurations:* A configuration is a mapping between a user-defined name and a list of three IP addresses. This configuration can be referenced later in order to execute a command without having to specify the system IP addresses (refer to the following command execution method in this list).

  These various configurations are stored on the local host running the XCLI utility and must be defined for each host. This system name can also be used for XCLI Sessions.

- *Invoking the XCLI to execute a command:* This method is the most basic and is used in scripts that contain XCLI commands. When invoking an XCLI command directly or in a script, you must also provide either the system's IP addresses or a configuration name.

- *Invoking the XCLI for general purpose functions:* These invocations can be used to get the XCLI's software version or to print the XCLI's help text.

The command to execute is generally specified along with parameters and their values.

A *script* can be defined to specify the name and path to the commands file (lists of commands will be executed in User Mode only).

For complete and detailed documentation of the IBM XIV Storage Manager Software, refer to the *XCLI Reference Guide*, GC27-2213-00 and the *XIV Session User Guide*. These documents can be found in the IBM XIV Storage System Information Center:

http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp

XCLI Session features

XCLI Session offers command and argument completions, along with possible values for the arguments. There is no need to enter user information or IP addresses for each command:

- *Executing a command:* Simply type the command.
- *Command completion:* Type part of a command and press Tab to see possible valid commands.
- *Command argument completion:* Type a command and press Tab to see a list of values for the command argument.

```
>> user_ <TAB>
user_define        user_delete     user_group_add_user    user_group_create
user_group_delete  user_group_list user_group_remove_user user_group_rename
user_group_update  user_list       user_rename            user_update

>> user_list <TAB>
show_users=   user=

>> user_list user= <TAB>
xiv_development   xiv_maintenance   admin    technician
smis_user         ITSO

>> user_list user=admin
Name   Category     Group Active Email Address Area Code Phone Number Access All
admin  storageadmin       yes
```

Figure 4-19 XCLI Session example

Customizing the XCLI environment

For convenience and more efficiency in using the XCLI, we recommend that you use the XCLI Session environment and invoke XCLI Session from the GUI menu. However, if you want to write scripts to execute XCLI commands, it is possible to customize your management workstation environment as described next.

As part of XIV's high-availability features, each system is assigned three IP addresses. When executing a command, the XCLI utility is provided with these three IP addresses and tries each of them sequentially until communication with one of the IP addresses is successful. You must pass at least one of the IP addresses (IP1, IP2, and IP3) with each command.

The default IP address for XIV is 14.10.202.250. To avoid too much typing and having to remember IP addresses, you can use a predefined configuration name. By default, XCLI uses the system configurations defined when adding systems to the XIV GUI. To list the current configurations, use the command shown in Example 4-1.

Example 4-1 List Configurations XCLI command

```
C:\> xcli -L

System       Managements IPs
XIV MN00035  9.11.237.125, 9.11.237.127, 9.11.237.128
XIV MN00019  9.11.237.107, 9.11.237.108, 9.11.237.106
```

Note: When executing a command, you must specify either a configuration or IP addresses, but not both.

To issue a command against a specific XIV Storage System, you also need to supply the username and the password for it. The default user is `admin` and the default password is `adminadmin`, which can be used with the following parameters:

`-u user or -user`

- This sets the user name that will be used to execute the command.

`-p password or -password`

- This is the XCLI password that must be specified in order to execute a command in the system.

`-m IP1 [-m IP2 [-m IP3]]`

- This defines the IP addresses of the Nextra™ system.

Example 4-2 illustrates a common command execution syntax on a given XIV Storage System.

Example 4-2 Simple XCLI command

```
xcli -u admin -p adminadmin -m 9.11.237.125 user_list
```

Managing the XIV Storage System by using the XCLI always requires that you specify these same parameters. To avoid repetitive typing, you can instead define and use specific environment variables. We recommend that you create a batch file in which you set the value for these specific environment variables, as shown in Example 4-3.

Example 4-3 Script file setup commands

```
@echo off
set XIV_XCLIUSER=admin
set XIV_XCLIPASSWORD=adminadmin
REM add the following command only to change the default xiv-systems.xml file
REM set XCLI_CONFIG_FILE=%HOMEDRIVE%%HOMEPATH%\My Documents\xcli\xiv-systems.xml
REM List the current configuration
xcli -L
```

The XCLI utility requires user and password options. If user and passwords are not specified, the default environment variables XIV_XCLIUSER and XIV_XCLIPASSWORD are utilized.

The configurations are stored in a file under the user's home directory. A different file can be specified by -f or --file switch (applicable to configuration creation, configuration deletion, listing configurations, and command execution). Alternatively, the environment variable XCLI_CONFIG_FILE, if defined, determines the file's name and path. The default file is in %HOMEDRIVE%%HOMEPATH%\Application Data\XIV\GUI10\properties\xiv-systems.xml.

After executing the setup commands, the shortened command syntax works as shown in Example 4-4.

Example 4-4 Short command syntax

```
REM S1 can be used in all commands to save typing and script editing
set S1="XIV MN00035"
xcli -c %S1% user_list
```

Getting help with XCLI commands

To get help about the usage and commands, proceed as shown in Example 4-5.

Example 4-5 XCLI help commands

```
xcli
xcli -c "XIV MN00035" help
xcli -c "XIV MN00035" help command=user_list format=full
```

The first command prints out the usage of **xcli**. The second one prints all the commands that can be used by the user in that particular system. The third one shows the usage of the **user_list** command with all the parameters.

There are various parameters to get the result of a command in a predefined format. The default is the user readable format. Specify the **-s** parameter to get it in a comma-separated format or specify the **-x** parameter to obtain an XML format.

Note: The XML format contains all the fields of a particular command. The user and the comma-separated formats provide just the default fields as a result.

To list the field names for a specific **xcli** command output, use the **-t** parameter as shown in Example 4-6.

Example 4-6 XCLI field names

```
xcli -c "XIV MN00035" -t name,fields help command=user_list
```

4.3 Storage Pools

We have introduced the concept of XIV Storage Pools in 2.3.3, "Storage Pool concepts" on page 20.

Storage Pools function as a means to effectively manage a related group of logical volumes and their snapshots. Storage Pools offer the following key benefits:

- *Improved management of storage space*: Specific volumes can be grouped within a Storage Pool, giving you the flexibility to control the usage of storage space by specific applications, a group of applications, or departments.
- *Improved regulation of storage space*: Automatic snapshot deletion occurs when the storage capacity limit is reached for each Storage Pool independently. Therefore, when a Storage Pool's size is exhausted, only the snapshots that reside in the affected Storage Pool are deleted.

The size of Storage Pools and the associations between volumes and Storage Pools are constrained by:

- The size of a Storage Pool can range from as small as possible (17.1 GB) to as large as possible (the entire system) without any limitation.
- The size of a Storage Pool can always be increased, limited only by the free space on the system.
- The size of a Storage Pool can always be decreased, limited only by the space already consumed by the volumes and snapshots in that Storage Pool.
- Volumes can be moved between Storage Pools without any limitations, as long as there is enough free space in the target Storage Pool.

Important: All of these operations are handled by the system at the metadata level, and they do not cause any data movement (copying) from one disk drive to another. Hence, they are completed almost instantly and can be done at any time without impacting the applications.

Thin provisioned pools

Thin provisioning is the practice of allocating storage on a "just-in-time" and "as needed" basis by defining a logical, or soft, capacity that is larger than the physical, or hard, capacity. Thin provisioning enables XIV Storage System administrators to manage capacity based on the total space actually consumed rather than just the space allocated.

Thin provisioning can be specified at the Storage Pool level. Each thinly provisioned pool has its own hard capacity (which limits the actual disk space that can be effectively consumed) and soft capacity (which limits the total logical size of volumes defined).

The difference is in the pool size:

- *Hard pool size*: The hard pool size represents the physical storage capacity allocated to volumes and snapshots in the Storage Pool. The hard size of the Storage Pool limits the total of the hard volume sizes of all volumes in the Storage Pool and the total of all storage consumed by snapshots.
- *Soft pool size*: This size is the limit on the total soft sizes of all the volumes in the Storage Pool. The soft pool size has no effect on snapshots.

For more detailed information about the concept of XIV thin provisioning and a detailed discussion of hard and soft size for Storage Pools and volumes, refer to 2.3.4, "Capacity allocation and thin provisioning" on page 23.

When using the GUI, you specify what type of pool is desired (*Regular Pool* or a *Thin Provisioned Pool*) when creating the pool. Refer to "Creating Storage Pools" on page 99. When using the XCLI, you create a thinly provisioned pool by setting the soft size to a greater value than its hard size.

In case of changing requirements, the pool's type can be changed (non-disruptively) later.

Tip: The thin provisioning management is performed individually for each Storage Pool, and running out of space in one pool does not impact other pools.

4.3.1 Managing Storage Pools with the XIV GUI

Managing pools with the GUI is fairly simple and intuitive. As always, the related tasks can be reached by either the menu bar or the corresponding function icon on the left (called Pools), as shown in Figure 4-20.

Figure 4-20 Opening the Pools menu

To view overall information about the Storage Pools, select **Pools** from the Pools menu shown in Figure 4-20 to display the Storage Pool window seen in Figure 4-21.

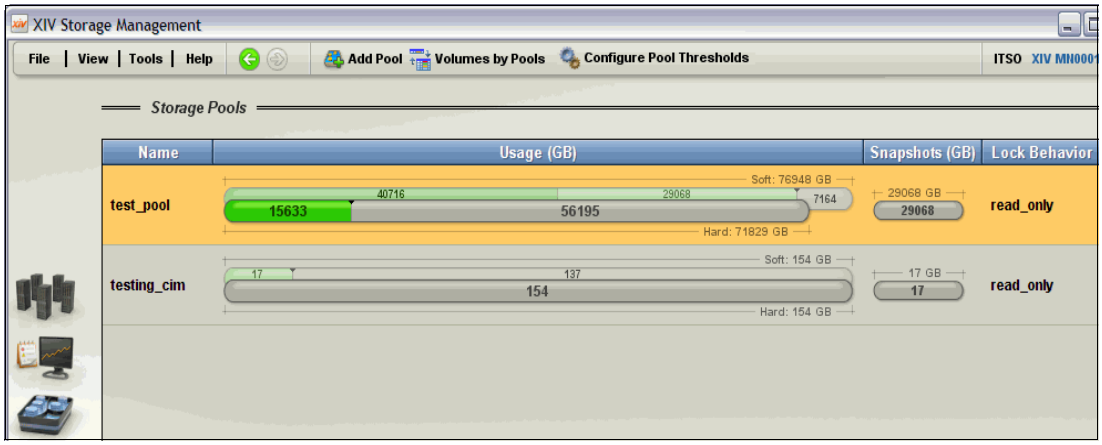


Figure 4-21 Storage Pools view

The Storage Pools GUI window displays a table of all the pools in the system combined with a series of gauges for each pool. This view gives the administrator a quick grasp and general overview of essential information about the system pools.

The capacity consumption by volumes and snapshots within a given Storage Pool is indicated by different colors:

- Green is the indicator for consumed capacity below 80%. Yellow represents a capacity consumption above 80%.
- Orange is the color for a capacity consumption of over 90%.
- Any Storage Pool with depleted hard capacity appears in red within this view.

The name, the size, and the separated segments are labeled adequately. Figure 4-22 shows the meaning of the various numbers.

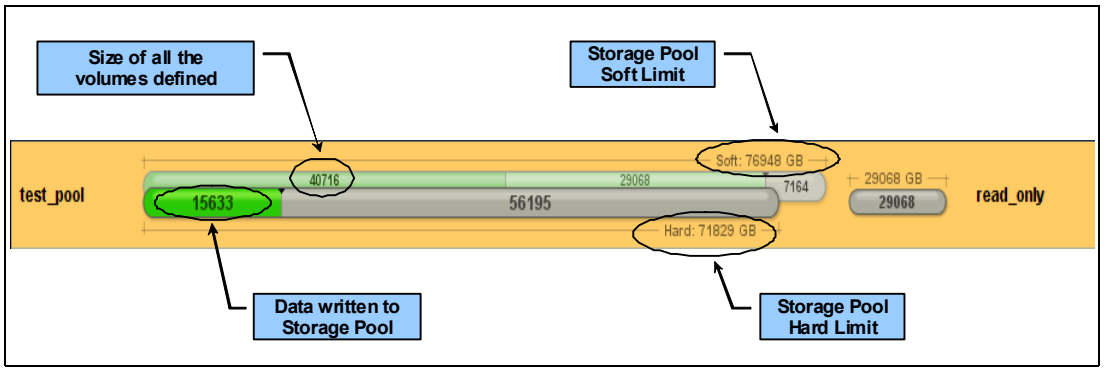


Figure 4-22 Storage Pool and size numbers

Creating Storage Pools

The creation and resizing of Storage Pools is relatively straightforward, and care need only be taken with the size allocation and re-allocation. The name of a Storage Pool must be unique in the system.

Note: The size of the Storage Pool is specified as an integer multiple of 10^9 bytes, but the actual size of the created Storage Pool is rounded up to the nearest integer multiple of $16x2^{30}$ bytes. According to this rule, the smallest pool size is 17.1 GB.

When creating a Storage Pool, a reserved area is automatically defined for snapshots. The system initially provides a default snapshots size, which can be changed at the time of creation or later to accommodate future needs.

Note: The Snapshots size (default or specified) is included in the specified pool size. It is not an additional space.

Sizing must take into consideration volumes that are to be added to (or already exist in) the specific Storage Pool, the current allocation of storage in the total system capacity, and future activity within the Storage Pool, especially with respect to snapshot propagation resulting from creating too many snapshots.

The system enables the assignment of the entire available capacity to user-created Storage Pools. The Storage Pool is initially empty and does not contain volumes. However, you cannot create a Storage Pool with zero capacity.

To create a Storage Pool:

1. Click **Add Pool** in the Storage Pools view or simply right-click in the body of the window. A Create Pool window displays as shown in Figure 4-23.

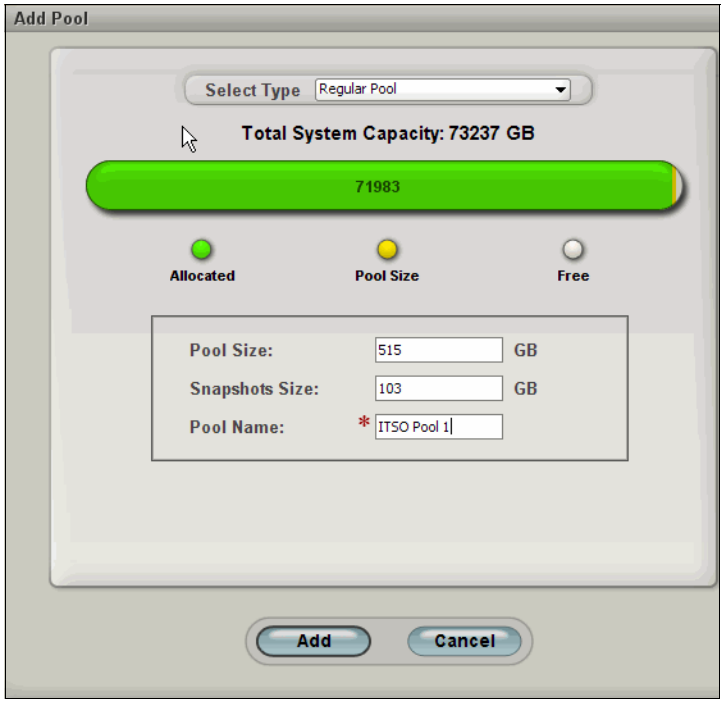


Figure 4-23 Create Pool

2. In the Select Type drop-down list box, choose *Regular* or *Thin Provisioned* according to your needs. For thinly provisioned pools, two new fields appear:
   - *Soft Size:* Here, you specify the upper limit of soft capacity.
   - *Lock Behavior:* Here, you specify the behavior in case of depleted capacity.

     This value specifies whether the Storage Pool is locked for write or whether it is disabled for both read and write when running out of storage space. The default value is `read only`.
3. In the Pool Size field, specify the required size of the Storage Pool.
4. In the Snapshots Size field, enter the required size of the reserved snapshot area.

> **Note:** Although it is possible to create a pool with identical snapshot and pool size, you cannot create a new volume in this type of a pool afterward without resizing first.

5. In the Pool Name field, enter the desired name (it must be unique across the Storage System) for the Storage Pool.
6. Click **Add** to add this Storage Pool.

Resizing Storage Pools

This action can be used to both increase or decrease a Storage Pool size. Capacity calculation is performed in respect to the total system net capacity. All reductions and increases are reflected in the remaining free storage capacity.

Notes:

- When increasing a Storage Pool size, you must ensure that the total system capacity holds enough free space to enable the increase in Storage Pool size.
- When decreasing a Storage Pool size, you must ensure that the Storage Pool itself holds enough free capacity to enable a reduction in size.

This operation is also used to shrink or increase the snapshot capacity inside the Storage Pool. This alteration only affects the space within the Storage Pool. In other words, increasing snapshot size will consume the free capacity only from the corresponding pool.

To change the size of one Storage Pool in the system, simply right-click in the Storage Pools view (Figure 4-21 on page 98) on the desired pool and choose **Resize**.

The window shown in Figure 4-24 is displayed. Change the pool hard size, soft size, or the snapshot size to match your new requirements.The green bar at the top of the window represents the system's actual hard capacity that is used by Storage Pools. The vertical red line indicates the current size, and the yellow part is the desired new size of the particular pool. Obviously, the remaining space in the bar without color indicates the consumable free capacity in the system.

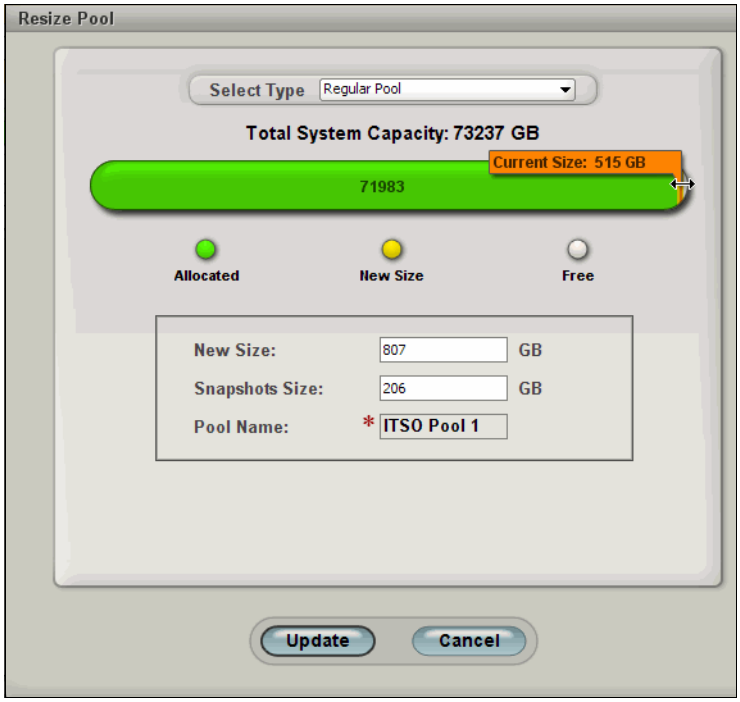


Figure 4-24 Resizing pool

The resize operation can also be used to change the type of Storage Pool from thin provisioned to regular or from regular to thin provisioned. Just change the type of the pool in the Resize Pool window Select Type list box. Refer to Figure 4-25:

- When a regular pool is converted to a thin provisioned pool, you have to specify an additional soft size parameter besides the existing hard size. Obviously, the soft size must be greater than the hard pool size.
- When a thin provisioned pool is changed to a regular pool, the soft pool size parameter will disappear from the window; in fact, its value will be equal to the hard pool size.

  If the space consumed by existing volumes exceeds the pool's actual hard size, the pool cannot be changed to a regular type pool. In this case, you have to specify a minimum pool hard size equal to the total capacity consumed by all the volumes within this pool.

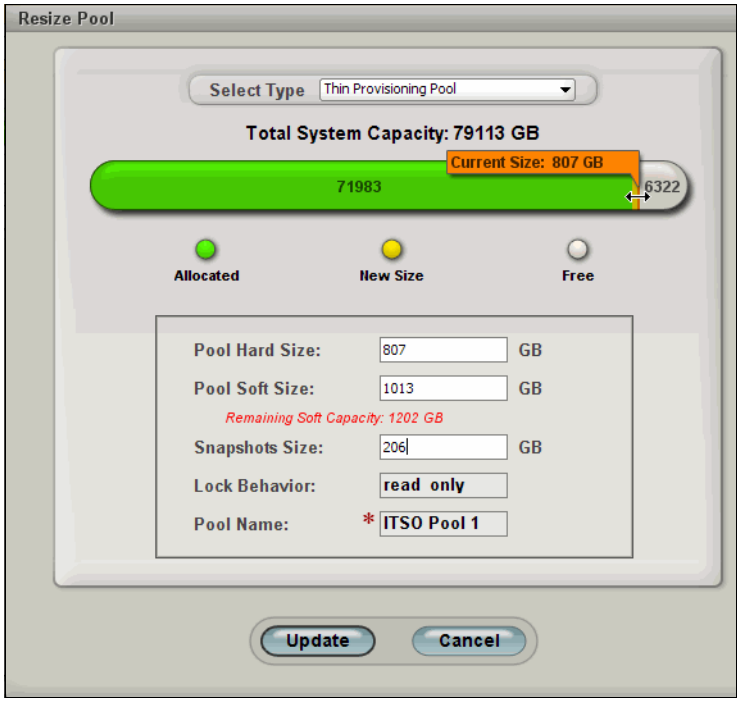


Figure 4-25 Resizing and changing the type of a pool

The remaining soft capacity is displayed in red characters and calculated by the system in the following manner:

```
Remaining Soft Capacity = [Current Storage Pool Soft Size + Remaining System Soft
Size] - Current Storage Pool Hard Size
```

Deleting Storage Pools

To delete a Storage Pool, right-click the Storage Pool and select **Delete**. The system will ask for a confirmation before deleting this Storage Pool.

The capacity of the deleted Storage Pool is reassigned to the system's free capacity, which means that the free hard capacity is increasing by the size of the deleted Storage Pool.

Restriction: You cannot delete a Storage Pool if it still contains volumes.

Moving volumes between Storage Pools

In order for a volume to be moved to a specific Storage Pool, there must be enough room for the volume to reside there. If there is not enough free capacity (meaning that adequate capacity has not been allocated), the Storage Pool must be resized, or other volumes must be moved out first to make room for the new volume.

When moving a master volume from one Storage Pool to another, all of its snapshots are moved along with it to the destination Storage Pool. You cannot move a snapshot alone, independently of its master volume.

The destination Storage Pool must have enough free storage capacity to accommodate the volume and its snapshots. The exact amount of storage capacity allocated from the destination Storage Pool is released at the source Storage Pool.

A volume that belongs to a Consistency Group cannot be moved without the entire Consistency Group.

As shown in Figure 4-26, in the Volume by Pools report, just select the appropriate volume with a right-click and initiate a Move to Pool operation to change the location of a volume.

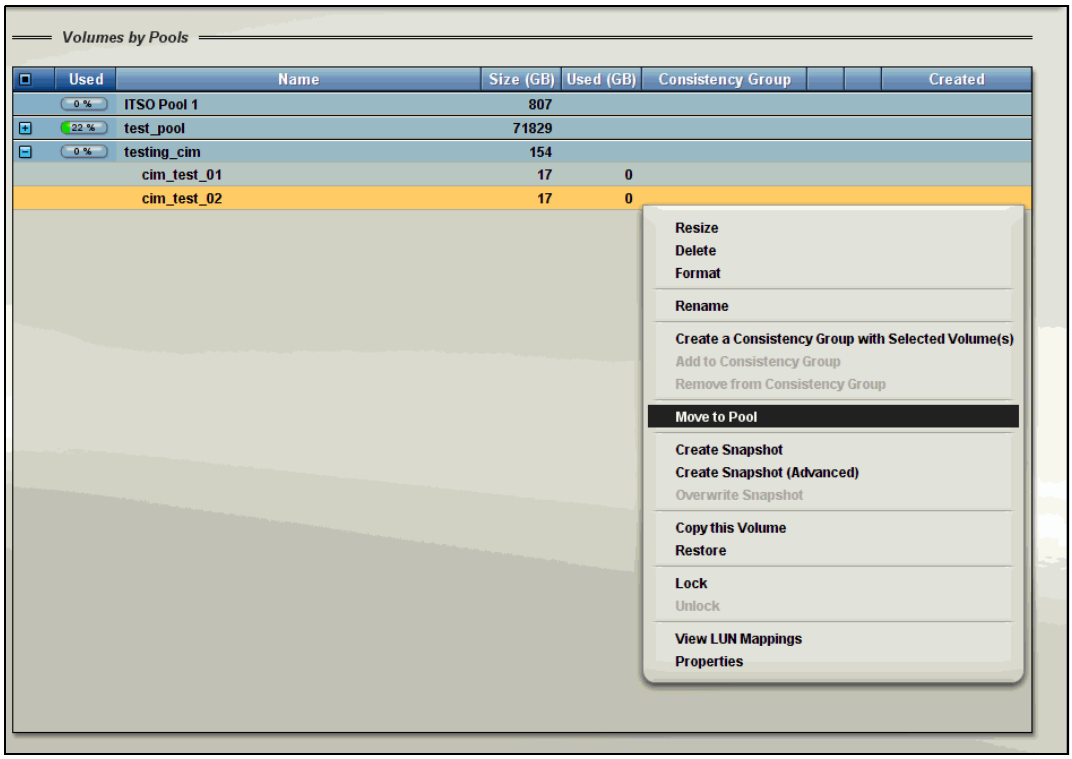


Figure 4-26 Volumes by Pools

In the pop-up window, select the appropriate Storage Pool as shown in Figure 4-27 and click **OK** to move the volume into it.

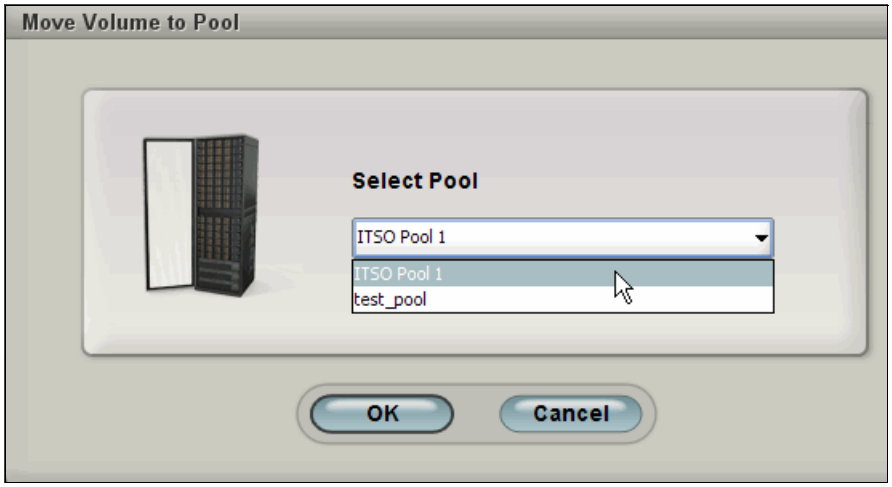


Figure 4-27 Move Volume to another Pool

4.3.2 Pool alert thresholds

You can use the XIV GUI to configure thresholds to trigger alerts at different severity level.

From the main GUI management window, select **Tools** → **Configure** → **Pool Alerts Thresholds** to get the menu shown in Figure 4-28.

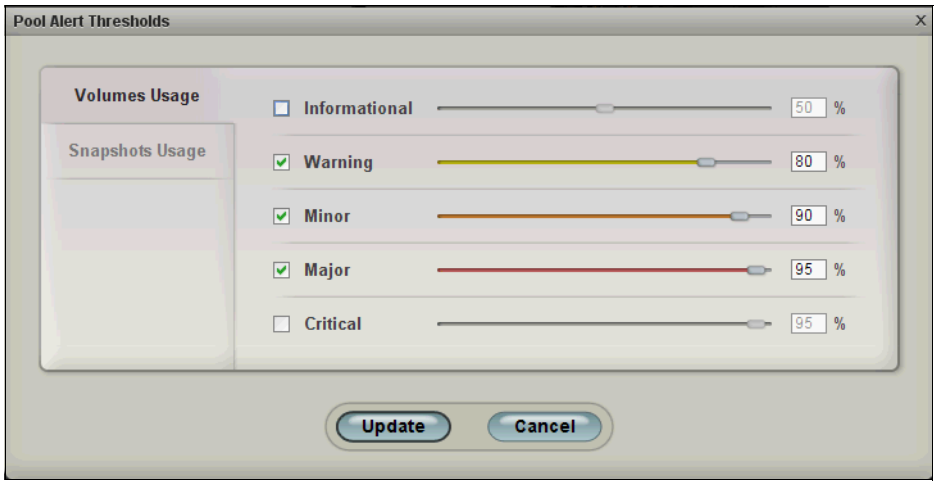


Figure 4-28 Set pool alert thresholds

4.3.3 Manage Storage Pools with XCLI

All of the operations described in 4.3.1, “Managing Storage Pools with the XIV GUI” on page 98, can also be done through the command line interface.

To get a list of all the Storage Pool-related XCLI commands, type the following command from the XCLI command shell:

```
help category=storage-pool
```

Important: Note that the commands shown in this section assume that you have started an XIV XCLI Session to the system selected; see “XCLI Session features” on page 94.

The output shown in Example 4-7 is displayed.

Example 4-7 All the Storage Pool-related commands

```
Category      Name                Description
storage-pool  cg_move             Moves a Consistency Group, all its volumes and all their
                                  Snapshots and Snapshot Sets from one Storage Pool to another.
storage-pool  pool_change_config  Changes the Storage Pool Snapshot limitation policy.
storage-pool  pool_create         Creates a Storage Pool.
storage-pool  pool_delete         Deletes a Storage Pool.
storage-pool  pool_list           Lists all Storage Pools or the specified one.
storage-pool  pool_rename         Renames a specified Storage Pool.
storage-pool  pool_resize         Resizes a Storage Pool.
storage-pool  vol_move            Moves a volume and all its Snapshot from one Storage Pool to
                                  another.
```

To list the existing Storage Pools in a system, use the following command:

```
pool_list
```

A sample result of this command is illustrated in Figure 4-29.

```
.                     Hard      Snapshot Empty     Used by     Used by
Name          Size(GB) Size(GB) Size(GB) Space(GB) volumes(GB) Snapshots(GB) Locked
test_pool     76948     71829     29068     7198   15633       0             no
testing_cim   154       154       17        103    0           0             no
ITSO Pool 1   1013      807       206       807    0           0             no
```

Figure 4-29 Result of the pool_list command

For the purpose of new pool creation, enter the following command:

```
pool_create pool="ITSO Pool 1" size=515 snapshot_size=103
```

The size of the Storage Pool is specified as an integer multiple of 10^9 bytes, but the actual size of the created Storage Pool is rounded up to the nearest integer multiple of $16x2^{30}$ bytes. The `snapshot_size` parameter specifies the size of the snapshot area within the pool. It is a mandatory parameter, and you must specify a positive integer value for it.

The following command shows how to resize one of the existing pools:

```
pool_resize pool="ITSO Pool 1" size=807 snapshot_size=206
```

With this command, you can increase or decrease the pool size. The **`pool_create`** and the **`pool_resize`** commands are also used to manage the size of the snapshot area within a Storage Pool.

To rename an existing pool, issue this command:

```
pool_rename new_name="ITSO Pool" pool="ITSO Pool 1"
```

To delete a pool, type:

```
pool_delete pool="ITSO Pool"
```

Use the following command to move the volume named `cim_test_02` to the Storage Pool `ITSO Pool 1`:

```
vol_move pool="ITSO Pool 1" vol="cim_test_02"
```

The command only succeeds if the destination Storage Pool has enough free storage capacity to accommodate the volume and its snapshots.

The following command will move a particular volume and its snapshots from one Storage Pool to another, but if the volume is part of a Consistency Group, the entire group must be moved. In this case, the **cg_move** command is the correct solution:

```
cg_move cg="ITSO CG" pool="ITSO Pool 1"
```

All volumes in the Consistency Group are moved, all snapshot groups of this Consistency Group are moved, and all snapshots of the volumes are moved.

Thin provisioned pools

To create thinly provisioned pools, specify the hard_size and the soft_size parameters. For thin provisioning concepts, refer to the 2.3.4, “Capacity allocation and thin provisioning” on page 23.

A typical Storage Pool creation command with thin provisioning parameters can be issued as shown in the following example:

```
pool_create pool="ITSO Pool" hard_size=807 soft_size=1013 lock_behavior=read_only
snapshot_size=206
```

The `soft_size` is the maximal storage capacity seen by the host and cannot be smaller than the `hard_size`, which is the hard physical capacity of the Storage Pool.

If a Storage Pool runs out of hard capacity, all of its volumes are locked to all write commands. Even though write commands that overwrite existing data can be technically serviced, they are blocked as well in order to ensure consistency.

To specify the behavior in case of depleted capacity reserves in a thin provisioned pool, use the following command:

```
pool_change_config pool="ITSO Pool" lock_behavior=no_io
```

This command specifies whether the Storage Pool is locked for write or whether it disables both read and write when running out of storage space.

Note: The lock_behavior parameter can be specified for non-thin provisioning pools, but it has no effect.

4.4 Volumes

After defining Storage Pools, the next milestone in the XIV Storage System configuration is volume management.

The XIV Storage System offers logical volumes as the basic data storage element for allocating usable storage space to attached hosts. This logical unit concept is well known and is widely used by other storage subsystems and vendors. However, neither the volume segmentation nor its distribution over the physical disks is conventional in the XIV Storage System.

Traditionally, logical volumes are defined within various RAID arrays, where their segmentation and distribution are manually specified. The result is often a suboptimal distribution within and across modules (expansion units) and is significantly dependent upon the administrator's knowledge and expertise.

As explained in 2.3, "Full storage virtualization" on page 14, the XIV Storage System uses true virtualization as one of the basic principles for its unique design. With XIV, each volume is divided into tiny 1 MB partitions, and these partitions are distributed randomly and evenly, and duplicated for protection. The result is optimal distribution in and across all modules, which means that for any volume, the physical drive location and data placement are invisible to the user. This method dramatically simplifies storage provisioning, letting the system lay out the user's volume in an optimal way.

This method offers complete virtualization, without requiring preliminary volume layout planning or detailed and accurate stripe or block size pre-calculation by the administrator. All disks are equally used to maximize the I/O performance and exploit all the processing power and all the bandwidth available in the storage system.

XIV Storage System virtualization incorporates an advanced snapshot mechanism with unique capabilities, which enables creating a virtually unlimited number of point-in-time copies of any volume, without incurring any performance penalties. The concept of snapshots is discussed in detail in the *Theory of Operations*, GA32-0639-03.

Volumes can also be grouped into larger sets called *Consistency Groups* and *Storage Pools*. Refer to 4.3, "Storage Pools" on page 96.

Important: As shown in Figure 4-30, the basic hierarchy is as follows:

- A volume can have multiple snapshots.
- A volume can be part of one and only one Consistency Group.
- A volume is always a part of one and only one Storage Pool.
- All volumes of a Consistency Group must belong to the same Storage Pool.

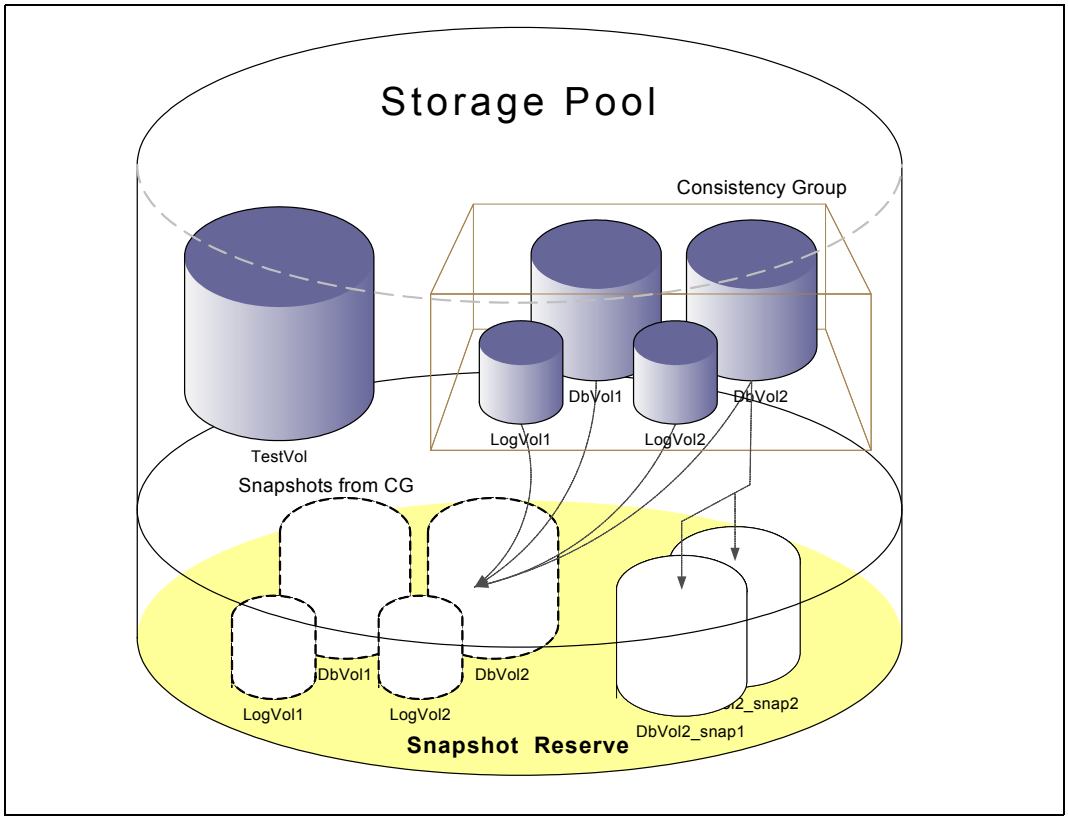


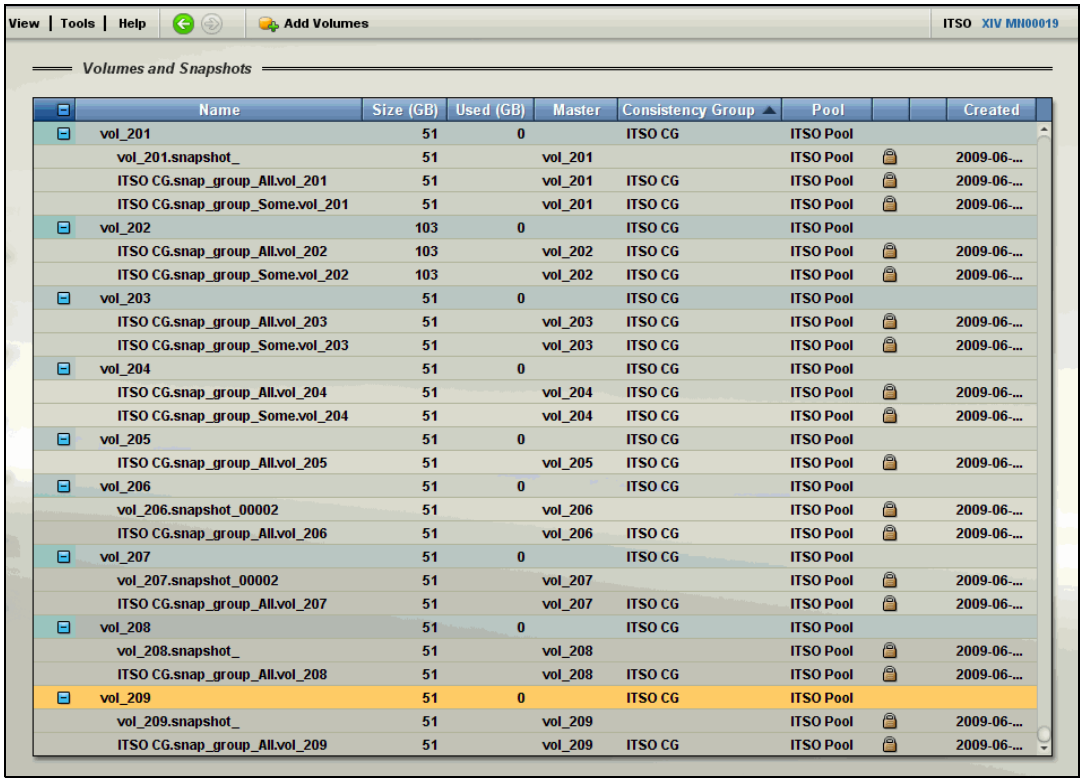
Figure 4-30 Basic storage hierarchy

4.4.1 Managing volumes with the XIV GUI

To start a volume management function from the XIV GUI, you can either select **View** → **Volumes** → **Volumes** from the menu bar or click the **Volume** icon and then select the appropriate menu item. Refer to Figure 4-31.

Figure 4-31 Opening the Volumes menu

The Volumes & Snapshots menu item is used to list all the volumes and snapshots that have been defined in this particular XIV Storage System. An example of the resulting window can be seen in Figure 4-32.



Name	Size (GB)	Used (GB)	Master	Consistency Group	Pool		Created
vol_201	51	0		ITSO CG	ITSO Pool		
vol_201.snapshot_	51		vol_201		ITSO Pool		2009-06-...
ITSO CG.snap_group_All.vol_201	51		vol_201	ITSO CG	ITSO Pool		2009-06-...
ITSO CG.snap_group_Some.vol_201	51		vol_201	ITSO CG	ITSO Pool		2009-06-...
vol_202	103	0		ITSO CG	ITSO Pool		
ITSO CG.snap_group_All.vol_202	103		vol_202	ITSO CG	ITSO Pool		2009-06-...
ITSO CG.snap_group_Some.vol_202	103		vol_202	ITSO CG	ITSO Pool		2009-06-...
vol_203	51	0		ITSO CG	ITSO Pool		
ITSO CG.snap_group_All.vol_203	51		vol_203	ITSO CG	ITSO Pool		2009-06-...
ITSO CG.snap_group_Some.vol_203	51		vol_203	ITSO CG	ITSO Pool		2009-06-...
vol_204	51	0		ITSO CG	ITSO Pool		
ITSO CG.snap_group_All.vol_204	51		vol_204	ITSO CG	ITSO Pool		2009-06-...
ITSO CG.snap_group_Some.vol_204	51		vol_204	ITSO CG	ITSO Pool		2009-06-...
vol_205	51	0		ITSO CG	ITSO Pool		
ITSO CG.snap_group_All.vol_205	51		vol_205	ITSO CG	ITSO Pool		2009-06-...
vol_206	51	0		ITSO CG	ITSO Pool		
vol_206.snapshot_00002	51		vol_206		ITSO Pool		2009-06-...
ITSO CG.snap_group_All.vol_206	51		vol_206	ITSO CG	ITSO Pool		2009-06-...
vol_207	51	0		ITSO CG	ITSO Pool		
vol_207.snapshot_00002	51		vol_207		ITSO Pool		2009-06-...
ITSO CG.snap_group_All.vol_207	51		vol_207	ITSO CG	ITSO Pool		2009-06-...
vol_208	51	0		ITSO CG	ITSO Pool		
vol_208.snapshot_	51		vol_208		ITSO Pool		2009-06-...
ITSO CG.snap_group_All.vol_208	51		vol_208	ITSO CG	ITSO Pool		2009-06-...
vol_209	51	0		ITSO CG	ITSO Pool		
vol_209.snapshot_	51		vol_209		ITSO Pool		2009-06-...
ITSO CG.snap_group_All.vol_209	51		vol_209	ITSO CG	ITSO Pool		2009-06-...

Figure 4-32 Volumes and Snapshots view

Volumes are listed in a tabular format. If the volume has snapshots, then a + or a - icon appears on the left. Snapshots are listed under their master volumes, and the list can be expanded or collapsed at the volume level by clicking the + or - icon respectively.

Snapshots are listed as a sub-branch of the volume of which they are a replica, and their row is indented and highlighted in off-white.

The Master column of a snapshot shows the name of the volume of which it is a replica. If this column is empty, the volume is the master.

Tip: To customize the columns in the lists, just click one of the column headings and make the required selection of attributes. The default column set does not contain the Master column. You can also resize the columns to allow for longer names or to make more columns visible.

Table 4-1 shows the columns of this view with their descriptions.

Table 4-1 Columns in the Volumes and Snapshots view

Column	Description	Default
Qty.	indicates the number snapshots belonging to a volume	N
Name	Name of a volume or snapshot	Y
Size (GB)	Volume or snapshot size. (value is zero if the volume is specified in blocks)	Y
Used (GB)	Used capacity in a volume	Y
Size (Blocks)	Volume size in blocks	N
Size (Consume)	Consumed capacity	N
Master	Snapshot Master's name	N
Consistency Group	Consistency Group name	Y
Pool	Storage Pool name	Y
( )	Indicates the locking status of a volume or snapshot. Lock icon.	Y
( )	Shows if the snapshot was unlocked or modified	Y
Deletion Priority	Indicates the priority of deletion by numbers for snapshots	N
Created	Shows the creation time of a snapshot	Y
Creator	Volume or snapshot creator name	N
Serial Number	Volume or snapshot serial number	N
Sync Type	Shows the mirroring type status	N

Most of the volume-related and snapshot-related actions can be selected by right-clicking any row in the table to display a drop-down menu of options. The options in the menu differ slightly for volumes and snapshots.

Menu option actions

The following actions can be performed through these menu options:

- Adding or creating volumes; refer to "Creating volumes" on page 111
- Resizing a volume; refer to "Resizing volumes" on page 115
- Deleting a volume or snapshot; refer to "Deleting volumes" on page 116
- Formatting a volume
- Renaming a volume or snapshot
- Creating a Consistency Group with these volumes

- Adding to a Consistency Group
- Removing from a Consistency Group
- Moving volumes Between Storage Pools; refer to “Moving volumes between Storage Pools” on page 103
- Creating a snapshot
- Creating a snapshot/(advanced)
- Overwriting a snapshot
- Copying a volume or snapshot
- Locking/unlocking a volume or snapshot
- Mappings
- Displaying properties of a volume or snapshot
- Changing a snapshot’s deletion priority
- Duplicating a snapshot or a snapshot (advanced)
- Restoring from a snapshot

Creating volumes

When you *create a volume in a traditional or regular Storage Pool*, the entire volume storage capacity is reserved (static allocation). In other words, you cannot define more space for volumes in a regular Storage Pool than the actual hard capacity of the pool, which guarantees the functionality and integrity of the volume.

If you *create volumes in a Thin Provisioned Pool,* the capacity of the volume will not be reserved immediately to the volumes, but a basic 17.1 GB piece, taken out of the Storage Pool hard capacity, will be allocated at the first I/O operation. In a Thin Provisioned Pool, you are able to define more space for volumes than the actual hard capacity of the pool, up to the soft size of the pool.

The volume size is the actual “net” storage space, as seen by the host applications, not including any mirroring or other data protection overhead. The free space consumed by the volume will be the smallest multiple of 17 GB that is greater than the specified size. For example, if we request an 18 GB volume to be created, the system will round this volume size to 34 GB. In case of a 16 GB volume size request, it will be rounded to 17 GB.

Figure 4-33 gives you several basic examples of volume definition and planning in a thinly provisioned pool. It depicts the volumes with the minimum amount of capacity, but the principle can be used for larger volumes as well.

As shown in this picture, we recommend that you plan carefully the number of volumes or the hard size of the thinly provisioned pool because of the minimum hard capacity that is consumed by one volume.

If you create more volumes in a thinly provisioned pool than the hard capacity can cover, the I/O operations against the volumes will fail at the first I/O attempt.

Note: We recommend that you plan the volumes in a Thin Provisioned Pool in accordance with this formula: `Pool Hard Size >= 17 GB x (number of volumes in the pool)`

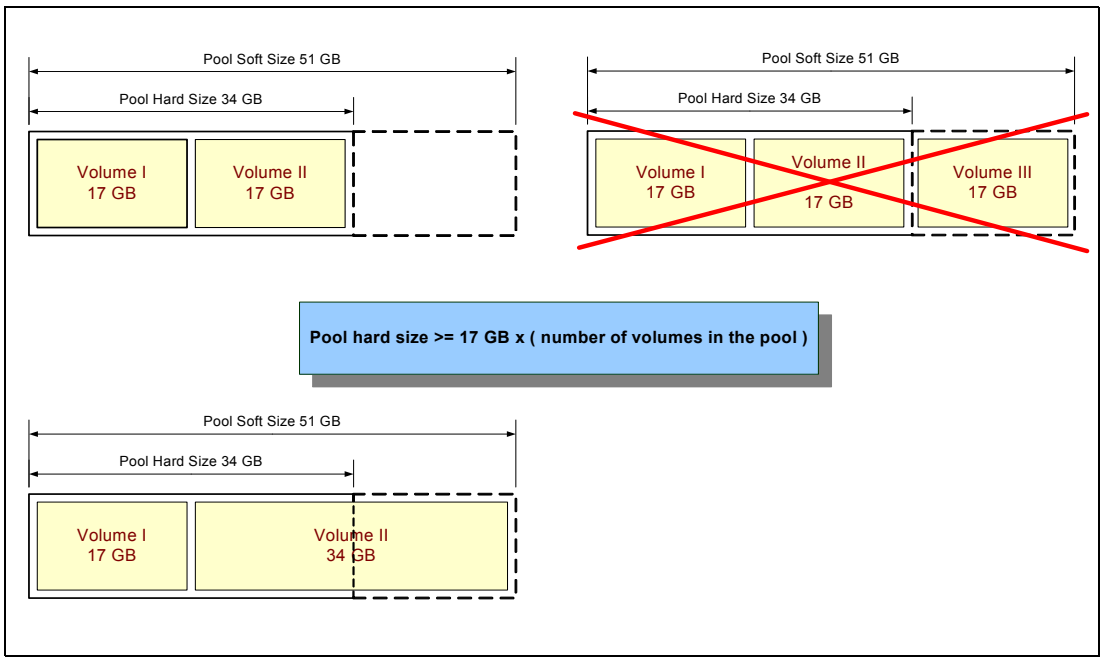


Figure 4-33 Planning the number of volumes in a Thin Provisioning Pool

The size of a volume can be specified either in gigabytes (GB) or in blocks (where each block is 512 bytes). If the size is specified in blocks, volumes are created in the exact size specified, and the size will be not rounded up. It means that the volume will show the exact block size and capacity to the hosts but will nevertheless consume a 17 GB size in the XIV Storage System. This capability is relevant and useful in migration scenarios.

If the size is specified in gigabytes, the actual volume size is rounded up to the nearest 17.1 GB multiple (making the actual size identical to the free space consumed by the volume, as just described). This rounding up prevents a situation where storage space is not fully utilized because of a gap between the free space used and the space available to the application.

The volume is logically formatted at creation time, which means that any read operation results in returning all zeros as a response.

To create volumes with the XIV Storage Management GUI:

1. Click the “add volumes” icon in the Volume and Snapshots view (Figure 4-32 on page 109) or right-click in the body of the window (not on a volume or snapshot) and select Add Volumes. The window shown in Figure 4-34 on page 113 is displayed.
2. From the Select Pool field, select the Pool in which this volume is stored. You can refer to 4.3, “Storage Pools” on page 96 for a description of how to define Storage Pools. The storage size and allocation of the selected Storage Pool is shown textually and graphically in a color-coded bar:
   - Green indicates the space already allocated in this Storage Pool.
   - Yellow indicates the space that will be allocated to this volume (or volumes) after it is created.
   - Gray indicates the space that remains free after this volume (or volumes) is allocated.

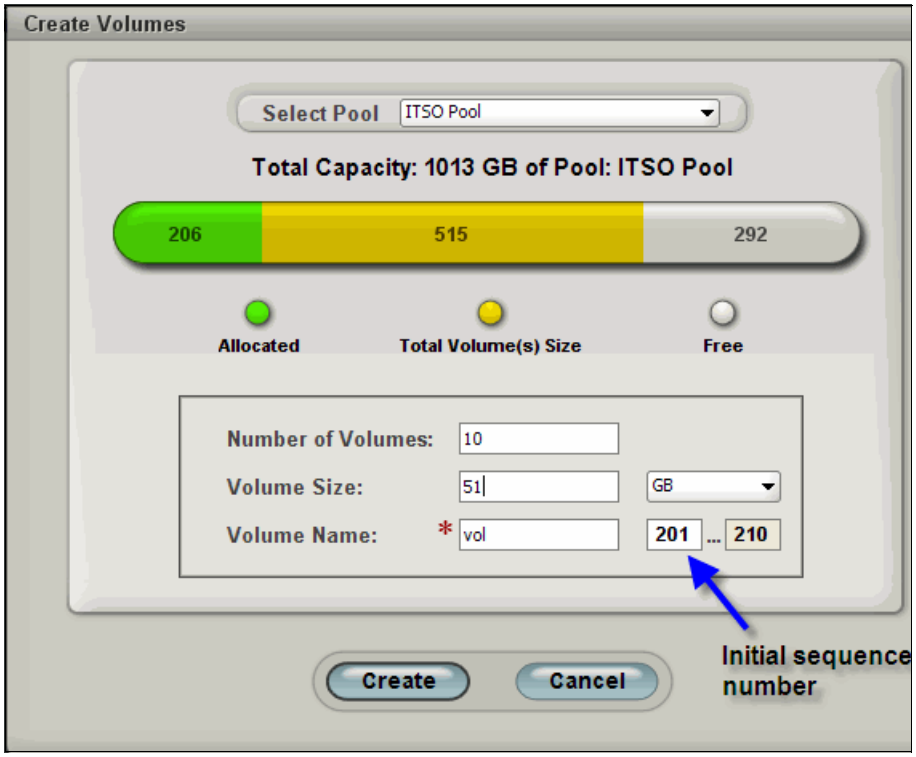


Figure 4-34 Create Volumes

3. In the Number of Volumes field, specify the required number of volumes.
4. In the Volume Size field, specify the size of each volume to define. The size can also be modified by dragging the yellow part of the size indicator.

   Note: When multiple volumes are created, they all have the same size as specified in the Volume Size field.

5. In the Volume Name field, specify the name of the volume to define. The name of the volume must be unique in the system. If you specified that more than one volume be defined, they are successively named by appending an incrementing number to end of the specified name. You can also add an initial sequence number.

6. Click **Create** to effectively create and add the volumes to the Storage Pool (Figure 4-35).

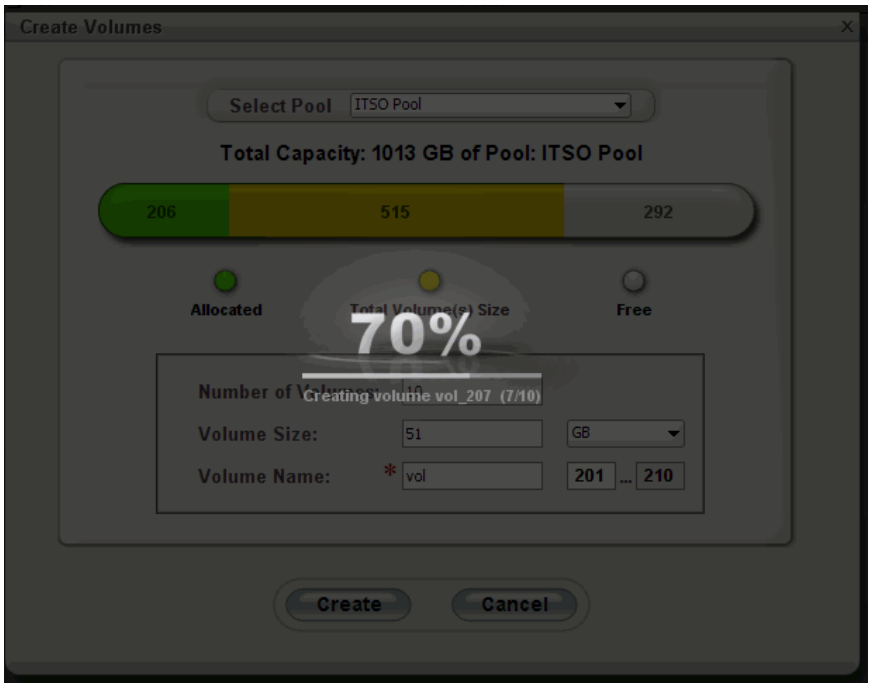
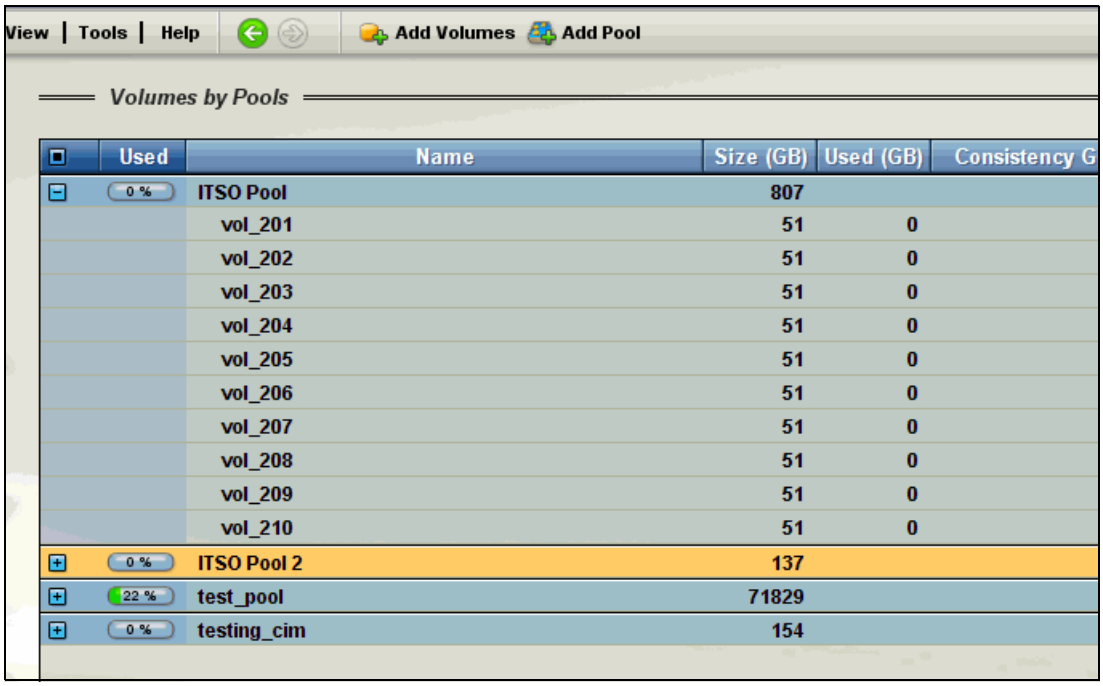


Figure 4-35 Volume Creation progress indicator

After a volume is successfully added, its state is unlocked, meaning that write, format, and resize operations are permitted. The creation time of the volume is set to the current time and is never changed. Notice the volume name sequence in Figure 4-36.



Used	Name	Size (GB)	Used (GB)	Consistency G
0 %	ITSO Pool	807		
	vol_201	51	0	
	vol_202	51	0	
	vol_203	51	0	
	vol_204	51	0	
	vol_205	51	0	
	vol_206	51	0	
	vol_207	51	0	
	vol_208	51	0	
	vol_209	51	0	
	vol_210	51	0	
0 %	ITSO Pool 2	137		
22 %	test_pool	71829		
0 %	testing_cim	154		

Figure 4-36 Volumes Created

Resizing volumes

Resizing volumes is an operation very similar to their creation. Only an unlocked volume can be resized. When you resize a volume, its size is specified as an integer multiple of 10^9 bytes, but the actual new size of the volume is rounded up to the nearest valid size, which is an integer multiple of 17 GB.

Note: The size of the volume can be decreased. However, to avoid possible data loss, you must contact your IBM XIV support personnel if you need to decrease a volume size. (Mapped volume size cannot be decreased.)

The volume address space is extended (at the end of the existing volume) to reflect the increased size, and the additional capacity is logically formatted (that is, zeroes are returned for all read commands).

When resizing a regular volume (not a writable snapshot), all storage space that is required to support the additional volume capacity is reserved (static allocation), which guarantees the functionality and integrity of the volume, regardless of the resource levels of the Storage Pool containing that volume.

Resizing a master volume does not change the size of its associated snapshots. These snapshots can still be used to restore their individual master volumes at their initial sizes.

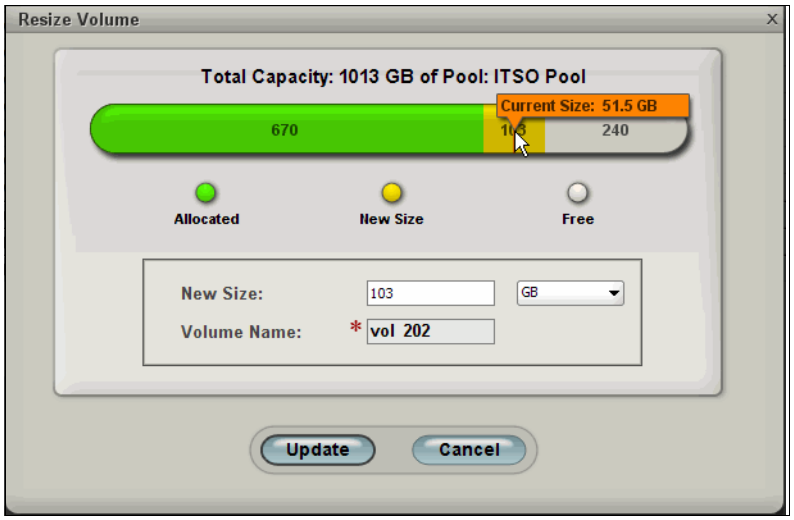


Figure 4-37 Resize an existing volume

To resize volumes with XIV Storage Management GUI:

1. Right-click the row of the volume to be resized and select **Resize**.

   The total amount of storage is presented both textually and graphically. The amount that is already allocated by the other existing volumes is shown in green. The amount that is free is shown in gray. The current size of the volume is displayed in yellow, to the left of a red vertical bar. This red bar provides a constant indication of the original size of the volume as you resize it. Place the mouse cursor over the red bar to display the volume's initial size.

2. In the New Size field, use the arrows to set the new size or type the new value.
3. Click Update to resize the volume.

Deleting volumes

With the GUI, the deletion of a volume is as easy as creating one.

Important: After you delete a volume or a snapshot, all data stored on the volume is lost and cannot be restored.

All the storage space that was allocated (or reserved) for the volume or snapshot is freed and returned to its Storage Pool. The volume or snapshot is then removed from all the logical unit number (LUN) Maps that contain mapping of this volume.

Deleting a volume deletes all the snapshots associated with this volume, even snapshots that are part of snapshot Groups. This deletion can only happen when the volume was in a Consistency Group and was removed from it.

You can delete a volume regardless of the volume's lock state, but you cannot delete a volume that is part of a Consistency Group.

To delete a volume or a snapshot:

1. Right-click the row of the volume to be deleted and select **Delete**.
2. Click to delete the volume.

Maintaining volumes

There are several other operations that can be issued on a volume. Refer to "Menu option actions" on page 110.

The usage of these operations is obvious, and you can initiate an operation with a right-mouse click. These operations are:

- *Format a volume:* A formatted volume returns zeros as a response to any read command. The formatting of the volume is done logically, and no data is actually written to the physical storage space allocated for the volume. Consequently, the formatting action is performed instantly.
- *Rename a volume:* A volume can be renamed to a unique name in the system. A locked volume can also be renamed.
- *Lock/Unlock a volume:* You can lock a volume so that hosts cannot write to it. A volume that is locked is write-protected, so that hosts can read the data stored on it, but they cannot change it. The volume appears then as a lock icon. In addition, a locked volume cannot be formatted or resized. In general, locking a volume prevents any operation (other than deletion) that changes the volume's image.

  Note: Master volumes are set to unlocked when they are created. Snapshots are set to locked when they are created.

- *Consistency Groups:* XIV Storage System enables a higher level of volume management provided by grouping volumes and snapshots into sets called *Consistency Groups*. This kind of grouping is especially useful for cluster-specific volumes.
- *Copy a volume:* You can copy a source volume onto a target volume. Obviously, all the data that was previously stored on the target volume is lost and cannot be restored.
- *Snapshot functions:* The XIV Storage System's advanced snapshot feature has unique capabilities that enable the creation of a virtually unlimited number of copies of any volume, with no performance penalties.

- *Map a volume:* While the storage system sees volumes and snapshots at the time of their creation, the volumes and snapshots are visible to the hosts only after the mapping procedure. To get more information about mapping, refer to 4.5, “Host definition and mappings” on page 118.

4.4.2 Managing volumes with XCLI

All of the operations that are explained in 4.4.1, “Managing volumes with the XIV GUI” on page 108 can also be performed through the command line interface. To get a list of all the volume-related commands, enter the following command in the XCLI Session:

```
help category=volume
```

Important: Note that the commands shown in this section assume that you have started an XIV XCLI Session to the system selected; see “XCLI Session features” on page 94

Example 4-8 shows the output of the command.

Example 4-8 All the volume-related commands

```
Category   Name                 Description
volume     reservation_clear    Clear reservations of a volume.
volume     reservation_key_list Lists reservation keys.
volume     reservation_list     Lists volume reservations.
volume     vol_by_id            Prints the volume name according to its specified SCSI serial number.
volume     vol_copy             Copies a source volume onto a target volume.
volume     vol_create           Creates a new volume.
volume     vol_delete           Deletes a volume.
volume     vol_format           Formats a volume.
volume     vol_list             Lists all volumes or a specific one.
volume     vol_lock             Locks a volume so that it is read-only.
volume     vol_rename           Renames a volume.
volume     vol_resize           Resizes a volume.
volume     vol_unlock           Unlocks a volume, so that it is no longer read-only and can be written to.
```

To list the existing volumes in a system, use the following command:

```
vol_list pool="ITSO Pool"
```

The result of this command is similar to the illustration given in Figure 4-38.

```
Name     Size (GB)   Master Name   Consistency Group   Pool Creator   Used Capacity(GB)
vol_201  51                                           ITSO Pool   ITSO     0
vol_202  103                                          ITSO Pool   ITSO     0
vol_203  51                                           ITSO Pool   ITSO     0
vol_204  51                                           ITSO Pool   ITSO     0
vol_205  51                                           ITSO Pool   ITSO     0
vol_206  51                                           ITSO Pool   ITSO     0
vol_207  51                                           ITSO Pool   ITSO     0
vol_208  51                                           ITSO Pool   ITSO     0
vol_209  51                                           ITSO Pool   ITSO     0
vol_210  51                                           ITSO Pool   ITSO     0
>>
```

Figure 4-38 vol_list command output

To find and list a specific volume by its SCSI ID, issue the following command:

```
vol_by_id=12
```

To create a new volume, enter the following command:

```
vol_create size=51 pool="ITSO Pool" vol="vol_201"
```

The size can be specified either in gigabytes or in blocks (where each block is 512 bytes). If the size is specified in blocks, volumes are created in the exact size specified. If the size is specified in gigabytes, the actual volume size is rounded up to the nearest 17 GB multiple (making the actual size identical to the free space consumed by the volume, as described above). This rounding up prevents a situation where storage space is not fully utilized because of a gap between the free space used and the space available to the application.

Note: If pools are already created in the system, the specification of the Storage Pool name is mandatory.

The volume is logically formatted at creation time, which means that any read operation results in returning all zeros as a response. To format a volume, use the following command:

```
xcli -c Redbook vol_format vol=DBVolume
```

Note that all data stored on the volume will be lost and unrecoverable. If you want to bypass the warning message, just put -y right after the XCLI command.

The following example shows how to resize one of the existing volumes:

```
vol_resize vol="vol_202" size=103
```

With this command, you can increase or decrease the volume size. However, to avoid data loss, contact the XIV Storage System support personnel if you need to decrease the size of a volume.

To rename an existing volume, issue this command:

```
vol_rename new_name="vol_200" vol="vol_210"
```

To delete an existing created volume, enter:

```
vol_delete vol="vol_200"
```

4.5 Host definition and mappings

Because the XIV Storage System can be attached to multiple, heterogeneous hosts, it is necessary to specify which particular host can access which specific logical drives in the XIV Storage System. In other words, mappings must be defined between hosts and volumes in the XIV Storage System.

The XIV Storage System is able to manage single hosts or hosts grouped together in clusters.

See 6.4, “Logical configuration for host connectivity” on page 209 for details related to Host definitions and volume mapping.

4.6 Scripts

IBM XIV Storage Manager software XCLI commands can be used in scripts or batch programs in case you need to use repetitive or complex operations. The XCLI can be used in a shell environment to interactively configure the system or as part of a script to perform specific tasks; see Figure 4-3 on page 95. In general, the XIV GUI or the XCLI Session environment will virtually eliminate the need for scripts.

5

Security

This chapter discusses the XIV Storage System security features from different perspectives. More specifically, it covers the following topics:

- System physical access security
- Native user authentication
- LDAP managed user authentication
- Securing LDAP communication with Security Socket Layer
- Audit event logging

5.1 Physical access security

When installing an XIV Storage System, you need to apply the same security best practices that you apply to any other business critical IT system. A good reference on storage security can be found at the Storage Networking Industry Association (SNIA) Web site:

http://www.snia.org/forums/ssif/programs/best_practices

A common risk with storage systems is the retention of volatile caches. The XIV Storage System is perfectly safe in regard to external operations and a loss of external power. In the case of a power failure, the internal Uninterruptible Power Supply (UPS) units provide power to the system. The UPS enables the XIV Storage System to gracefully shut down.

However, if someone were to gain physical access to the equipment, that person might manually shut off components by bypassing the recommended process. In this case, the storage system will likely lose the contents of its volatile caches, resulting in a data loss and system unavailability. To eliminate or greatly reduce this risk, the XIV rack is equipped with lockable doors; you can prevent unauthorized people from accessing the rack by simply locking the doors, which will also protect against unintentional as well as malicious changes *inside* the system rack.

Important: Protect your XIV Storage System by locking the rack doors and monitoring physical access to the equipment.

5.2 Native user authentication

To prevent unauthorized access to the configuration of the storage system and ultimately to the information on its volumes, the XIV Storage System uses password based user authentication. Password based authentication is a form of challenge-response authentication protocol where the authenticity of a user is established by presenting that user with a question “challenge” and comparing the answer “response” with information stored in a credential repository.

By default, the XIV Storage System is configured to use native (XIV managed) user authentication. Native user authentication makes use of the credential repository stored locally on the XIV system. The XIV local credential repository maintains the following types objects:

- User name
- User password
- User role
- User group
- Optional account attributes

User name

A user name is a string of 1-64 characters that can only contain ‘a-z’, ‘A-Z’, ‘0-9’, ‘.-_~’ and ‘space’ symbols. User names are case sensitive. The XIV Storage System is configured with a set of predefined user names. Predefined user names and corresponding default passwords exist to provide initial access to XIV at the time of installation, for system maintenance, and for integration with application such as the Tivoli Storage Productivity Center.

The following user accounts are predefined on the XIV system:

- *technician*: This account is used by the IBM support representative to install the XIV Storage system
- *admin:* This account provides the highest level of customer access to the system. It can be used for creating new users and change passwords for existing users in native authentication mode.

> **Important:** Use of the *admin* account should be limited to the initial configuration when no other user accounts are available. Access to the *admin* account must be restricted and securely protected.

- *smis_user:* This user account has read-only permissions and is used to provide access for IBM Tivoli Storage Productivity Center software to collect capacity and configuration related data.
- *xiv_development and xiv_maintenance user:* These IDs are special case pre-defined internal IDs that can only be accessed by qualified IBM development and service support representatives (SSRs).

Predefined user accounts cannot be deleted from the system and are always authenticated natively by the XIV Storage System even if the system operates under LDAP authentication mode.

New user accounts can initially be created by the *admin* user only. After the *admin* user creates a new user account and assigns it to the *storageadmin* (Storage Administrator) role, then other user accounts can be created by this newly created *storageadmin* user.

In native authentication mode, the system is limited to creating up to 32 user accounts. This number includes the predefined users.

User password

The *user password* is a secret word or phrase used by the account owner to gain access to the system. The user password is used at that time of authentication to establish the identity of that user. User passwords can be 6 to 12 characters long, using the characters [a-z][A-Z]~!@#$%^&*()_+-={}|:;<>?,./\[] , and must not include any space between characters. In native authentication mode, the XIV Storage System verifies the validity of a password at the time the password is assigned.

Predefined users have the following default passwords assigned at the time of XIV Storage System installation:

Table 5-1 Default passwords

Predefined user	Default password
admin	adminadmin
technician	predefined and is used only by the IBM XIV Storage System technicians
smis_user	password
xiv_development	predefined and is used only by the IBM XIV development team
xiv_maintenance	predefined and is used only by the IBM XIV maintenance team

Important: As a best practice, the default password should be changed at installation time to prevent unauthorized access to the system

The following restrictions apply when working with passwords in native authentication mode:

- For security purposes, passwords are not shown in user lists.
- Passwords are user changeable. Users can change only their own passwords.
- Only predefined user *admin* can change the passwords of other users.
- Passwords are changeable from both the XCLI and the GUI.
- Passwords are case-sensitive.
- User password assignment is mandatory at the time new user account is created.
- Creating user accounts with empty password or removing password from an existing user account is not permitted.

User roles

There are four predefined user roles (in the XIV GUI and the XCLI. Roles are referred to as *categories* and are used for day to day operation of the XIV Storage System. The following section describes predefined roles, their level of access, and applicable use:

- *storageadmin*

  The *storageadmin* (Storage Administrator) role is the user role with highest level of access available on the system. A user assigned to this role has an ability to perform changes on any system resource except for maintenance of physical components or changing the status of physical components.

- *applicationadmin*

  The *applicationadmin* (Application Administrator) role is designed to provide flexible access control over volume snapshots. User assigned to the *applicationadmin* role can create snapshots of specifically assigned volumes, perform mapping of their own snapshots to a specifically assigned host, and delete their own snapshots. The user group to which an application administrator belongs determines the set of volumes that the application administrator is allowed to manage. If a user group is defined with *access_all="yes"*, application administrators who are members of that group can manage all volumes on the system. For more details on user group membership and group to host association, see “User groups” on page 125.

- *readonly*

  As the name implies, users assigned to the *readonly* role can only view system information. Typical use for the *readonly* role is a user who is responsible for monitoring system status, system reporting, and message logging, and who must not be permitted to make any changes on the system.

- *technician*

  The *technician* role has a single predefined user name (*technician*) assigned to it, and is intended to be used by IBM support personnel for maintaining the physical components of the system. The technician is limited to the following tasks: physical system maintenance and phasing components in or out of service. The technician has extremely restricted access to the system and is unable to perform any configuration changes to pools, volumes, or host definitions on the XIV Storage System.

- *xiv_development*

  The *xiv_development* role has a single predefined user name (*xiv_development*) assigned to it and is intended to be used by IBM development personnel.

- *xiv_maintenance*

  The *xiv_maintenance* role has a single predefined user name (*xiv_maintenance*) assigned to it and is intended to be used by IBM maintenance personnel.

Note: There is no capability to add new user roles or to modify predefined roles. In native authentication mode, after a user is assigned a role, the only way to assign a new role is to first delete the user account and then recreate it.

Table 5-2 Predefined user role assignment

Predefined user	User role
admin	storageadmin
technician	technician
smis_user	readonly
xiv_development	xiv_development
xiv_maintenance	xiv_maintenance

Native authentication mode implements user role mechanism as a form of *Role Based Access Control* (RBAC). Each predefined user role determines the level of system access and associated functions that a user is allowed to use.

Note: The XIV Storage System implements Role Based Access Control (RBAC) based authentication and authorization mechanisms

All user accounts must be assigned to a single user role. Assignment to multiple roles is not permitted. Deleting or modifying role assignment of natively authenticated users is also not permitted.

User groups

A user group is a group of application administrators who share the same set of snapshot creation permissions. The permissions are enforced by associating the user groups with hosts or clusters. User groups have the following characteristics:

- Only users assigned to the *applicationadmin* role can be members of a user group.
- A user can be a member of a single user group.
- A maximum of eight user groups can be created.
- In native authentication mode, a user group can contain up to eight members.
- If a user group is defined with *access_all="yes"*, users assigned to the *applicationadmin* role who are members of that group can manage all snapshots on the system.
- A user must be assigned to the *storageadmin* role to be permitted to create and manage user groups.

Important: A user group membership can only be defined for users assigned to the *applicationadmin* role

User group and host associations

Hosts and clusters can be associated with only a single user group. When a user is a member of a user group that is associated with a host, that user is allowed to manage snapshots of the volumes mapped to that host.

User group and host associations have the following properties:

- User groups can be associated with both hosts and clusters. This enables limiting group member access to specific volumes.
- A host that is part of a cluster can only be associated with a user group through user group to cluster association. Any attempts to create user group association for that host will fail.
- When a host is added to a cluster, the association of that host is removed. Limitations on the management of volumes mapped to the host is controlled by the association of the cluster.
- When a host is removed from a cluster, the association of that cluster remains unchanged. This enables continuity of operations so that all scripts relying on this association will continue to work.

Optional account attributes

In this topic we discuss optional attributes for e-mail and phone numbers:

- *E-mail:* E-mail is used to notify specific users about events through e-mail messages. E-mail addresses must follow standard formatting procedures.

  Acceptable value: Any valid e-mail address. Default value is not defined.
- *Phone and area code:* Phone numbers are used to send SMS messages to notify specific users about system events. Phone numbers and area codes can be a maximum of 63 digits, hyphens (-) and periods (.)

  Acceptable value: Any valid telephone number. Default value is not defined.

5.2.1 Managing user accounts with XIV GUI

This section illustrates the use of the XIV GUI in native authentication mode for creating and managing user accounts, as well for creating user groups and defining group membership.

Adding users with the GUI

The following steps require that you initially log on to the XIV Storage System with storage administrator access rights (*storageadmin* role). If this is the first time that you are accessing the system, use the predefined user *admin* (*default password* `adminadmin`):

1. Open the XIV GUI and log on as shown in Figure 5-1.

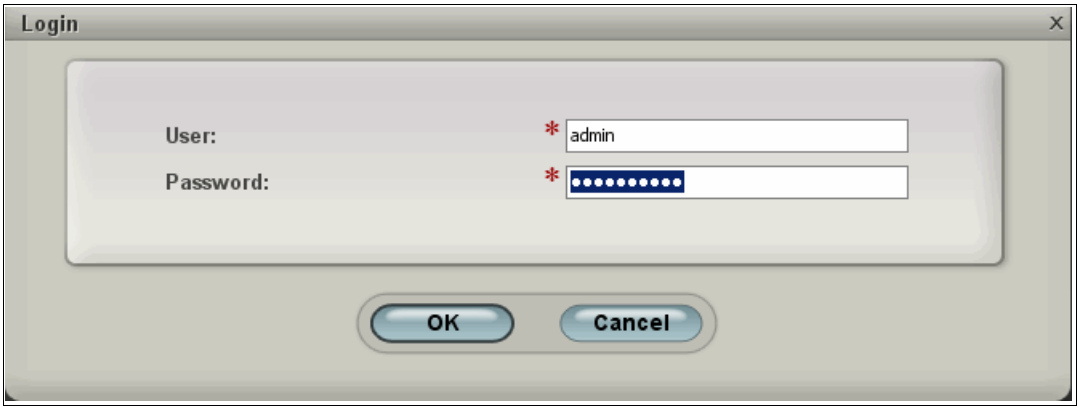


Figure 5-1 GUI Login

2. Users are defined per system. If you manage multiple systems and they have been added to the GUI, select the particular system with which you want to work.
3. In the main Storage Manager GUI window, move the mouse pointer over the padlock icon to display the Access menu. All user access operations can be performed from the Access menu (refer to Figure 5-2). There are three choices:
   - *Users:* Define or change single users
   - *Users Groups:* Define or change user groups, and assign application administrator users to groups
   - *Access Control:* Define or change user groups, and assign application administrator users or user group to hosts
4. Move the mouse over the **Users** menu item (it is now highlighted in yellow) and click it, as shown in Figure 5-2.

Figure 5-2 GUI Access menu

5. The Users window is displayed.

   If the storage system is being accessed for the first time, the window displays the predefined users only. Refer to Figure 5-3 for an example. The default columns are Name, Category, Group, Phone, and E-mail.

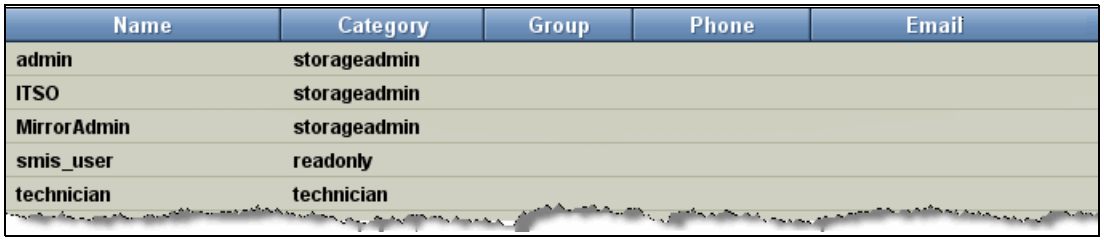


Figure 5-3 GUI Users management

An additional column called Full Access can be displayed (this only applies to users assigned to the *applicationadmin role*). To add the Full Access column, right-click the blue heading bar to display the Customize Columns dialog shown in Figure 5-4.

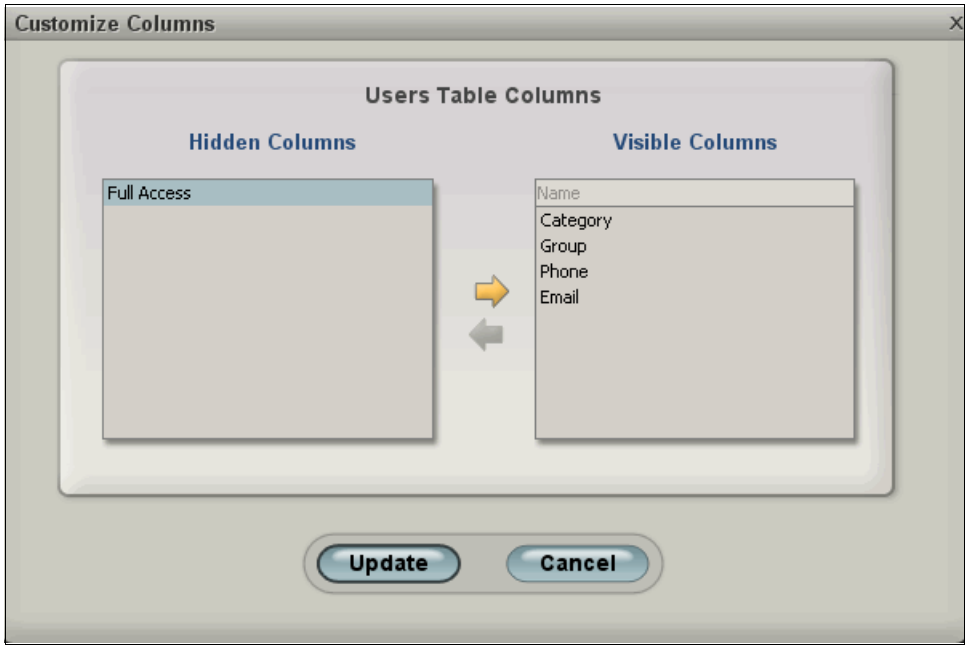


Figure 5-4 Customize columns dialog

a. We recommend that you change the default password for the admin user, which can be accomplished by right-clicking the user name and selecting **Change Password from the context menu**, as illustrated in Figure 5-5.

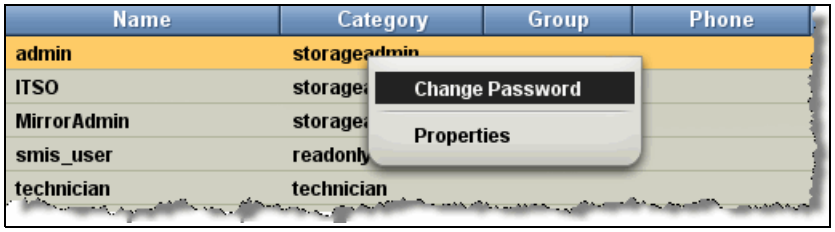


Figure 5-5 GUI admin user change password

6. To add a new user, you can either click the Add icon in the menu bar or right-click the empty space to get the context menu. Both options are visible in Figure 5-6. Click **Add User**.

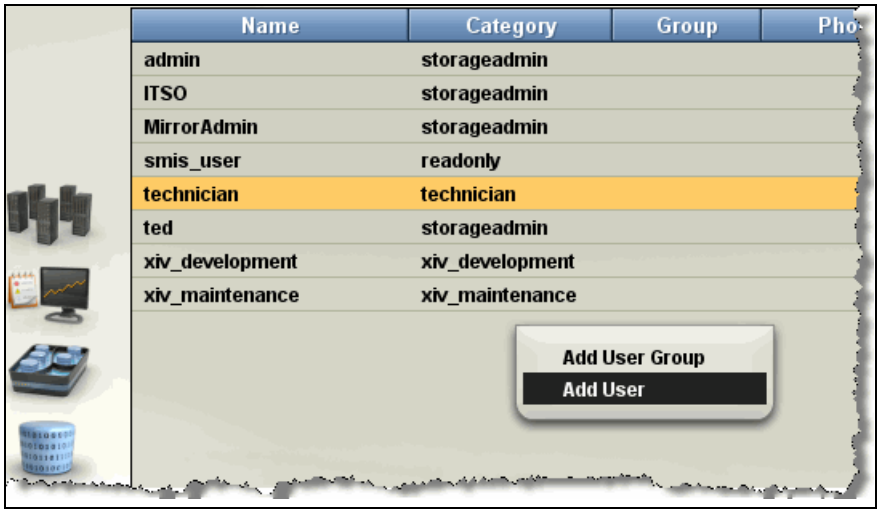


Figure 5-6 GUI Add User option

7. The Define User dialog is displayed. A *user* is defined by a unique name and a password (refer to Figure 5-7). The default role (denoted as Category in the dialog panel) is Storage Administrator. Category must be assigned. Optionally, enter the e-mail address and phone number for the user. Click **Add** to create the user and return to the Users window.

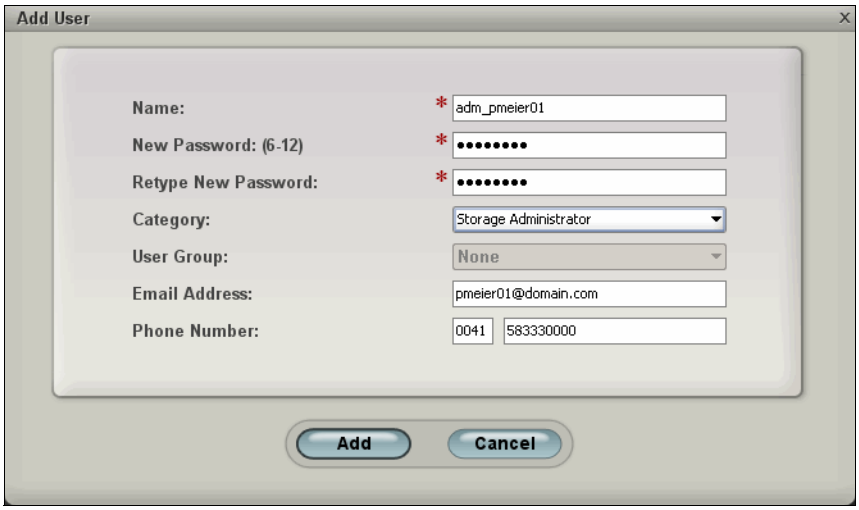


Figure 5-7 GUI Define User attributes

8. If you need to test the user that you just defined, click the current user name shown in the upper right corner of the IBM XIV Storage Manager window (Figure 5-8), which allows you to log in as a new user.

Figure 5-8 GUI quick user change

Defining user groups with the GUI

The IBM XIV Storage system can simplify user management tasks with the capability to create user groups. User groups only apply to users assigned to the *applicationadmin role*.

A user group can also be associated with one or multiple hosts or clusters.

The following steps illustrate how to create user groups, add users (with application administrator role) to the group, and how to define host associations for the group:

1. Be sure to log in as `admin` (or another user with storage administrator rights). From the Access menu, click **Users Groups** as shown in Figure 5-9. In our scenario, we create a user group called `EXCHANGE CLUSTER 01`. As shown in Figure 5-9, the user groups can be accessed from the *Access* menu (padlock icon).

Figure 5-9 Select Users Groups

2. The Users Groups window displays. To add a new user group, either click the Add User Group icon (shown in Figure 5-10) in the menu bar, or right-click in an empty area of the User Groups table and select **Add User Group** from the context menu as shown in Figure 5-10.

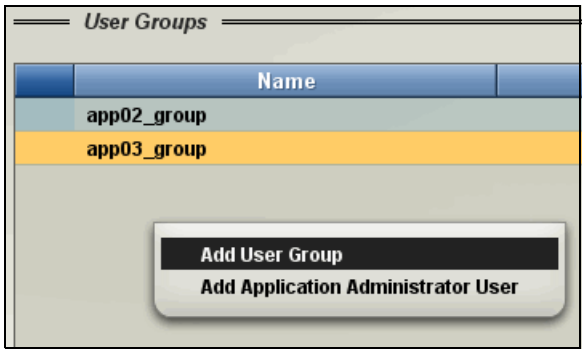


Figure 5-10 Add User Group

3. The Create User Group dialog displays. Enter a meaningful group name and click **Add** (refer to Figure 5-11).

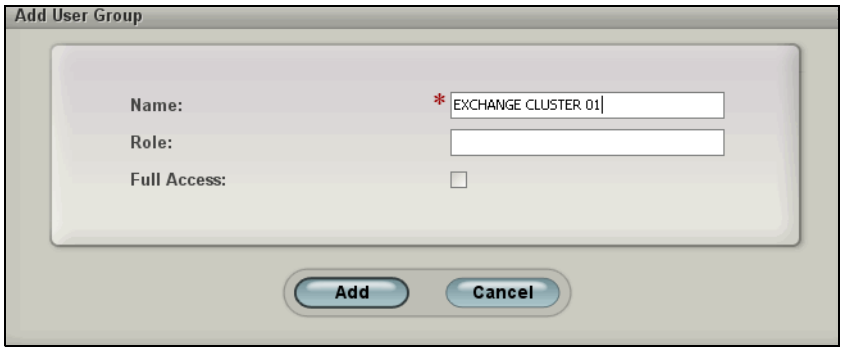


Figure 5-11 Enter New User Group Name

Note: The role field is not applicable to user group definition in native authentication mode and will have no effect even if a value is entered.

If a user group has the *Full Access* flag turned on, all members of that group will have unrestricted access to all snapshots on the system.

4. At this stage, the user group `EXCHANGE CLUSTER 01` still has no members and no associations defined. Next, we create an association between a host and the user group. Select **Access Control** from the Access menu as shown in Figure 5-12. The Access Control window appears.

Figure 5-12 Access Control

5. Right-click the name of the user group that you have created to bring up a context menu and select **Updating Access Control** as shown in Figure 5-13.

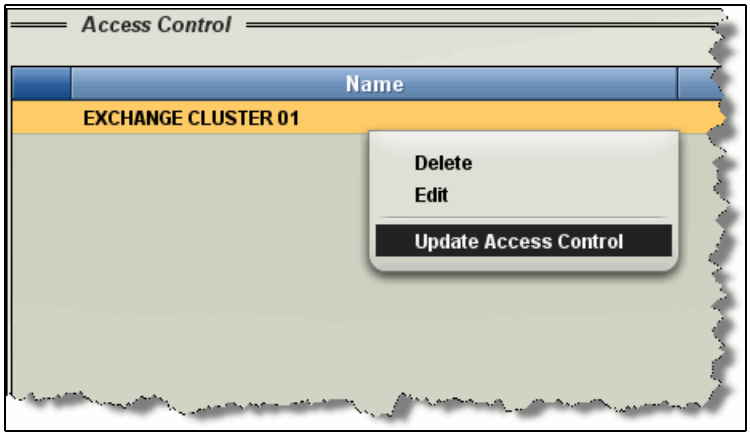


Figure 5-13 Update Access Control for a user group

6. The *User Group Access Control* dialog that is shown in Figure 5-14 is displayed. The panel contains the names of all the hosts and clusters defined to the XIV Storage System. The left pane displays the list of Unauthorized Hosts/Clusters for this particular user group and the right pane shows the list of hosts that have already been associated to the user group. You can add or remove hosts from either list by selecting a host and clicking the appropriate arrow. Finally, click **Update** to save the changes.

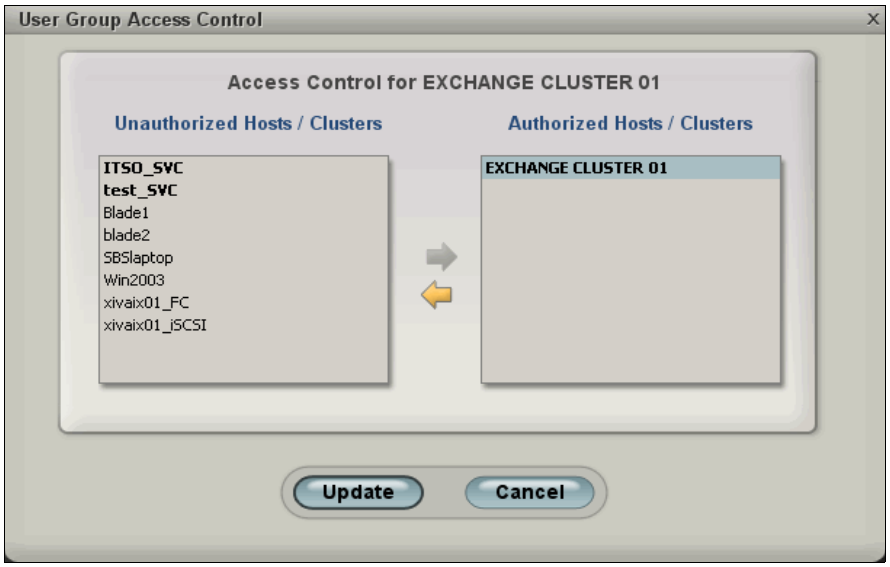


Figure 5-14 Access Control Definitions panel

7. After a host (or multiple hosts) have been associated with a user group, you can define user membership for the user group (remember that a user must have the application administrator role to be added to a user group). Go to the *Users* window and right-click the user name to display the context menu. From the context menu (refer to Figure 5-15), select **Add to Group** to add this user to a group.

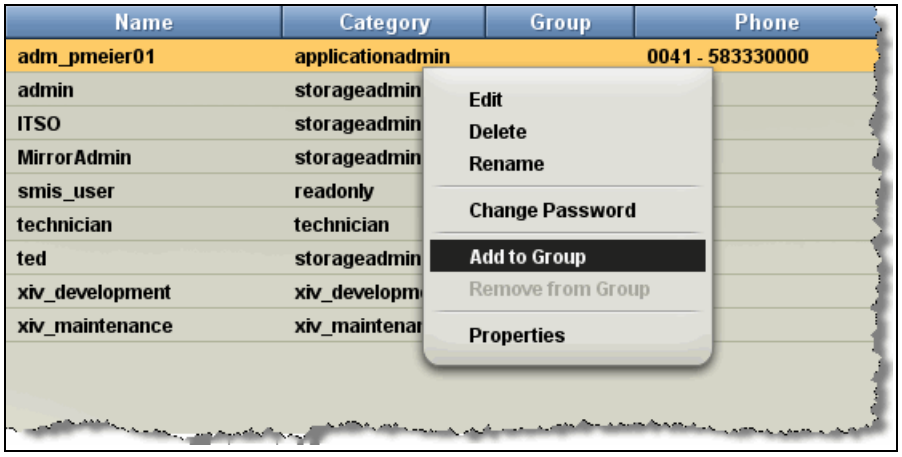


Figure 5-15 Add a user to a group

8. The *Select User Group* dialog is displayed. Select the desired group from the pull-down list and click **OK** (refer to Figure 5-16).

Figure 5-16 Select User Group

9. The user `adm_pmeier01` has been added as a member to the user group `EXCHANGE CLUSTER 01` in this example. You can verify this group membership in the Users panel as shown in Figure 5-17.

Name	Category	Group	Phone
adm_pmeier01	applicationadmin	EXCHANGE CLUSTER 01	0041 - 583330000

Figure 5-17 View user group membership

10. The user `adm_pmeier01` is an applicationadmin with the Full Access right set to `no`. This user can now perform snapshots of the EXCHANGE CLUSTER 01 volumes. Because the EXCHANGE CLUSTER 01 is the only cluster associated with the group, `adm_pmeier01` is only allowed to map those snapshots to the EXCHANGE CLUSTER 01. However, you can add another host or a cluster, such as a test or backup host, to allow `adm_pmeier01` to map a snapshot volume to a test or backup server.

5.2.2 Managing user accounts using XCLI

This section summarizes the commands and options available to manage user accounts, user roles, user groups, group memberships, and user group to host associations through the XCLI user interface.

Table 5-3 shows the various commands and a brief description for each command. The table also indicates the user role required to issue specific commands.

Table 5-3 XCLI access control commands

Command	Description	Role required to use command
access_define	Defines an association between a user group and a host.	storageadmin
access_delete	Deletes an access control definition.	storageadmin
access_list	Lists access control definitions.	storageadmin, readonly, and applicationadmin
user_define	Defines a new user.	storageadmin
user_delete	Deletes a user.	storageadmin
user_group_add_user	Adds a user to a user group.	storageadmin
user_group_create	Creates a user group.	storageadmin
user_group_delete	Deletes a user group.	storageadmin
user_group_list	Lists all user groups or a specific one.	storageadmin, readonly, and applicationadmin
user_group_remove_user	Removes a user from a user group.	storageadmin
user_group_rename	Renames a user group.	storageadmin
user_list	Lists all users or a specific user.	storageadmin, readonly, and applicationadmin
user_rename	Renames a user.	storageadmin
user_update	Updates a user. You can rename the user, change password, modify the Access All setting, modify e-mail, area code, and/or phone number.	storageadmin, and applicationadmin

Adding users with the XCLI

To perform the following steps, the XCLI component must be installed on the management workstation, and a storageadmin user is required. The following examples assume a Windows- based management workstation:

1. Open a Windows command prompt and execute the command **`xcli -L`** to see the registered managed systems. In Example 5-1, there are two IBM XIV Storage systems registered. The configuration is saved with the serial number as the system name.

Example 5-1 XCLI List managed systems

```
C:\>xcli -L
System   Managements IPs
MN00050  9.155.56.100, 9.155.56.101, 9.155.56.102
1300203  9.155.56.58, 9.155.56.56, 9.155.56.57
```

2. In Example 5-2, we start an XCLI session with a particular system with which we want to work and execute the **state_list** command.

Example 5-2 XCLI state_list

```
>> state_list
Command completed successfully
Category          Value
shutdown_reason   No Shutdown
target_state      on
off_type          off
redundancy_status Full Redundancy
system_state      on
safe_mode         no
```

3. XCLI commands are grouped into categories. The **help** command can be used to get a list of all commands related to the category `accesscontrol`. Example 5-3 is a subset of the `accesscontrol` category commands that can be used for account management in native authentication mode. Commands applicable to LDAP authentication mode only are not included.

Example 5-3 Native authentication mode XCLI accesscontrol commands

```
Name                    Description
access_define           Defines an association between a user group and a host.
access_delete           Deletes an access control definition.
access_list             Lists access control definitions.
user_define             Defines a new user.
user_delete             Deletes a user.
user_group_add_user     Adds a user to a user group.
user_group_create       Creates a user group.
user_group_delete       Deletes a user group.
user_group_list         Lists all user groups or a specific one.
user_group_remove_user  Removes a user from a user group.
user_group_rename       Renames a user group.
user_group_update       Updates a user group.
user_list               Lists all users or a specific user.
user_rename             Renames a user.
user_update             Updates a user.
```

4. Use the **user_list** command to obtain the list of predefined users and categories as shown in Example 5-4. This example assumes that no users, other than the default users, have been added to the system.

Example 5-4 XCLI user_list

```
>> user_list
Name           Category         Group/Active/EmailAddress/Phone/AccessAll
admin          storageadmin              yes
technician     technician                yes
smis_user      readonly                  yes
ted            storageadmin              yes
ITSO           storageadmin              yes
MirrorAdmin    storageadmin              yes
adm_pmeier01   applicationadmin    EXCHANGE CLUSTER 01   yes
pmeier01@domain.com   0041        583330000      no
```

5. If this is a new system, you must change the default passwords for obvious security reasons. Use the **update_user** command as shown in Example 5-5 for the user technician.

Example 5-5 XCLI user_update

```
>> user_update user=technician password=dOItNOW password_verify=dOItNOW
Command completed successfully
```

6. Adding a new user is straightforward as shown in Example 5-6. A user is defined by a unique name, password, and role (designated here as category).

Example 5-6 XCLI user_define

```
>> user_define user=adm_itso02 password=wr1teFASTER password_verify=wr1teFASTER
category=storageadmin
Command completed successfully
```

7. Example 5-7 shows a quick test to verify that the new user can log on.

Example 5-7 XCLI user_list

```
C>> user_list name=adm_itso02
Name             Category       Group   Active Email Address/Area Code...
adm_itso02       storageadmin            yes
```

Defining user groups with the XCLI

To use the GUI to define user groups:

1. Use the **user_group_create** command as shown in Example 5-8 to create a user group called EXCHANGE_CLUSTER_01.

Example 5-8 XCLI user_group_create

```
>> user_group_create user_group=EXCHANGE_CLUSTER_01
Command completed successfully
```

Note: Avoid spaces in user group names. If spaces are required, the group name must be placed between single quotation marks, such as 'name with spaces'.

2. The user group EXCHANGE_CLUSTER_01 is empty and has no associated host. The next step is to associate a host or cluster. In Example 5-9, user group EXCHANGE_CLUSTER_01 is associated to EXCHANGE_CLUSTER_MAINZ.

Example 5-9 XCLI access_define

```
>> access_define user_group="EXCHANGE_CLUSTER_01"
cluster="EXCHANGE_CLUSTER_MAINZ"
Command completed successfully
```

3. A host has been assigned to the user group. The user group still does not have any users included. In Example 5-10, we add the first user.

Example 5-10 XCLI user_group_add_user

```
>> user_group_add_user user_group="EXCHANGE_CLUSTER_01" user="adm_mike02"
Command completed successfully
```

4. The user adm_mike02 has been assigned to the user group EXCHANGE_CLUSTER_ 01. You can verify the assignment by using the XCLI **user_list** command as shown in Example 5-11.

Example 5-11 XCLI user_list

```
>> user_list
Name             Category         Group Access All
xiv_development  xiv_development
xiv_maintenance  xiv_maintenance
admin            storageadmin
technician       technician
adm_itso02       storageadmin
adm_mike02       applicationadmin EXCHANGE_CLUSTER_01 no
```

The user adm_mike02 is an applicationadmin with the Full Access right set to no. This user can now perform snapshots of the EXCHANGE_CLUSTER_01 volumes.

Because EXCHANGE_CLUSTER_01 is the only cluster (or host) in the group, adm_mike02 is only allowed to map those snapshots to the same EXCHANGE_CLUSTER_01. This is not useful in practice and is not supported in most cases. Most servers (operating systems) cannot handle having two disks with the same metadata mapped to the system. In order to prevent issues with the server, you need to map the snapshot to another host, not the host to which the master volume is mapped.

Therefore, to make things practical, a user group is typically associated to more than one host.

5.2.3 Password management

Password management in native authentication mode is internal to the XIV Storage System. The XIV system has no built-in password management rules such as password expiration, preventing reuse of the same passwords and/or password strength verification. Furthermore, if you want to log on to multiple systems at any given time through the GUI, your must be registered with the same password on all the XIV systems.

Password resets

In native authentication mode as long as users can log in, they can change their own passwords.

The predefined user admin is the only user is authorized to change other users passwords. Direct access to user credential repository is not permitted. System security is enforced by allowing password changes only through XIV GUI and XCLI.

Figure 5-18 shows that you can change a password by right-clicking the selected user in the Users window. Then, select **Change Password from the context menu**.

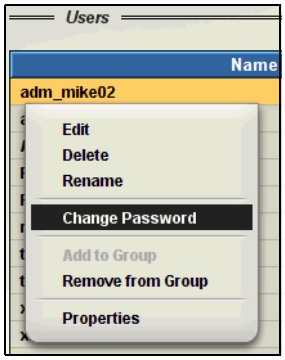


Figure 5-18 GUI change password context menu

The Change Password dialog shown in Figure 5-19 is displayed. Enter the New Password and then retype it for verification in the appropriate field (remember that only alphanumeric characters are allowed). Click **Update**.

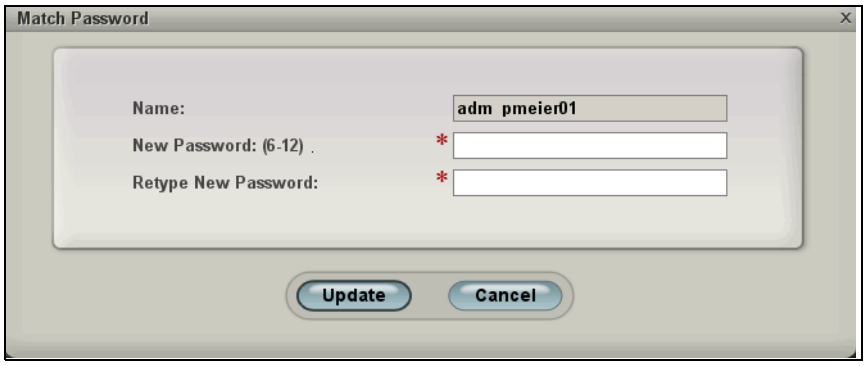


Figure 5-19 GUI Change Password window

Example 5-12 shows the same password change procedure using the XCLI. Remember that a user with the *storageadmin* role is required to change the password on behalf of another user.

Example 5-12 XCLI change user password

```
>> user_update user=adm_mike02 password=workLESS password_verify=workLESS
Command completed successfully
```

5.2.4 Managing multiple systems

Managing multiple XIV Storage Systems is straightforward in native authentication mode. Due to the fact that user credentials are stored locally on every XIV system, it is key to keep the same user name and password on different XIV Storage Systems to allow for quick transitions between systems in the XIV GUI. This approach is especially useful in Remote Mirror configurations, where the storage administrator is required to switch from source to target system.

Figure 5-20 illustrates the GUI view of multiple systems when using non-synchronized passwords. For this example, the system named ARCXIVJEMT1 has a user account `xivtestuser2` that provides the storage admin level of access. Because the tester ID is not configured for the other XIV Storage Systems, only the ARCXIVJEMT1 system is currently shown as accessible.

The user can see the other systems, but is unable to access them with the `xivtestuser2` user name (the unauthorized systems appear in black and white). They also state that the user is unknown. If the system has the `xivtestuser2` defined with a different password, the systems are still displayed in the same state.

Figure 5-20 Single User Login

In order to allow simultaneous access to multiple systems, the simplest approach is to have corresponding passwords manually synchronized among those systems. Figure 5-21 illustrates the use of user account with passwords synchronized among four XIV systems. The storage administrator can easily switch between these systems for the activities without having to log on each time with a different password. The XIV system where the user was successfully authenticated is now displayed in color with an indication of its status.

Figure 5-21 Manual user password synchronization among multiple XIV systems

5.3 LDAP managed user authentication

Starting with code level 10.1, the XIV Storage System offers the capability to use LDAP server based user authentication (the previous version of code only supported XIV native authentication mode).

When LDAP authentication is enabled, the XIV system accesses a specified LDAP directory to authenticate users whose credentials are maintained in the LDAP directory (with the exception of the admin, technician, development and SMIS_user which remain locally administered and maintained).

The benefits of an LDAP based centralized user management can be substantial when considering the size and complexity of the overall IT environment. Maintaining local user credentials repositories is relatively straightforward and convenient when only dealing with a small number of users and a small number storage systems. However, as the number of users and interconnected systems grows, the complexity of user account management rapidly increases and managing such an environment becomes a time consuming task.

In this section, we review some of the benefits of this approach. Although the benefits from utilizing LDAP are significant, you must also evaluate the considerable planning effort and complexity of deploying LDAP infrastructure, if it is not already in place.

5.3.1 Introduction to LDAP

The Lightweight Directory Access Protocol (LDAP) is an open industry standard that defines a standard method for accessing and updating information in a directory. It is being supported by a growing number of software vendors and is being incorporated into a growing number of products and applications.

A directory is a listing of information about objects arranged in some order that gives details about each object. Common examples are a city telephone directory and a library card catalog. In computer terms, a directory is a specialized database, also called a data repository, that stores typed and ordered information about objects. A particular directory might list information about users (the objects) consisting of typed information such as user names, passwords, e-mail addresses and so on. Directories allow users or applications to find resources that have the characteristics needed for a particular task.

Directories in LDAP are accessed using the client/server model. An application that wants to read or write information in a directory does not access the directory directly, but uses a set of programs or APIs that cause a message to be sent from LDAP client to LDAP server. LDAP server retrieves the information on behalf of the client application and returns the requested information if the client has permission to see the information. LDAP defines a message protocol used between the LDAP clients and the LDAP directory servers. This includes methods to search for information, read information, and to update information based on permissions.

LDAP-enabled directories have become a popular choice for storing and managing user access information. LDAP provides a centralized data repository of user access information that can be securely accessed over the network. It allows the system administrators to manage the users from multiple XIV Storage Systems in one central directory.

5.3.2 LDAP directory components

An LDAP directory is a collection of objects organized in a tree structure. The LDAP naming model defines how objects are identified and organized. Objects are organized in a tree-like structure called the Directory Information Tree (DIT). Objects are arranged within the DIT based on their distinguished name (DN). Distinguished name (DN) defines location of an object within DIT. Each object is also referred to as an entry in a directory belonging to an object classes. An object class describes the content and purpose of the object. It also contains a list of attributes, such as a telephone number or surname, that can be defined in an object of that object class.

As shown in Figure 5-22, the object with the distinguished name (DN) `cn=mbarlen, ou=Marketing, o=IBM` belongs to object class `objectClass=ePerson.`

`Object class ePerson` contains attributes:
`cn` (common name),
`mail`,
`sn` (surname),
`givenName`
`telephoneNumber.`

Each attribute has a value assigned to it:
`cn=mbarlen,`
`mail=marion@ibm.com,`
`sn=Barlen,`
`givenName=Marion,`
`telephoneNumber=112.`

In this example, the object represents a single employee record. If a record for a new employee in organizational unit (ou) Marketing of organization (o) IBM needs to be created, the same location in DIT will be the same, `ou=Marketing, o=IBM,` and the same set of attributes defined by objectClass `ePerson` will also be used. The new object will be defined using its own set of attribute values because the new employee will have their own name, e-mail address, phone number, and so on.

For more information about the directory components, refer to the IBM Redbooks publication, *Understanding LDAP - Design and Implementation,* SG24-4986.

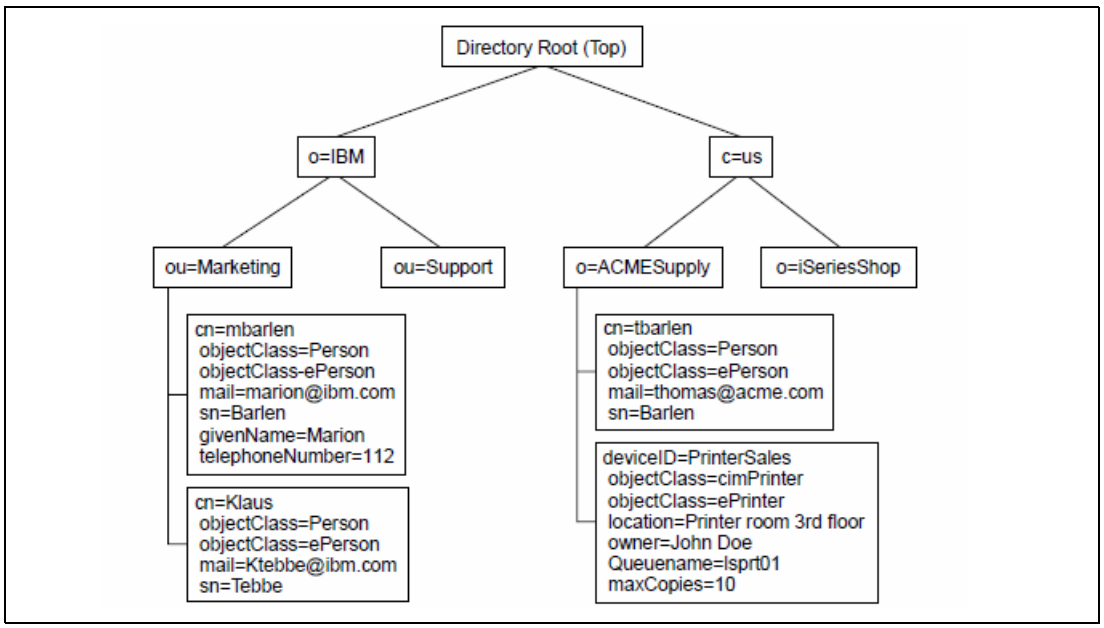


Figure 5-22 Example of a Directory Information Tree (DIT)

All the objects and attributes with their characteristics are defined in a schema. The schema specifies what can be stored in the directory.

5.3.3 LDAP product selection

LDAP Authentication for version 10.1 of the XIV Storage System supports two LDAP server products:

- Microsoft Active Directory
- Sun Java Services Directory Server Enterprise Edition

The current skill set (of your IT staff) is always an important consideration when choosing any products for centralized user authentication. If you have skills in running a particular directory server, then it might be a wise choice to standardize on this server because your skilled people will best be able to customize and tune the server as well as to provide the most reliable and highly-available implementation for the LDAP infrastructure.

Microsoft Active Directory might be a better choice for an enterprise with most of its infrastructure components deployed using Microsoft Windows operating system. Sun Java Services Directory Server Enterprise Edition on the other hand provides support for UNIX-like operating systems including Linux, as well as Microsoft Windows.

All LDAP servers share many basic characteristics because they are based on the industry standard Request for Comments (RFCs). However, because of implementation differences, they are not always entirely compatible with each other. For more information about RFCs, particularly regarding LDAP RFC 4510-4533, see the following Web page:

http://www.ietf.org/rfc.html

Current implementation of LDAP based user authentication for XIV does not support connectivity to multiple LDAP server of different types. However, it is possible to configure an XIV Storage System to use multiple LDAP servers of the same type to eliminate a single point of failure. The IBM XIV system will support communication with a single LDAP server at a time. The LDAP authentication configuration allows specification of multiple LDAP servers that the IBM XIV Storage System can connect to if a given LDAP server is inaccessible.

5.3.4 LDAP login process overview

The XIV login process when LDAP authentication is enabled is depicted in Figure 5-23:

1. The XIV user login process starts with the user launching a new XIV or GUI session and submitting credentials (user name and password) to the XIV system (step “1”).
2. In step “2” the XIV system logs into a previously defined LDAP server using the credentials provided by the user. If login to the LDAP server fails, a corresponding error message is returned to the user and the login process terminates.
3. If XIV successfully logs into the LDAP server, it retrieves attributes for establishing LDAP role mapping (step “3”). If the XIV system cannot establish LDAP role mapping, the user login process terminates and a corresponding error message is returned to the user.
4. If LDAP role mapping is successfully established, XIV creates a new session and returns a prompt to the user (step “4”). For more information about LDAP role mapping, see 5.3.5, “LDAP role mapping”

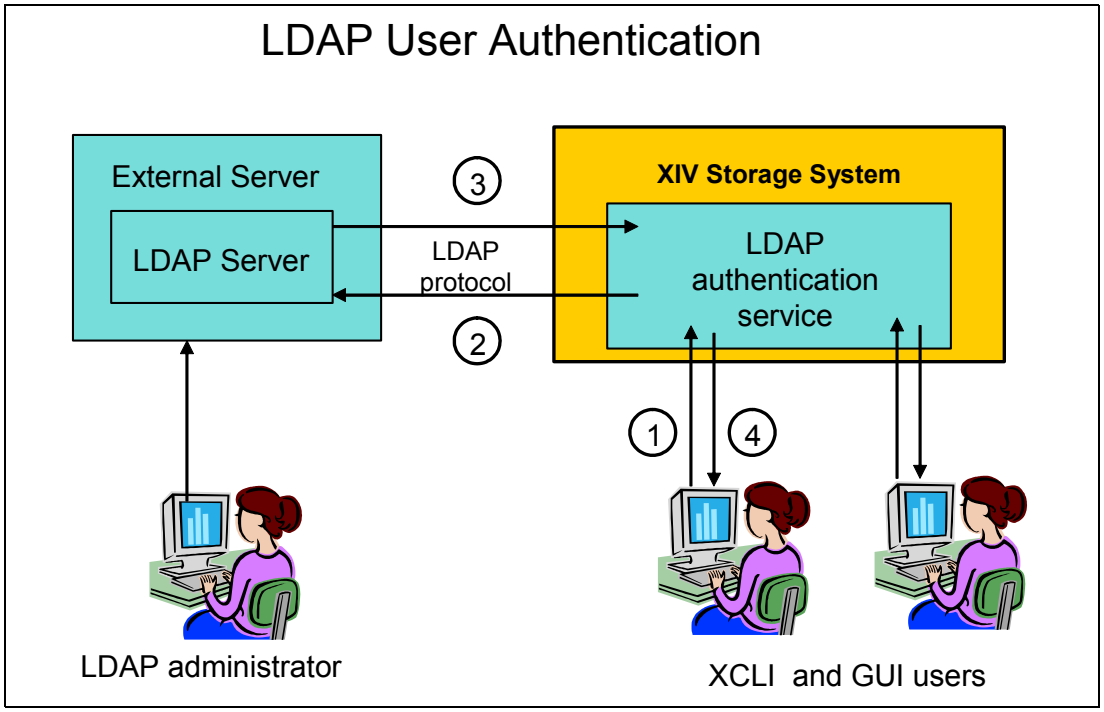


Figure 5-23 LDAP authentication process overview

5.3.5 LDAP role mapping

Before any LDAP user can be granted access to XIV, the user must be assigned to one of the supported user roles. XIV uses the *storageadmin, readonly* and *applicationadmin* predefined roles. The mechanism used for determining what role a particular user is assigned to is called role mapping. In native mode a role is explicitly assigned to a user at the time of *user account* creation. In LDAP mode, the role of a specific user is determined at the time the user logs in to the XIV system. This process is called role mapping.

When initially planning to use LDAP based authentication with XIV, the LDAP server administrator has to make a decision as to which LDAP attribute can be used for role mapping. As discussed in 5.3.2, “LDAP directory components” on page 141 each LDAP object has a number of associated attributes. The type of LDAP object classes used to create user account for XIV authentication depends on the type of LDAP server being used.

The SUN Java Directory server uses the `inetOrgPerson` LDAP object class, and Active Directory uses the `organizationalPerson` LDAP object class for definition of user accounts for XIV authentication.

A detailed definition of the `inetOrgPerson` LDAP object class and list of attributes can found at the Internet FAQ Archive Web site:

`http://www.faqs.org/rfcs/rfc2798.html`

A detailed definition of the `organizationalPerson` LDAP object class and list of attributes can found at the Microsoft Web site:

`http://msdn.microsoft.com/en-us/library/ms683883(VS.85).aspx`

The designated attribute (as established by the LDAP administrator) will be used for storing a value that will map a user to a role. In our examples we used the *description* attribute for the purpose of role mapping. Ultimately the decision on what attribute is to be used for role mapping should be left to the LDAP administrator. If the *description* attribute is already used for something else, then the LDAP administrator has to designate a different attribute. For the purpose of illustration, we assume that the LDAP administrator agrees to use the *description* attribute.

The XIV administrator now can make the necessary configuration change to instruct the XIV system to use *description* as the attribute name. This is done by assigning *description* value to the *`xiv_group_attrib`* configuration parameter, using `ldap_config_set` XCLI command:

```
ldap_config_set xiv_group_attrib=description
```

Next, the XIV administrator defines two names that will be used as role names in LDAP. In our example the XIV administrator uses “Storage Administrator” and “Read Only” names for mapping to *storageadmin* and *readonly* roles. XIV administrator sets *corresponding parameters in XIV system*

```
ldap_config_set storage_admin_role="Storage Administrator"
ldap_config_set read_only_role="Read Only"
```

Note: The LDAP server uses case-insensitive string matching for the *description* attribute value. For example, “`Storage Administrator`”, “`storage administrator`” and “`STORAGE ADMINISTRATOR`” will be recognized as equal strings. To simplify XIV system administration, however, we recommend treating both the XIV configuration parameter and LDAP attribute value as if they were case-sensitive and assign “`Storage Administrator`” value to both.

“`Storage Administrator`” and “`Read Only`” names were used simply because both strings can be easily associated with their corresponding XIV roles: `storageadmin` and `readonly` respectively. It is not necessary to use the same names in your XIV system configuration.

However, if you were to change these parameters, consider using names that are self descriptive and easy to remember, in order to simplify the LDAP server administration tasks. Every time the LDAP server administrator will be creating a new XIV account, one of the names will have to be entered as a description attribute value (except for the *applicationadmin* role, as we explain later). After being configured in both XIV and LDAP, changing these parameters, although possible, can potentially be time consuming due to the fact that each existing LDAP account will have to be changed individually, to reflect the new attribute value.

These configuration tasks can also be done from the XIV GUI. On the main XIV Storage Management panel, select **Configure System**, and select the **LDAP** panel on the left panel. Enter `description` in the XIV Group Attribute field, `Read Only` in the Read Only Role field, and `Storage Administrator` in the Storage Admin Role field as shown in Figure 5-24. Finally, click **Update** to save the changes.

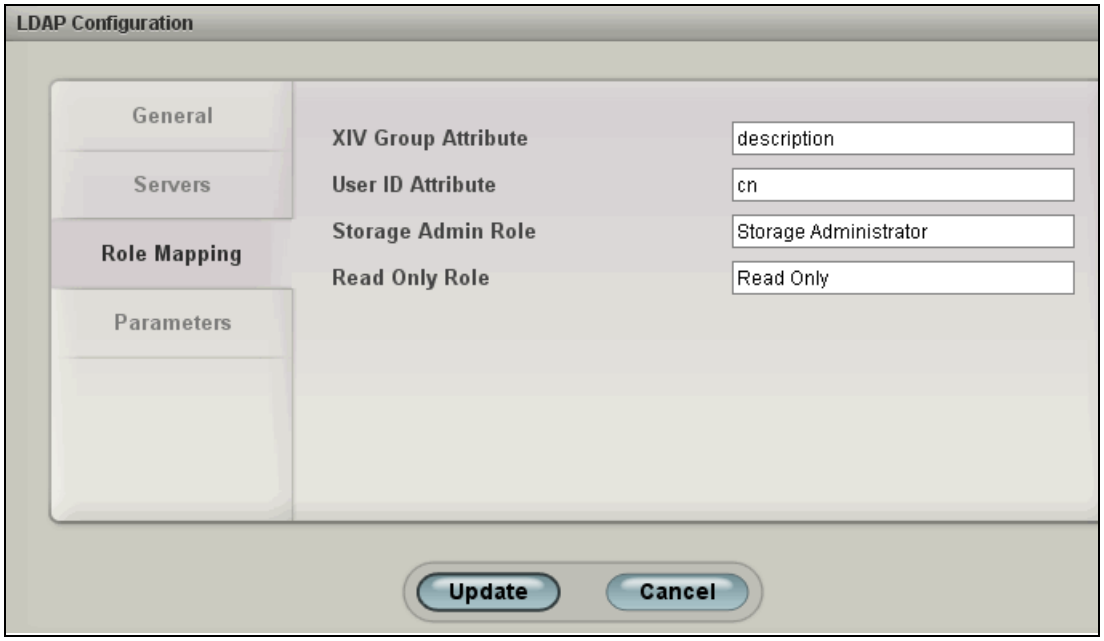


Figure 5-24 Configuring XIV Group Attribute and role parameters

The XIV administrator informs the LDAP administrator that the *Storage Administrator* and *Read Only* names have to be used for role mapping:

- The LDAP administrator creates a user account in LDAP and assigns the “Storage Administrator” value to the *description* attribute. When the newly created user logs in to the system, XIV performs the role mapping as depicted in Figure 5-25.

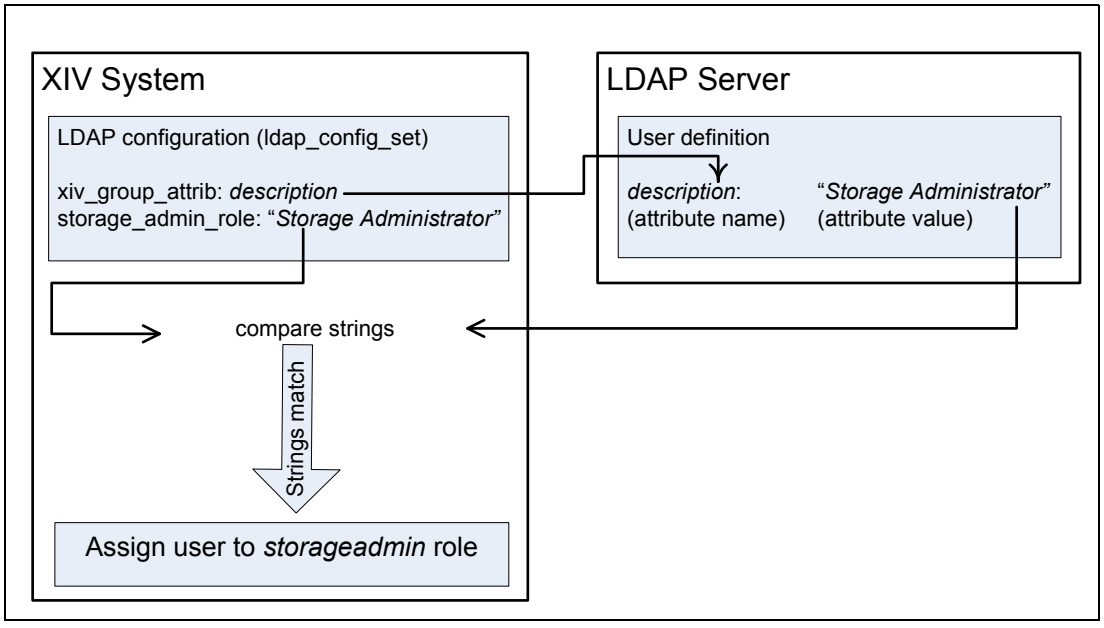


Figure 5-25 Assigning LDAP authenticated user to storageadmin role

- The LDAP administrator creates an account and assigns *Read Only* to the *description* attribute. The newly created user is assigned to the *readonly* role. When this user logs into the XIV system, XIV performs the role mapping as depicted in Figure 5-25.

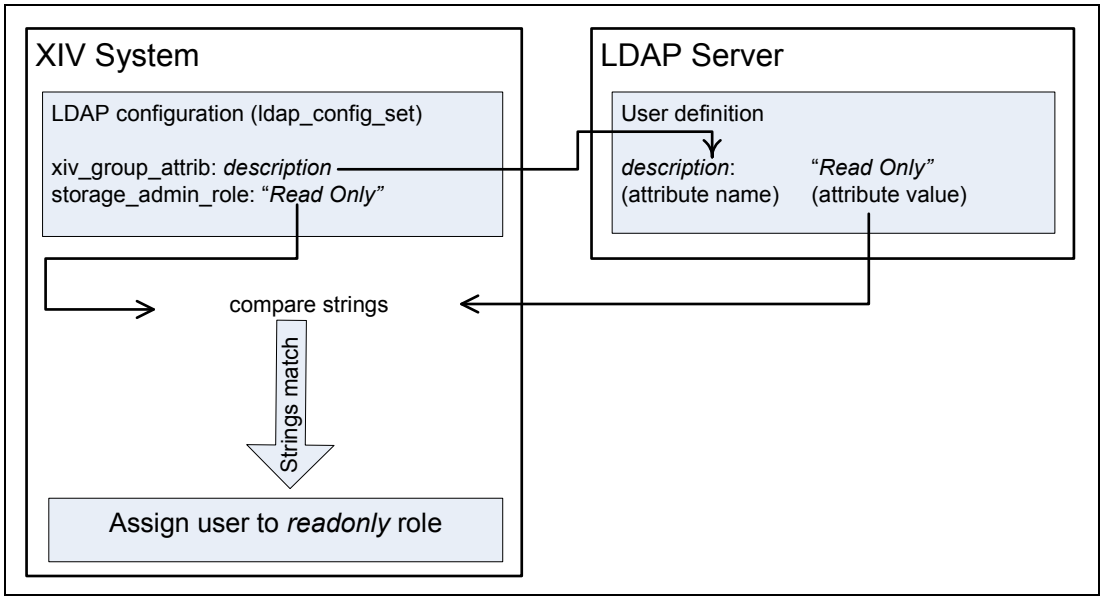


Figure 5-26 Assigning LDAP authenticated user to readonly role

LDAP role mapping for applicationadmin

The LDAP account can also be assigned to an *applicationadmin* role, but the mechanism of creating role mapping in this case is different than the one used for *storageadmin* and *readonly* role mapping.

The XIV system will assign a user to the *applicationadmin* role if it can match the value of the *description* attribute with the *ldap_role* parameter of any user groups defined in XIV. If an account is assigned the *applicationadmin* role, it also becomes a membership of the user group whose *ldap_role* parameter matches the value of the user's *description* attribute.

The user group must be created before the user logs in to the system, otherwise the login will fail. The XIV administrator creates a user group, with the **user_group_create** XCLI command, as follows:

```
user_group_create user_group=app01_group ldap_role=app01_administrator
```

After the LDAP administrator has created the user account and assigned the *app01_administrator* value to the *description attribute*, the user can be authenticated by the XIV system. The role assignment and group membership inheritance for a newly created user is depicted in Figure 5-27.

If the XIV system cannot find a match for the value assigned to the *description* attribute of a user, then the user is denied system access.

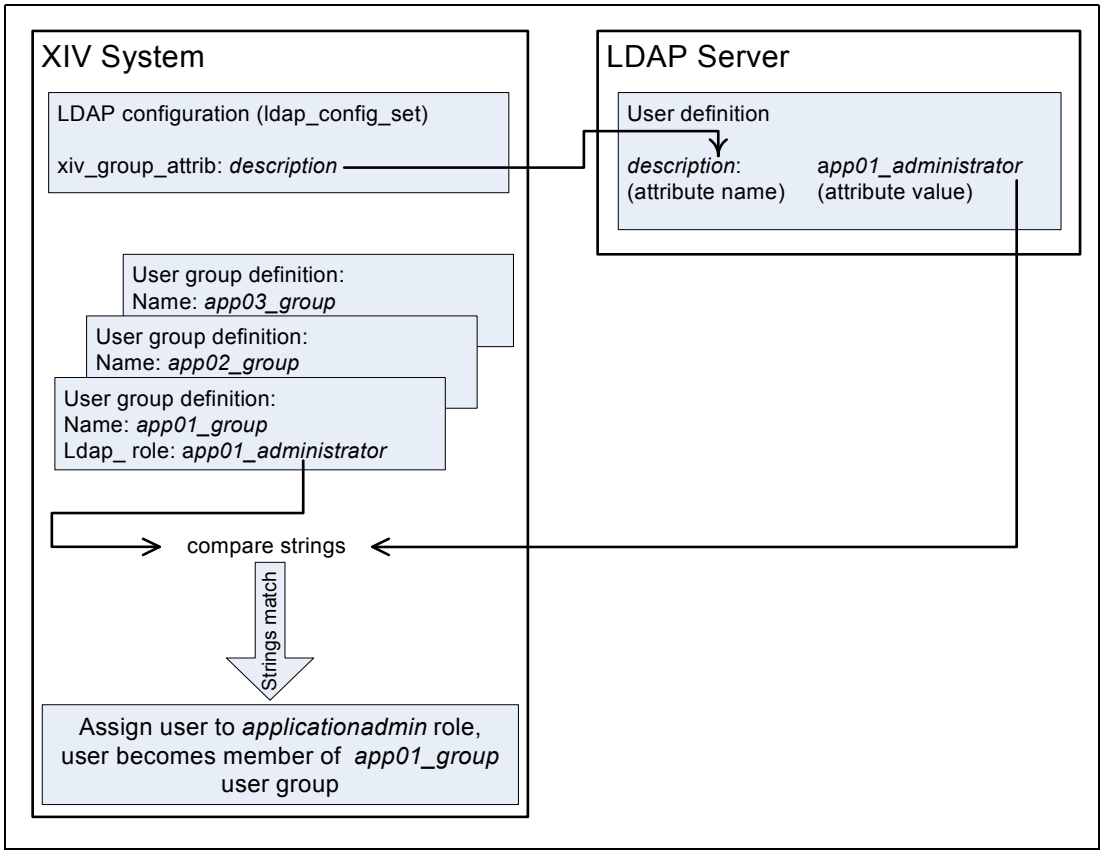


Figure 5-27 Assigning LDAP authenticated user to applicationadmin role

Table 5-4 summarizes the LDAP to XIV role mappings.

Table 5-4 LDAP role mapping summary

XIV role	XIV configuration parameter name	XIV configuration parameter setting	LDAP attribute value
storageadmin	storage_admin_role	Storage Administrator	Storage Administrator
readonly	read_only_role	Read Only	Read Only
applicationadmin	N/A	N/A	arbitrary (must match *ldap_role* field in XIV user group)

5.3.6 Configuring XIV for LDAP authentication

By default, XIV is configured to use native authentication mode, and LDAP authentication mode is inactive. The LDAP server needs to be configured, tested, and fully operational before enabling LDAP authentication mode on an XIV system configured to use that LDAP server. Properly designing and deploying Microsoft Active Directory or SUN Java Directory is an involved process that must consider corporate structure, security policies, organizational boundaries, existing IT infrastructure, and many other factors.

Designing an enterprise class LDAP solution is beyond the scope of this book. Appendix A, “Additional LDAP information” on page 355 provides an example of installation and configuration procedures for Microsoft Active Directory and SUN Java Directory.

Before an LDAP directory can be populated with the XIV user credentials information, some coordination is required between the LDAP server administrator and XIV system administrators. They need to concur on the names and designate an attribute to be used for LDAP role mapping.

The XCLI command sample in Example 5-13 illustrates the set of LDAP-related configuration parameters on the XIV system.

Example 5-13 Default LDAP configuration parameters

```
>> ldap_config_get
Name                      Value
base_dn
xiv_group_attrib
third_expiration_event    7
version                   3
user_id_attrib            objectSiD
current_server
use_ssl                   no
session_cache_period
second_expiration_event   14
read_only_role
storage_admin_role
first_expiration_event    30
bind_time_limit           0

>> ldap_list_servers
No LDAP servers are defined in the system

>> ldap_mode_get
Mode
Inactive
```

Before the LDAP mode can be activated on the XIV system, all of the LDAP configuration parameters need to have an assigned value. This starts with configuring user authentication against Microsoft Active Directory or SUN Java Directory LDAP server.

As a first step in the configuration process, the LDAP server administrator needs to provide a the fully qualified domain name (fqdn) and the corresponding IP address (of the LDAP server) to the XIV system administrator. In our scenario, those are respectively, `xivhost1.xivhost1ldap.storage.tucson.ibm.com` and `9.11.207.232` for Active Directory, and `xivhost2.storage.tucson.ibm.com` and `9.11.207.233`, for SUN Java Directory.

Registering the LDAP server in the XIV system

The **`ldap_add_server`** XCLI command, as shown in Example 5-14 and Example 5-15, is used for adding an LDAP server to the XIV system configuration. The command adds the server but does not activate the LDAP authentication mode. At this stage LDAP authentication should still be disabled at the XIV system.

Example 5-14 Adding LDAP server in XCLI - Active Directory

```
>> ldap_add_server fqdn=xivhost1.xivhost1ldap.storage.tucson.ibm.com
address=9.11.207.232  type="MICROSOFT ACTIVE DIRECTORY"
Command executed successfully.

>> ldap_list_servers
```

```
FQDN                                              Address        Type   Has
Certificate  Expiration Date
xivhost1.xivhost1ldap.storage.tucson.ibm.com   9.11.207.232   Microsoft Active
Directory   no
```

Example 5-15 Adding LDAP server in XCLI - SUN Java Directory

```
>> ldap_add_server fqdn=xivhost2.storage.tucson.ibm.com  address=9.11.207.233
type="SUN DIRECTORY"
Command executed successfully.

>> ldap_list_servers
FQDN                               Address        Type                Has
Certificate  Expiration Date
xivhost2.storage.tucson.ibm.com   9.11.207.233 Sun Directory         no
```

Important: As best practice, the LDAP server and XIV system should have their clocks synchronized to the same time source, be registered and configured to use the same Domain Name Server servers.

The next step for the XIV administrator is to verify Domain Name Server (DNS) name resolution as illustrated in Example 5-16.

Example 5-16 DNS name resolution verification

```
>> dns_test name=xivhost1.xivhost1ldap.storage.tucson.ibm.com
Name                                          IP (Primary DNS)   IP (Secondary DNS)
xivhost1.xivhost1ldap.storage.tucson.ibm.com   9.11.207.232       9.11.207.232
```

If the **dns_test** command returns an unexpected result, do not proceed further with the configuration steps until the DNS name resolution issue is resolved.

LDAP role mapping

For a detailed description of the LDAP role mapping, see *5.3.5, "LDAP role mapping" on page 143*. The XIV configuration parameters `storage_admin_role`, `read_only_role` and `xiv_group_attrib` must have values assigned for LDAP role mapping to work. See Example 5-17.

Example 5-17 Configuring LDAP role mapping

```
>> ldap_config_set storage_admin_role="Storage Administrator"
Command executed successfully.

>> ldap_config_set read_only_role="Read Only"
Command executed successfully.

>> ldap_config_set xiv_group_attrib=description
Command executed successfully.

>> ldap_config_get
Name                     Value
base_dn
xiv_group_attrib         description
third_expiration_event   7
version                  3
```

```
user_id_attrib           objectSiD
current_server
use_ssl                  no
session_cache_period
second_expiration_event  14
read_only_role           Read Only
storage_admin_role       Storage Administrator
first_expiration_event   30
bind_time_limit          0
```

After the configuration is done in XIV, the only two attribute values that can be used in LDAP are “Read Only” and “Storage Administrator”. With the three configuration parameters xiv_group_attrib, read_only_role and storage_admin_role being defined in XIV, the LDAP administrator has sufficient information to create LDAP accounts and populate LDAP attributes with corresponding values.

Creating LDAP accounts in the LDAP Directory

At this stage, user accounts should be created in the LDAP Directory. Refer to Appendix A, “Additional LDAP information” on page 355.

Now that all configuration and verification steps are completed, the LDAP mode can be activated on the XIV system, as illustrated in Example 5-18.

Note: LDAP mode activation “ldap_mode_set mode=active” is an interactive XCLI command. It cannot be invoked using batch mode because it is expecting a Y/N user response. You should use a lower case “y” to respond to the Y/N question, because XCLI will accept the “Shift” key stroke as a response that will not be interpreted as “Y”.

Example 5-18 LDAP mode activation using interactive XCLI session

```
>>ldap_mode_set mode=active

Warning: ARE_YOU_SURE_YOU_WANT_TO_ENABLE_LDAP_AUTHENTICATION Y/N:
Command executed successfully.
>> ldap_mode_get
Mode
Active
>>
```

The LDAP authentication mode is now fully configured, activated, and ready to be tested. A simple test that can validate the authentication result would be to open an XCLI session using credentials of a newly created Active Directory account xivtestuser1 and run **”**ldap_user_test**”.** This command can only be successfully executed by a user authenticated through LDAP (see Example 5-19).

Example 5-19 LDAP authentication validation

```
$ xcli -c "ARCXIVJEMT1" -u xivtestuser1 -p pass2remember ldap_user_test
Command executed successfully.
```

As shown by the command output, xivtestuser1 has been successfully authenticated by the LDAP server and granted access to XIV system. The last step of the verification process is to validate that the current_server configuration parameter is populated with the correct value. This is demonstrated in Example 5-20.

Example 5-20 Validating current_server LDAP configuration parameter

```
>> ldap_config_get
Name                          Value
base_dn CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com
xiv_group_attrib              description
third_expiration_event        7
version                       3
user_id_attrib                objectSiD
current_server                xivhost1.xivhost1ldap.storage.tucson.ibm.com
use_ssl                       no
session_cache_period          10
second_expiration_event       14
read_only_role                Read Only
storage_admin_role            Storage Administrator
first_expiration_event        30
bind_time_limit               30
```

The `current_server` configuration parameter has been populated with the correct value of the LDAP server fully qualified domain name.

5.3.7 LDAP managed user authentication

As previously explained, when the XIV system is configured for LDAP authentication, user credentials are stored in the centralized LDAP repository. The LDAP repository resides on an LDAP server and is accessed by the XIV system, using an LDAP protocol. The LDAP repository maintains the following types of credential objects:

- LDAP user name
- LDAP user password
- LDAP user role
- User groups

LDAP user name

The XIV system limitations for acceptable user names, such as number of characters and character set, no longer apply when user names are stored in an LDAP repository. Each LDAP product has its own set of rules and limitations that applies to user names. Generally, we do not recommend using very long names and non-alpha-numeric characters even if your LDAP product of choice supports it. If at some point you decide to migrate user credentials between local and LDAP repositories or vice-versa, the task can be greatly simplified if the same set of rules is applied to both local and centralized repositories. In fact, the set of rules enforced by the XIV system for local user names should be used for LDAP as well, because it is the most restrictive of the two. For details about XIV system limitations for user names, refer to “User name” on page 122.

Special consideration should be given to using the “space” character in user names. Although this feature is supported with LDAP, it has a potential for making certain administrative tasks more difficult because the user names in this case will have to be enclosed in quotes to be interpreted correctly.

The same set of locally stored predefined user names exist on the XIV system regardless of the authentication mode. Users *technician, admin,* and *smis_user* are always authenticated locally even on a system with activated LDAP authentication mode. Creating LDAP *user accounts* with the same names should be avoided.

If a user account with the same user name is registered in both local and LDAP repositories, and LDAP authentication mode is in effect, then LDAP authentication takes precedence, and the XIV system will perform authentication using LDAP account credentials. The only exception to this rule are the predefined user names listed in the previous paragraph. To reduce complexity and simplify maintenance, it is generally not recommended to have the same user names registered in local and LDAP repositories.

If a user account was registered in the local repository on the XIV system before the LDAP authentication mode was activated, then this account will not be accessible while LDAP authentication is in effect. The account will become accessible again upon deactivation of the LDAP authentication mode.

LDAP user passwords

User passwords are stored in the LDAP repository when the XIV system is in LDAP authentication mode. Password management becomes a function of the LDAP server. The XIV system relies entirely on the LDAP server to provide functionality such as enforcing initial password resets, password strength, password expiration, and so on. Different LDAP server products provide different sets of tools and policies for password management.

The examples in Figure 5-28 and Example 5-21 provide an illustration of some techniques that can be used for password management, and by no means represent a complete list of product password management capabilities.

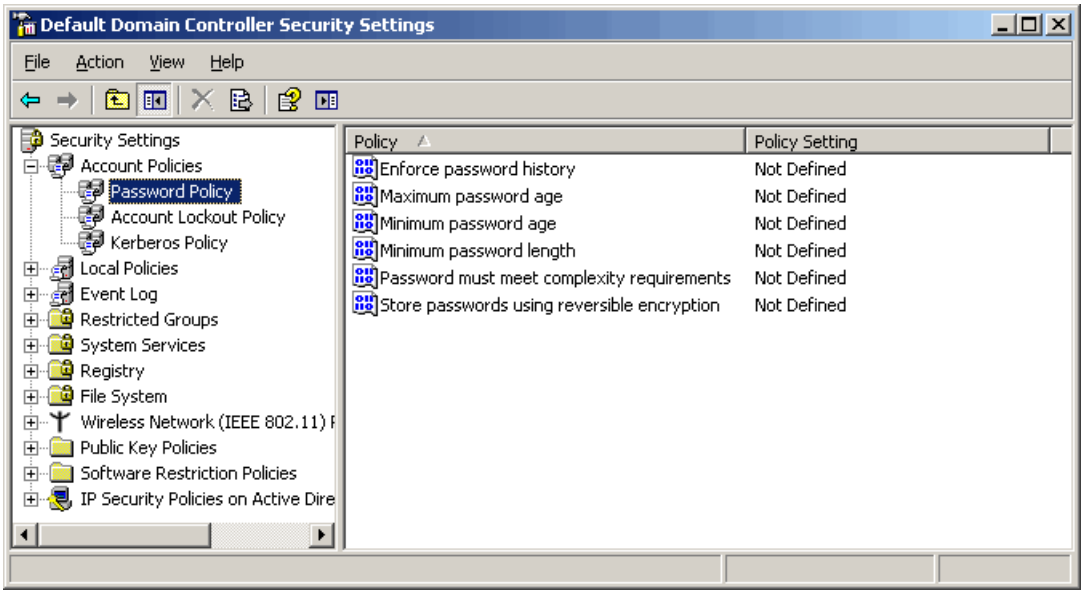


Figure 5-28 Default Active Directory Password Policy settings

Example 5-21 Default SUN Java Directory Password Policy settings

```
# /opt/sun/ds6/bin/dsconf get-server-prop | grep ^pwd
Enter "cn=Directory Manager" password:
pwd-accept-hashed-pwd-enabled      :  N/A
pwd-check-enabled                  :  off
pwd-compat-mode                    :  DS5-compatible-mode
pwd-expire-no-warning-enabled      :  on
pwd-expire-warning-delay           :  1d
pwd-failure-count-interval         :  10m
pwd-grace-login-limit              :  disabled
pwd-keep-last-auth-time-enabled    :  off
```

```
pwd-lockout-duration               :  1h
pwd-lockout-enabled                :  on
pwd-lockout-repl-priority-enabled  :  on
pwd-max-age                        :  disabled
pwd-max-failure-count              :  3
pwd-max-history-count              :  disabled
pwd-min-age                        :  disabled
pwd-min-length                     :  6
pwd-mod-gen-length                 :  6
pwd-must-change-enabled            :  off
pwd-root-dn-bypass-enabled         :  off
pwd-safe-modify-enabled            :  off
pwd-storage-scheme                 :  SSHA
pwd-strong-check-dictionary-path   :  /opt/sun/ds6/plugins/words-english-big.txt
pwd-strong-check-enabled           :  off
pwd-strong-check-require-charset   :  lower
pwd-strong-check-require-charset   :  upper
pwd-strong-check-require-charset   :  digit
pwd-strong-check-require-charset   :  special
pwd-supported-storage-scheme       :  CRYPT
pwd-supported-storage-scheme       :  SHA
pwd-supported-storage-scheme       :  SSHA
pwd-supported-storage-scheme       :  NS-MTA-MD5
pwd-supported-storage-scheme       :  CLEAR
pwd-user-change-enabled            :  on
```

In the event of a user's password expiration or account lockout, the user will get the message shown in Example 5-22, while attempting to login to XCLI.

Example 5-22 XCLI authentication error due to account lockout

```
>> ldap_user_test
Error:   USER_NOT_AUTHENTICATED_BY_LDAP_SERVER
Details: User xivtestuser2 was not authenticated by LDAP server
'xivhost2.storage.tucson.ibm.com'.
```

The XIV GUI in this situation will also fail with the error message seen in Figure 5-29.

Figure 5-29 XIV GUI authentication failure due to account lockout

Although password policy implementation will greatly enhance overall security of the system, all advantages and disadvantages of such implementation should be carefully considered. One of the possible disadvantages is increased management overhead for account management as a result of implementing complex password management policies.

Note: Recommending a comprehensive solution for user password policy implementation is beyond the scope of this book.

LDAP user roles

There are predefined user roles (also referred to as *categories*) used for day to day operation of the XIV Storage System. In the following section we describe predefined roles, their level of access, and applicable use:

- *storageadmin*

  The *storageadmin* (Storage Administrator) role is the user role with the highest level of access available on the system. A user assigned to this role has an ability to perform changes on any system resource except for maintenance of physical components or changing the status of physical components. The assignment of the *storageadmin* role to an *LDAP user* is done through *LDAP role mapping process*. For a detailed description, see *5.3.5, "LDAP role mapping" on page 143*.

- *applicationadmin*

  The *applicationadmin* (Application Administrator) role is designed to provide flexible access control over volume snapshots. Users assigned to the *applicationadmin* role can create snapshots of specifically assigned volumes, perform mapping of their own snapshots to a specifically assigned host, and delete their own snapshots. The user group to which an application administrator belongs determines the set of volumes that the application administrator is allowed to manage. If a user group is defined with *access_all="yes"*, application administrators who are members of that group can manage all volumes on the system. The assignment of the *applicationadmin* role to an LDAP user is done through the *LDAP role mapping process*. For a detailed description, see *5.3.5, "LDAP role mapping" on page 143*. Detailed description of user group to host association is provided in "User group membership for LDAP users".

- *readonly*

  As the name implies, users assigned to *readonly* role can only view system information. Typical use for *readonly* role is a user responsible for monitoring system status, system reporting and message logging and must not be permitted to make any changes on the system. The assignment of *readonly* role to an *LDAP user* is done through *LDAP role mapping process*. For a detailed description, see *5.3.5, "LDAP role mapping" on page 143*.

Note: There is no capability to add new user roles or to modify predefined roles. In LDAP authentication mode, role assignment can be changed by modifying the LDAP attribute (`description` in our example).

LDAP authentication mode implements user role mechanism as a form of *Role Based Access Control* (RBAC). Each predefined user role determines the level of system access and associated functions a user is allowed to use.

Note: The XIV Storage System implements Role Based Access Control (RBAC) based authentication and authorization mechanisms.

All user accounts must be assigned to a single user role. Any LDAP user assigned to multiple roles will not be authenticated by the XIV system. Deleting role assignment (by removing the description attribute value in the LDAP object) of LDAP users will also lead to XIV's inability to authenticate that user.

User group membership for LDAP users

A user group is a group of application administrators who share the same set of snapshot management permissions. The permissions are enforced by associating the user groups with hosts or clusters. User groups are defined locally on the XIV system.

User group membership for an LDAP user is established during the login process by matching the designated LDAP attribute value with the ldap_role parameter assigned to a user group. A user group is associated with host volumes through access definition. An LDAP user, member of the user group, is permitted to manage snapshots of volumes mapped to the host associated with the user group.

User groups have the following characteristics in LDAP authentication mode:

- Only users assigned to the *applicationadmin* role can be members of a user group.
- An LDAP user can only be a member of a single user group.
- A maximum of eight user groups can be created.
- In LDAP authentication mode, there is no limit on the number of members in a user group.
- If a user group is defined with *access_all="yes"*, users assigned to the *applicationadmin* role who are members of that group can manage all snapshots on the system.
- The user group parameter *ldap_role* can only be assigned a single value.
- The *ldap_role* parameter must be unique across all defined user groups.
- Only users assigned to the *storageadmin* role can create, modify and delete user groups.
- Only users assigned to the *storageadmin* role can modify *ldap_role parameter* of a user group.

Important: A user group membership can only be defined for users assigned to the *applicationadmin* role.

Figure 5-30 illustrates the relationship between LDAP user, LDAP role, XIV role, user group membership, associated host, mapped volumes, and attached snapshots.

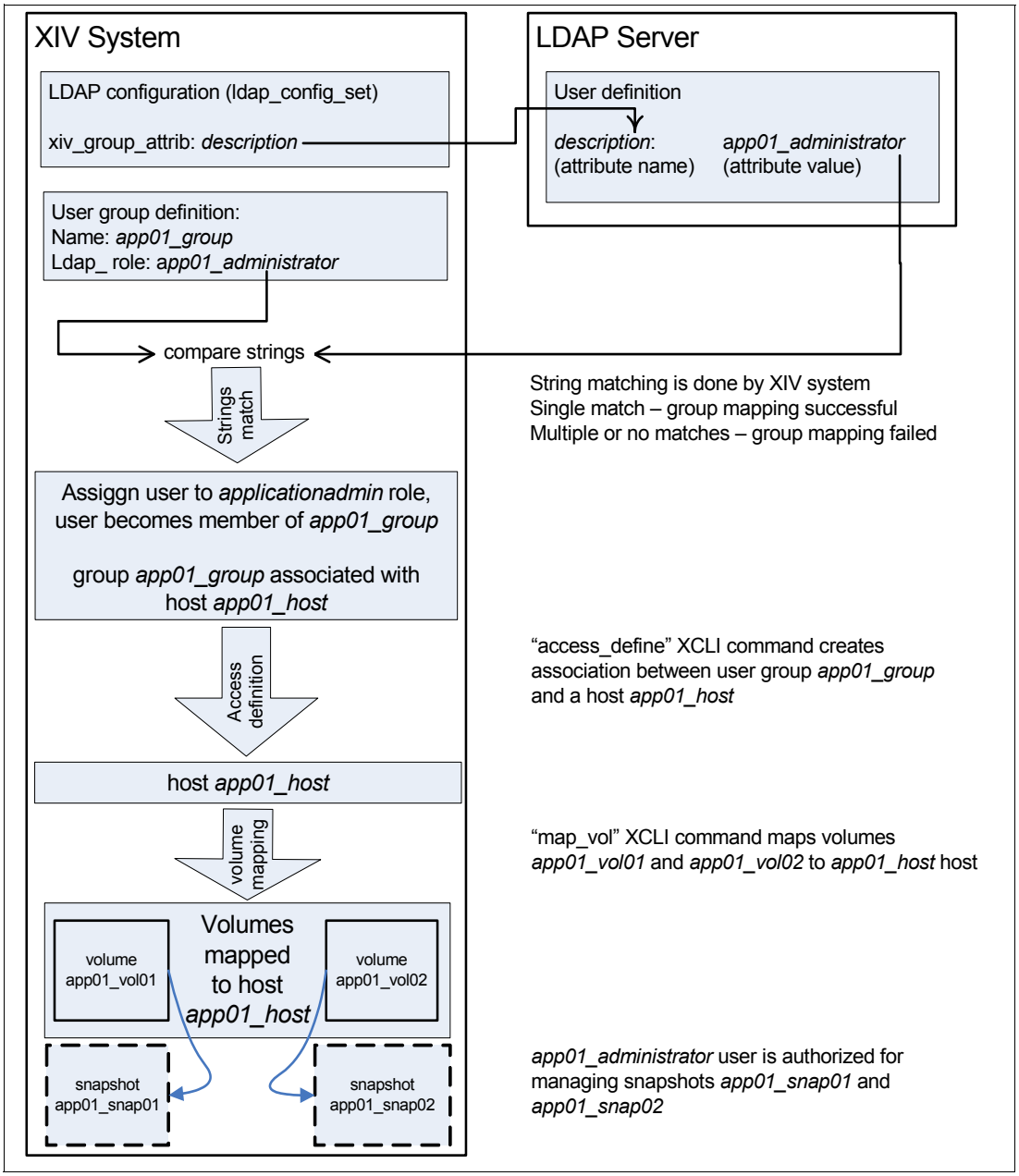


Figure 5-30 User group membership for LDAP user

Example 5-23 is a Korn shell script that can be used to demonstrate the relationship between LDAP user, LDAP role, XIV role, user group membership, associated host, mapped volumes and attached snapshots.

Example 5-23 query_snapshots.ksh listing user's role, group membership, volumes and snapshots

```
#!/bin/ksh

# XIV customer-configurable  LDAP server paramaters:

      LDAPHOSTNAME=xivhost2.storage.tucson.ibm.com
  XIV_GROUP_ATTRIB=description
    READ_ONLY_ROLE="Read Only"
```

```
STORAGE_ADMIN_ROLE="Storage Administrator"
           BASE_DN="dc=xivauth"

echo "Enter username: "
read USERNAME

echo "Enter password: "

stty -echo
read USERPASSWORD
stty echo

LDAP_ROLE=$(/opt/sun/dsee6/bin/ldapsearch -x -b $BASE_DN -D uid=$USERNAME, \
$BASE_DN -w $USERPASSWORD  -h $LDAPHOSTNAME uid=$USERNAME  $XIV_GROUP_ATTRIB \
| grep ^$XIV_GROUP_ATTRIB | awk -F': ' '{ print $2 }')

if [[ $? -ne 0 ]]
then
        echo Failed to query LDAP account details
        exit -1
fi
         xcli -c ARCXIVJEMT1 -u $USERNAME -p $USERPASSWORD ldap_user_test
if [[ $? -ne 0 ]]
then
        exit -1
fi

echo "----------------------------------------------------------------"
if [[   $LDAP_ROLE == $READ_ONLY_ROLE ]]
        then
        echo XIV Role: \"readonly\"
        exit 0
elif [[ $LDAP_ROLE == $STORAGE_ADMIN_ROLE ]]
        then
        echo XIV Role: \"storageadmin\"
        exit 0
else
        echo User:      $USERNAME
        echo LDAP role: $LDAP_ROLE
        echo XIV Role:  \"applicationadmin\"
        USER_GROUP_NAME=`xcli -c ARCXIVJEMT1 -u $USERNAME -p $USERPASSWORD
user_group_list | grep -w $LDAP_ROLE | awk '{ print $1 }'`
echo Member of user group: \"$USER_GROUP_NAME\"

        for HOSTNAME in `xcli -c ARCXIVJEMT1 -u $USERNAME -p $USERPASSWORD
access_list user_group=$USER_GROUP_NAME | grep -v "Type   Name         User Group"
|  awk '{ print $2 }'`
        do
        echo Host: $HOSTNAME associated with $USER_GROUP_NAME user group
        done

        for HOSTNAME in `xcli -c ARCXIVJEMT1 -u $USERNAME -p $USERPASSWORD
access_list user_group=$USER_GROUP_NAME | grep -v "Type   Name" |  awk '{ print $2
}'`
        do
```

```
                for VOLUME  in `xcli -c ARCXIVJEMT1 -u $USERNAME -p $USERPASSWORD
mapping_list  host=app01_host | grep -v LUN | awk '{ print $2 }'`
                do
                SNAPSHOT=`xcli -c ARCXIVJEMT1 -u $USERNAME -p $USERPASSWORD
snapshot_list vol=$VOLUME | grep -v ^Name |  awk '{ print $1 }'`
                echo VOLUME: $VOLUME '->'  SNAPSHOT: $SNAPSHOT
                done
        done

fi
```

The sample output of the *query_snapshots.ksh* script shown in Example 5-24 provides an illustration of the configuration described in Figure 5-30 on page 156.

Example 5-24 Sample run output of query_snapshots.ksh script

```
$ query_snapshots.ksh
Enter username:
app01_administrator
Enter password:
-----------------------------------------------------
User: xivtestuser3
LDAP_ROLE: app01_administrator
XIV Role: "applicationadmin"
Member of user group: "app01_group"
Host: app01_host associated with app01_group user group
VOLUME: app01_vol01 -> SNAPSHOT: app01_snap01
VOLUME: app01_vol02 -> SNAPSHOT: app01_snap02
```

5.3.8 Managing LDAP user accounts

Managing user accounts in LDAP authentication mode is done using LDAP management tools. The XCLI commands and XIV GUI tools cannot be used for creating, deleting, modifying, or listing LDAP user accounts. The set of tools for LDAP account management is specific to the LDAP server type. Examples of LDAP account creation are provided in Appendix A in the topics, “Creating user accounts in Microsoft Active Directory” on page 356 and “Creating user accounts in SUN Java Directory” on page 361. The same set of LDAP management tools can also be used for account removal, modification, and listing.

To generate a list of all LDAP user accounts registered under the Base_DN (XIV system configuration parameter specifying the location of LDAP accounts in the directory information tree DIT), you can use the `ldapsearch` queries shown in Example 5-25 and Example 5-26.

Example 5-25 Generating list of LDAP accounts registered in SUN Java Directory

```
# ldapsearch  -x -b dc=xivauth -H ldap://xivhost2.storage.tucson.ibm.com:389 -D
cn="Directory Manager" -w passw0rd  uid | grep ^uid:

uid: xivsunproduser3
uid: xivsunproduser4
uid: xivsunproduser5
```

Example 5-26 Generating list of LDAP accounts registered in Active Directory

```
# ldapsearch -x  -H "ldap://xivhost1.xivhost1ldap.storage.tucson.ibm.com:389"  -D
'CN=Administrator,CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com' -w
Passw0rd -b 'CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com'
cn=xivtestuser1 | grep ^cn:

cn: xivadproduser10
cn: xivadproduser11
cn: xivadproduser12
```

The queries generating LDAP accounts lists are provided as a demonstration of LDAP tools capabilities to perform a search of information stored in LDAP directory and generate simple reports. Note that both queries are issued on behalf of the LDAP administrator account, `cn="Directory Manager"` and `cn="Administrator"` for SUN Java Directory and Active Directory respectively. A privileged account such as LDAP administrator has the authority level allowing it not only listing but also creating, modifying, and removing other user accounts.

The Active Directory management interface also allows you to build custom views based on LDAP search queries. The following example builds a query that generates the list of XIV accounts whose names start with `xiv`, and whose description is one of the following three: “Storage Administrator“, “`Read Only`“, or starts with “`app`”:

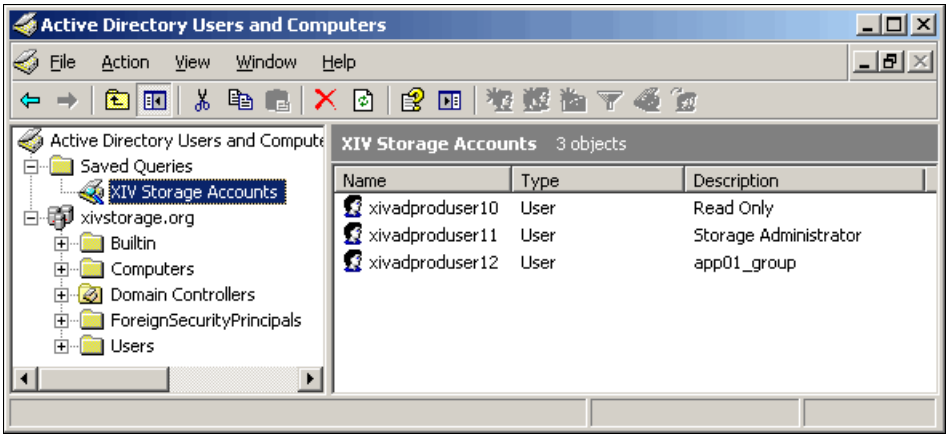


Figure 5-31 Active Directory query listing XIV accounts

For generating “XIV Storage Accounts” view, we used the LDAP query shown in Example 5-27.

Example 5-27 LDAP query for generating list of XIV user accounts

```
(&(&(objectCategory=user)(cn=xiv*)(|(description=Read Only)(description=Storage
Administrator)(description=app*))))
```

To create this query, select **Saved Queries** → **New** → **Query** → XIV Storage Accounts (query name in this example) → **Define Query** → Select **Custom Search** in the **Find** scroll down list → **Advanced** and paste the LDAP query from Example 5-27 into the *Enter LDAP Query* field.

When a new user account is created and its name and attributes satisfy the search criterion, this user account will automatically appear in the *XIV Storage Accounts* view. A similar technique can be applied for managing user accounts in the SUN Java Directory. Any LDAP GUI front-end supporting LDAP version 3 protocol can be used for creating views and managing LDAP entries (XIV user accounts). An example of such an LDAP front end for the SUN Java Directory is the *Directory Editor* product. This product is part of the SUN Java Directory product suit. Demonstrating Directory Editor capabilities for managing LDAP objects is outside of the scope for this book.

Table 5-5 provides a list of commands that *cannot* be used for user account management when LDAP authentication mode is active.

Table 5-5 XIV commands unavailable in LDAP authentication mode

XIV command
user_define
user_update
user_rename
user_group_add_user
user_group_remove_user

Note: When the XIV system operates in LDAP authentication mode, user account creation, listing, modification, and removal functionality is provided by the LDAP Server.

It should be noted that the **user_list** command can still operate when LDAP authentication mode is active. However, this command will only show locally defined XIV user accounts and not LDAP accounts. See Example 5-28.

Example 5-28 user_list command output in LDAP authentication mode

```
>> user_list show_users=all
Name                Category              Group    Active
xiv_development     xiv_development                yes
xiv_maintenance     xiv_maintenance                yes
admin               storageadmin                   yes
technician          technician                     yes
smis_user           readonly                       yes
ITSO                storageadmin                   no
>> ldap_mode_get
Mode
Active
>>
```

As shown in Example 5-28, the `Active` parameter is set to **"**`no`**"** for user `ITSO`. The parameter specifies whether a user can login in current authentication mode. All predefined local XIV users can still login when LDAP authentication mode is active.

Defining user groups with the GUI in LDAP authentication mode

User group information is stored locally on the XIV system regardless of the authentication mode. The user group concept only applies to users assigned to an application_administrator role.

A user group can also be associated with one or multiple hosts or clusters.

The following steps illustrate how to create user groups, how to add users (with application administrator role) to the group, and how to define host associations for the group:

1. Be sure to log in as `admin` (or another user with storage administrator rights). From the Access menu, click **Users Groups** as shown in Figure 5-32. In our scenario, we create a user group called `app01_group`. The user groups can be selected from the Access menu (padlock icon).

Figure 5-32 Select Users Groups

2. The Users Groups window displays. To add a new user group, either click the Add User Group icon (shown in Figure 5-33) in the menu bar, or right-click in an empty area of the User Groups table and select **Add User Group** from the context menu.

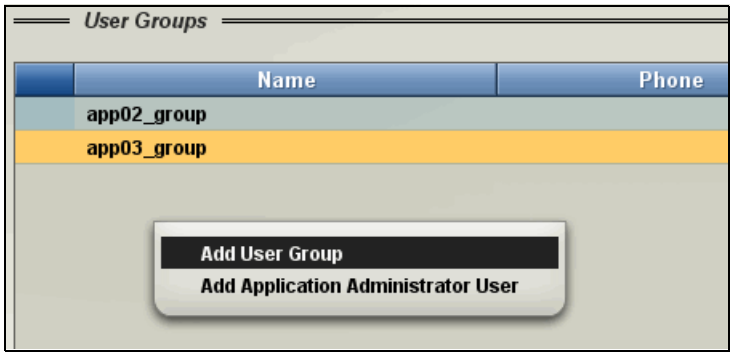


Figure 5-33 Add User Group

3. The Create User Group dialog displays. Enter a meaningful group name, specify role for LDAP role mapping described in *5.3.5, "LDAP role mapping" on page 143* and click **Add** (refer to Figure 5-34). To avoid potential conflicts with already registered user groups, the XIV system verifies the uniqueness of the group name and the role. If a user group with the same name or the same role exists in the XIV repository, the attempt to create a new user group will fail and an error message is displayed.

   The *Full Access* flag has the same significance as in native authentication mode. If a user group has the *Full Access* flag turned on, all members of that group will have unrestricted access to all snapshots on the system.

Figure 5-34 Enter New User Group Name and Role for LDAP role mapping

4. At this stage, the user group `app01_group` is still empty. Next, we add a host to the user group. Select **Access Control** from the Access menu as shown in Figure 5-35. The Access Control window appears.

Figure 5-35 Access Control

5. Right-click the name of the user group that you have created to bring up a context menu and select **Update Access Control** as shown in Figure 5-36.

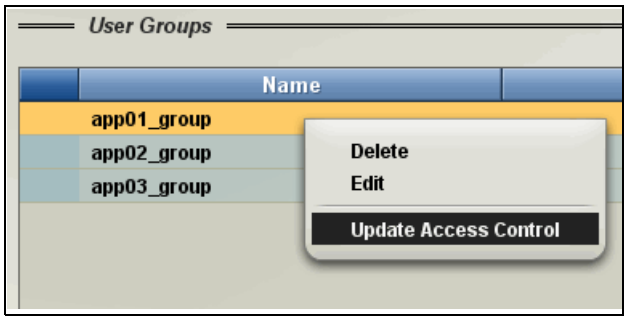


Figure 5-36 Updating Access Control for a user group

6. The User Group Access Control dialog shown in Figure 5-37 is displayed. The panel contains the names of all the hosts and clusters defined on the XIV Storage System. The left pane displays the list of Unauthorized Hosts/Clusters for this particular user group and the right pane shows the list of hosts that have already been associated with the user group. You can add or remove hosts from either list by selecting a host and clicking the appropriate arrow. Finally, click **Update** to save the changes.

Figure 5-37 Access Control Definitions panel

7. Unlike in native authentication mode, in LDAP authentication mode, user group membership cannot be defined using the XIV GUI or XCLI. The group membership is determined at the time the LDAP authenticated user logs into the XIV system, based on the information stored in LDAP directory. A detailed description of the process of determining user group membership can be found in *5.3.5, "LDAP role mapping" on page 143*.

After a user group is defined, it cannot be removed as long as the XIV system operates in LDAP authentication mode.

5.3.9 Managing user groups using XCLI in LDAP authentication mode

This section summarizes the commands and options available to manage user groups, roles, and associated host resources through the XCLI.

Defining user groups with the XCLI

To use the GUI to define user groups:

1. Use the **`user_group_create`** command as shown in Example 5-29 to create a user group called `app01_group` with corresponding LDAP role `app01_administrator`.

Example 5-29 XCLI user_group_create in LDAP authentication mode

```
>> user_group_create user_group=app01_group ldap_role=app01_administrator
Command completed successfully
```

Note: Avoid spaces in user group names. If spaces are required, the group name must be placed between single quotation marks, such as '`name with spaces`'.

2. The user group `app01_group` is empty and has no associated hosts or clusters. The next step is to associate a host or cluster with the group. In Example 5-30, user group `app01_group` is associated to `app01_host`.

Example 5-30 XCLI access_define

```
>> access_define user_group=app01_group host=app01_host
Command completed successfully
```

5.3.10 Active Directory group membership and XIV role mapping

In all previous examples in this chapter, the XIV group membership was defined based on the value of the description attribute of a corresponding LDAP object (LDAP user account). When a user logs in to the system, the value of that description attribute is compared with the value of the XIV configuration parameters `read_only_role` and `storage_admin_role`, or with the `ldap_role` parameter of the defined user groups (for details, refer to *5.3.5, "LDAP role mapping" on page 143*).

This approach works consistently for both LDAP server products, Active Directory and SUN Java Directory. However, it has certain limitations. If the `description` attribute is used for role mapping, it can no longer be used for anything else. For instance, you will not be able to use it for entering the actual description of the object. Another potential limitation is that every time you create a new account in LDAP, you must type text (typically `read_only_role`, `storage_admin_role`, or `ldap_role`) that has to match exactly how the configuration parameters were defined in the XIV system. The process is manual and potentially error prone. Any typing error in the role mapping attribute value will lead to the user's inability to login to the XIV system.

The alternative to using the `description` attribute for role mapping is to use Active Directory group membership. In Active Directory, a user can be a member of a single group or multiple groups. An LDAP group is a collection of users with common characteristics. *Group* is defined in the Active Directory container *Users*. A group is defined first as an empty container, and then existing users can be assigned as members of this group. A group is represented as a separate object in the LDAP Directory Information Tree (DIT) and gets a distinguished name (DN) assigned to it.

Groups defined in the Active Directory can be used for XIV role mapping. When a user becomes a member of a group in the Active Directory, it gets a new attribute assigned. The value of the new attribute points to the DN of the group. `MemberOf` is the name of that attribute. The `MemberOf` attribute value determines the Active Directory group membership.

To create a group in Active Directory:

1. Start Active Directory Users and Computer by selecting **Start** → **Administrative Tools** → **Active Directory Users and Computers**
2. Right-click on "Users" container, select **New** → **Group.**
3. Enter a group name and click **OK.**

The new Active Directory group creation dialog is shown in Figure 5-38.

Figure 5-38 Creating Active Directory group

To assign an existing user to the new group:

1. Start Active Directory Users and Computer by selecting **Start** → **Administrative Tools** → **Active Directory Users and Computers**.
2. Expand the *Users* container, right-click the user name that you want to make a member of the new group, and select **Properties**.
3. In the Properties panel, select the **Member Of** tab and click **Add** → **Advanced** → **Find Now.** From the presented list of existing user groups, select **XIVReadOnly** and click **OK**.
4. You should now see a group selection dialog as shown in Figure 5-39. Confirm your choice by clicking **OK**.

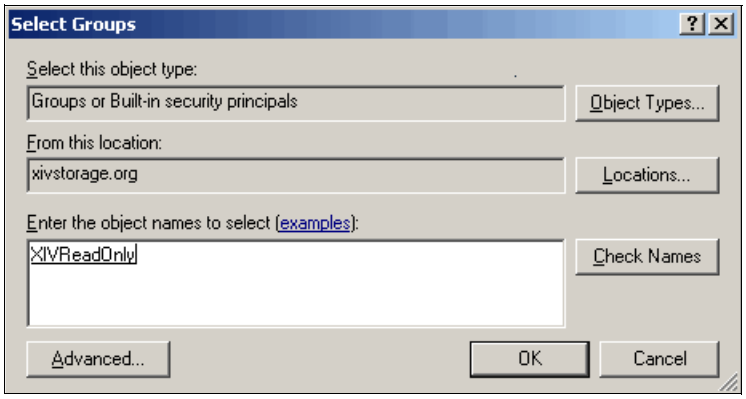


Figure 5-39 Active Directory group selection dialog

To illustrate the new `memberOf` attribute in the existing LDAP user object and the new LDAP object representing the “XIVReadOnly” group, we run the **`ldapsearch`** queries against the Active directory LDAP server as shown in Example 5-31.

Example 5-31 Active Directory group membership ldapsearch queries

```
# ldapsearch -x -H "ldap://xivstorage.org:389" -b CN=Users,dc=xivstorage,dc=org
-D cn=xivadproduser10,CN=Users,dc=xivstorage,dc=org -w pass2remember
cn=xivadproduser10 memberOf
dn: CN=xivadproduser10,CN=Users,DC=xivstorage,DC=org
memberOf: CN=XIVReadOnly,CN=Users,DC=xivstorage,DC=org

# ldapsearch -x -H "ldap://xivstorage.org:389" -b CN=Users,dc=xivstorage,dc=org
-D cn=xivadproduser10,CN=Users,dc=xivstorage,dc=org -w pass2remember
cn=XIVReadOnly

dn: CN=XIVReadOnly,CN=Users,DC=xivstorage,DC=org
objectClass: top
objectClass: group
cn: XIVReadOnly
member: CN=xivadproduser10,CN=Users,DC=xivstorage,DC=org
distinguishedName: CN=XIVReadOnly,CN=Users,DC=xivstorage,DC=org
instanceType: 4
whenCreated: 20090712021348.0Z
whenChanged: 20090712021451.0Z
uSNCreated: 61451
uSNChanged: 61458
name: XIVReadOnly
objectGUID:: Ai3j9w7at02cEjV1lpk/fQ==
objectSid:: AQUAAAAAAAUVAAAA1a5i5i2p5CXmfcb4aQQAAA==
sAMAccountName: ReadOnly
sAMAccountType: 268435456
groupType: -2147483646
objectCategory: CN=Group,CN=Schema,CN=Configuration,DC=xivstorage,DC=org
```

In the first **ldapsearch** query, we intentionally limited our search to the `memberOf` attribute (at the end of the **ldapsearch** command) so that the output is not obscured with unrelated attributes and values. The value of the `memberOf` attribute contains the DN of the group.

The second **ldapsearch** query illustrates the `CN=XIVReadOnly` LDAP object content. Among other attributes, it contains the `member` attribute that points at the DN of the user defined as a member. The attribute `member` is a multivalued attribute; there could be more than one user assigned to the group as a member. `MemberOf` is also a multivalued attribute, and a user can be a member of multiple groups.

The XIV system can now be configured to use the `memberOf` attribute for role mapping. In Example 5-32 we are mapping the Active Directory group `XIVReadOnly` to the XIV `read_only_role`, `XIVStorageadmin` to `storage_admin_role`, and XIV `user group app01_group` to Active Directory group `XIVapp01_group`. You must be logged on as admin.

Example 5-32 Configuring XIV to use Active Directory groups for role mapping

```
>> ldap_config_set xiv_group_attrib=memberOf
Command executed successfully.
>> ldap_config_set read_only_role=CN=XIVReadOnly,CN=Users,DC=xivstorage,DC=org
Command executed successfully.
>> ldap_config_set
storage_admin_role=CN=XIVStorageadmin,CN=Users,DC=xivstorage,DC=org
Command executed successfully.
>> ldap_config_get
```

```
Name                      Value
base_dn                   CN=Users,dc=xivstorage,dc=org
xiv_group_attrib          memberOf
third_expiration_event    7
version                   3
user_id_attrib            objectSid
current_server            xivstorage.org
use_ssl                   no
session_cache_period      10
second_expiration_event   14
read_only_role            CN=XIVReadOnly,CN=Users,DC=xivstorage,DC=org
storage_admin_role        CN=XIVStorageadmin,CN=Users,DC=xivstorage,DC=org
first_expiration_event    30
bind_time_limit           30

>> user_group_update user_group=app01_group
ldap_role=cn=XIVapp01_group,CN=Users,DC=xivstorage,DC=org
Command executed successfully.
>> user_group_list user_group=app01_group
Name          Access All  LDAP Role                                          Users
app01_group   no          cn=XIVapp01_group,CN=Users,DC=xivstorage,DC=org
```

Alternatively, the same configuration steps could be accomplished through the XIV GUI. To change the LDAP configuration settings in the XIV GUI, open the “Tools” menu at the top of the main XIV Storage Manager panel, select **Configure → LDAP → Role Mapping,** and change the configuration parameter settings as shown in Figure 5-40.

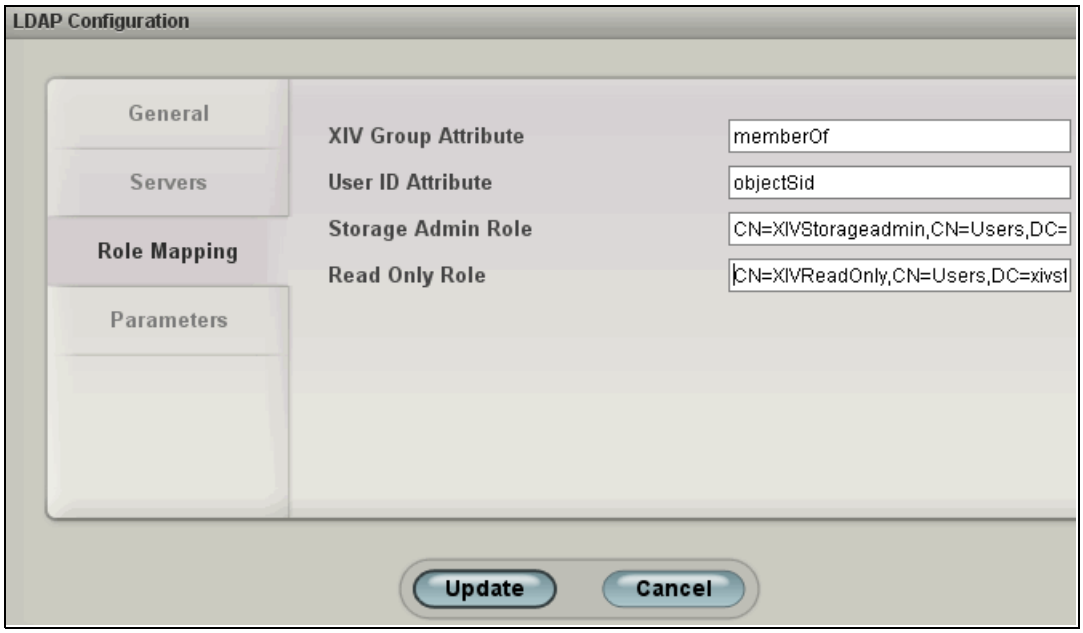


Figure 5-40 Using XIV GUI to configure LDAP role mapping

Important: The XIV configuration parameters “Storage Admin Role” and “Read Only Role” can only accept a string of up to 64 characters long. In some cases the length of the domain name might prevent you from using the memberOf attribute for role mapping because the domain name is encoded in the attribute value (“`DC=xivstorage,DC=org`” in this example represents the `xivstorage.org` domain name).

Now, by assigning Active Directory group membership, you can grant access to the XIV system as shown in Figure 5-39 on page 165.

A user in Active Directory can be a member of multiple groups. If this user is a member of more than one group with corresponding role mapping, XIV fails authentication for this user due to the fact that the role cannot be uniquely identified. In Example 5-33, user `xivadproduser10` can be mapped to `Storage Admin` and `Read Only` roles, hence the authentication failure followed by the `USER_HAS_MORE_THAN_ONE_RECOGNIZED_ROLE` error message.

Example 5-33 LDAP user mapped to multiple roles authentication failure

```
$ xcli -c "ARCXIVJEMT1" -u xivadproduser10 -p pass2remember ldap_user_test
Error:   USER_HAS_MORE_THAN_ONE_RECOGNIZED_ROLE
Details: User xivadproduser10 has more than one recognized LDAP role.

$ ldapsearch -x -H "ldap://xivstorage.org:389" -b CN=Users,dc=xivstorage,dc=org
-D cn=xivadproduser10,CN=Users,dc=xivstorage,dc=org -w pass2remember
cn=xivadproduser10 memberOf

dn: CN=xivadproduser10,CN=Users,DC=xivstorage,DC=org
memberOf: CN=XIVReadOnly,CN=Users,DC=xivstorage,DC=org
memberOf: CN=XIVStorageadmin,CN=Users,DC=xivstorage,DC=org
```

An LDAP user can be a member of multiple Active Directory groups and successfully authenticate to XIV as long as only one of those groups is mapped to an XIV role. As illustrated in Example 5-34, the user `xivadproduser10` is a member of two Active Directory groups `XIVStorageadmin` and `nonXIVgroup`. Only `XIVStorageadmin` is mapped to an XIV role.

Example 5-34 LDAP user mapped to a single roles authentication success

```
$ xcli -c "ARCXIVJEMT1" -u xivadproduser10 -p pass2remember ldap_user_test
Command executed successfully.

$ ldapsearch -x -H "ldap://xivstorage.org:389" -b CN=Users,dc=xivstorage,dc=org
-D cn=xivadproduser10,CN=Users,dc=xivstorage,dc=org -w pass2remember
cn=xivadproduser10 memberOf

dn: CN=xivadproduser10,CN=Users,DC=xivstorage,DC=org
memberOf: CN=nonXIVgroup,CN=Users,DC=xivstorage,DC=org
memberOf: CN=XIVStorageadmin,CN=Users,DC=xivstorage,DC=org
```

After all Active Directory groups are created and mapped to corresponding XIV roles, the complexity of managing LDAP user accounts will be significantly reduced because the role mapping can now be done through Active Directory group membership management. The easy to use point and click interface leaves less room for error when it comes to assigning group membership, as opposed to entering text into `description` field as previously described.

5.3.11 SUN Java Directory group membership and XIV role mapping

SUN Java Directory group membership can be used for XIV role mapping similar to the way described in 5.3.10, “Active Directory group membership and XIV role mapping” on page 164.

In SUN Java Directory, a user can be a member of a single or multiple groups. A group is a collection of users with common characteristics. Groups can defined anywhere in DIT in SUN Java Directory. A group is represented as a separate object in LDAP Directory Information Tree (DIT) and gets a distinguished name (DN) assigned to it.

Groups defined in SUN Java Directory can be used for XIV role mapping. When a user becomes member of a group in SUN Java Directory, he or she gets a new attribute assigned. The value of the new attribute points to the DN of the group. `IsmemberOf` is the name of that attribute. The `IsmemberOf` attribute value determines the SUN Java Directory group membership.

To create a group in SUN Java Directory using SUN Java Web Console tool:

1. Point your Web browser to HTTPS port 6789, in our case, **“**`https://xivhost2.storage.tucson.ibm.com:6789`**”**
2. Login to the system and select the “Directory Service Control Center (DSCC)” application and Authenticate to Directory Service Manager.
3. Select **Directory Servers** tab, click on `xivhost2.storage.tucson.ibm.com:389` and select **Entry Management** tab. Verify that dc=xivauth suffix is highlighted in the left panel and click **New Entry** in the Selected Entry panel.
4. Accept `dc=xivauth` in **Entry Parent DN:** field → **Next** → **Entry Type** “Static Group - (groupOfUniqueNames) → **Next** → and enter attribute values as shown in Figure 5-41.

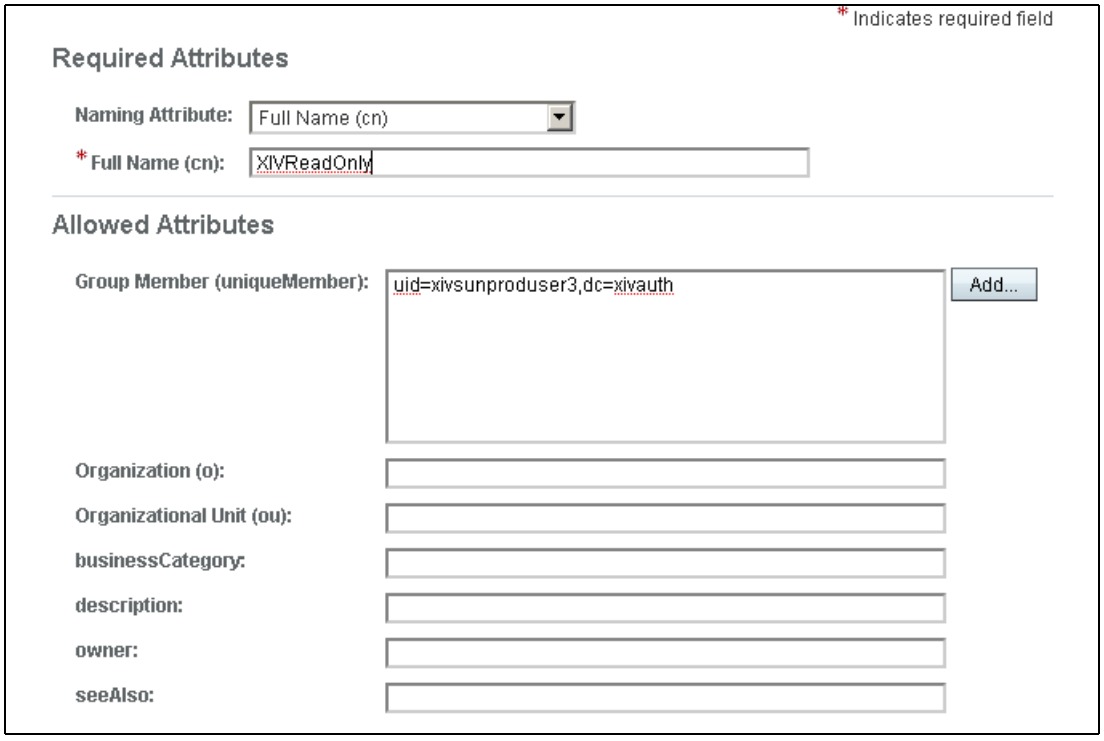


Figure 5-41 Creating new group in SUN Java Directory

Confirm your choice by clicking **Finish** in the panel that is partially shown in Figure 5-42.

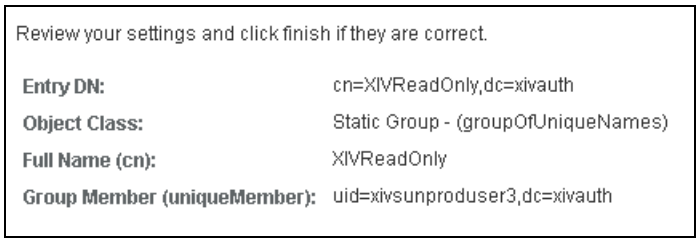


Figure 5-42 Confirming group creation in SUN Java Directory

To illustrate the new IsemberOf attribute in the existing LDAP user object and the new LDAP object representing the "XIVReadOnly" group, we run the **ldapsearch** queries against SUN Java Directory LDAP server as shown in Example 5-35.

Example 5-35 SUN Java Directory group membership ldapsearch queries

```
$ ldapsearch -x -H ldap://xivhost2.storage.tucson.ibm.com:389 -D
uid=xivsunproduser3,dc=xivauth -w passw0rd  -b dc=xivauth uid=xivsunproduser3
ismemberof

dn: uid=xivsunproduser3,dc=xivauth
ismemberof: cn=XIVReadOnly,dc=xivauth


$ ldapsearch -x -H ldap://xivhost2.storage.tucson.ibm.com:389 -D
uid=xivsunproduser3,dc=xivauth -w passw0rd  -b dc=xivauth cn=XIVReadOnly
dn: cn=XIVReadOnly,dc=xivauth
objectClass: groupOfUniqueNames
objectClass: top
uniqueMember: uid=xivsunproduser3,dc=xivauth
cn: XIVReadOnly
```

In the first **ldapsearch** query, we intentionally limited our search to the `ismember of` the attribute (at the end of the **ldapsearch** command) so that the output is not obscured with unrelated attributes and values. The value of the `ismemberof` attribute contains the DN of the group. The second **ldapsearch** query illustrates the `CN=XIVReadOnly` LDAP object content. Among other attributes, it contains the `uniqueMember` attribute, which points at the DN of the user defined as a member. The attribute `uniqueMember` is a multivalued attribute, and there could be more than one user assigned to the group as a member. `Ismemberof` is also a multivalued attribute, and a user can be a member of multiple groups.

XIV can now be configured to use the `ismemberof` attribute for role mapping. In example Example 5-36, we are mapping the SUN Java Directory group `XIVReadOnly` to the XIV `read_only_role`, `XIVStorageadmin` to the `storage_admin_role`, and the XIV `user group app01_group` to the `SUN Java Directory group XIVapp01_group`. You must be logged in to the XCLI as admin.

Example 5-36 Configuring XIV to use SUN Java Directory groups for role mapping

```
$ xcli -c "ARCXIVJEMT1" -u admin -p s8cur8pwd ldap_config_set
xiv_group_attrib=ismemberof
Command executed successfully.
>> ldap_config_set read_only_role=cn=XIVReadOnly,dc=xivauth
Command executed successfully.
```

```
>> ldap_config_set storage_admin_role=cn=XIVStorageAdmin,dc=xivauth
Command executed successfully.

>> ldap_config_get
Name                      Value
base_dn                   dc=xivauth
xiv_group_attrib          ismemberof
third_expiration_event    7
version                   3
user_id_attrib            uid
current_server
use_ssl                   no
session_cache_period      10
second_expiration_event   14
read_only_role            cn=XIVReadOnly,dc=xivauth
storage_admin_role        cn=XIVStorageAdmin,dc=xivauth
first_expiration_event    30
bind_time_limit           30

>> user_group_update user_group=app01_group ldap_role=cn=XIVapp01_group,dc=xivauth
Command executed successfully.
>> user_group_list user_group=app01_group
Name          Access All   LDAP Role                       Users
app01_group   no           cn=XIVapp01_group,dc=xivauth
```

Alternatively, the same configuration steps could be accomplished through the XIV GUI. To change the LDAP configuration settings in XIV GUI open “Tools” menu at the top of main XIV Storage Manager panel, select **Configure** → **LDAP** → **Role Mapping** and change the configuration parameter settings as shown in Figure 5-43.

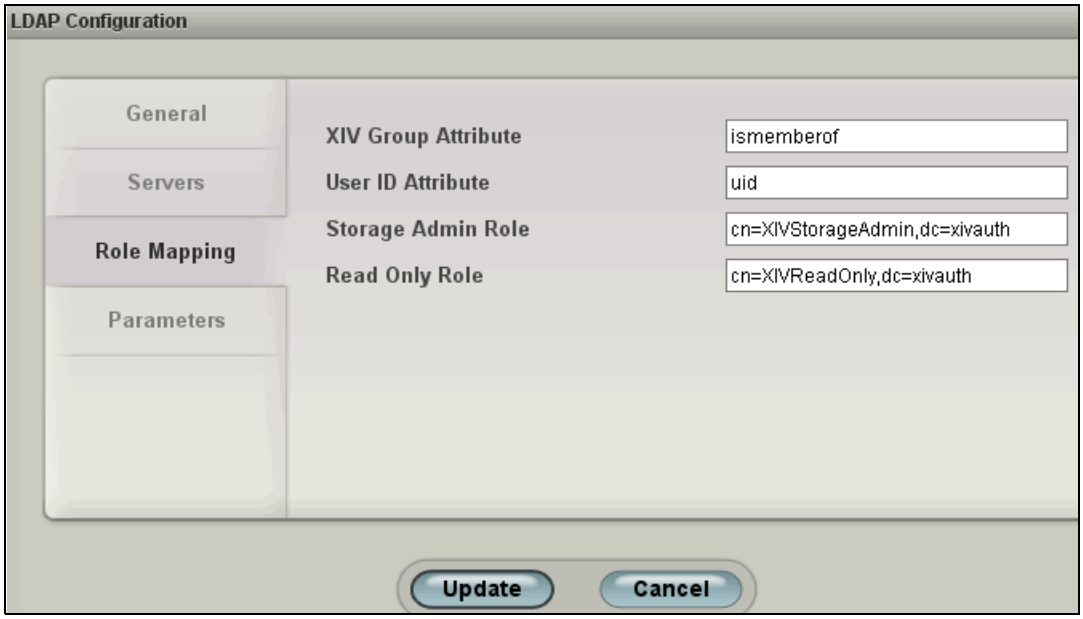


Figure 5-43 Using XIV GUI to configure LDAP role mapping

Important: The XIV configuration parameters “Storage Admin Role” and “Read Only Role” can only accept a string of up to 64 characters long. In some cases, the length of the distinguished name (DN) might prevent you from using the `ismemberof` attribute for role mapping because the DN is encoded in the attribute value (“`dc=xivauth`**”**) in this example.

Now, by assigning SUN Java Directory group membership, you can grant access to the XIV system as shown in Figure 5-41 on page 169.

A user in the SUN Java Directory can be a member of multiple groups. If this user is a member of more than one group with corresponding role mapping, XIV fails authentication for this user because the role cannot be uniquely identified. In Example 5-37, user `xivsunproduser3` can be mapped to the `Storage Admin` and `Read Only` roles, hence the authentication failure followed by the `USER_HAS_MORE_THAN_ONE_RECOGNIZED_ROLE` error message.

Example 5-37 LDAP user mapped to multiple roles authentication failure

```
$ xcli -c "ARCXIVJEMT1" -u xivsunproduser3 -p pass2remember ldap_user_test
Error:   USER_HAS_MORE_THAN_ONE_RECOGNIZED_ROLE
Details: User xivsunproduser3 has more than one recognized LDAP role.

$ ldapsearch -x -H ldap://xivhost2.storage.tucson.ibm.com:389 -D
uid=xivsunproduser3,dc=xivauth -w passw0rd  -b dc=xivauth uid=xivsunproduser3
ismemberof

dn: uid=xivsunproduser3,dc=xivauth
ismemberof: cn=XIVReadOnly,dc=xivauth
ismemberof: cn=XIVStorageAdmin,dc=xivauth
```

An LDAP user can be a member of multiple SUN Java Directory groups and successfully authenticate in XIV as long as only one of those groups is mapped to an XIV role. As illustrated in Example 5-38, the user `xivsunproduser3` is a member of two SUN Java Directory groups, `XIVStorageadmin` and `nonXIVgroup`. Only `XIVStorageadmin` is mapped to an XIV role.

Example 5-38 LDAP user mapped to a single roles authentication success

```
$ xcli -c "ARCXIVJEMT1" -u xivsunproduser3 -p pass2remember ldap_user_test
Command executed successfully.

$ ldapsearch -x -H ldap://xivhost2.storage.tucson.ibm.com:389 -D
uid=xivsunproduser3,dc=xivauth -w passw0rd  -b dc=xivauth uid=xivsunproduser3
ismemberof

dn: uid=xivsunproduser3,dc=xivauth
ismemberof: cn=XIVReadOnly,dc=xivauth
ismemberof: cn=nonXIVgroup,dc=xivauth
```

After all SUN Java Directory groups are created and mapped to corresponding XIV roles, the complexity of managing LDAP user accounts will be significantly reduced because the role mapping can now be done through SUN Java Directory group membership management. The easy to use point and click interface leaves less room for error when it comes to assigning group membership, as opposed to entering text into the `description` field as previously described.

5.3.12 Managing multiple systems in LDAP authentication mode

The task of managing multiple XIV Storage Systems can be significantly simplified by using LDAP authentication mode. Because user credentials are stored centrally in the LDAP directory, it is no longer necessary to synchronize user credentials among multiple XIV systems. After a user account is registered in LDAP, multiple XIV systems can use credentials stored in LDAP directory for authentication. Because the user's password is stored in the LDAP directory, all connected XIV systems will authenticate the user with this password, and if the password is changed, all XIV systems will automatically accept the new password. This mode of operation is often referred to as "Single Sign-On". This mode allows for quick transitions between systems in the XIV GUI because the password has to be entered only once. This approach is especially useful in Remote Mirror configurations, where the storage administrator is required to frequently switch from source to target system.

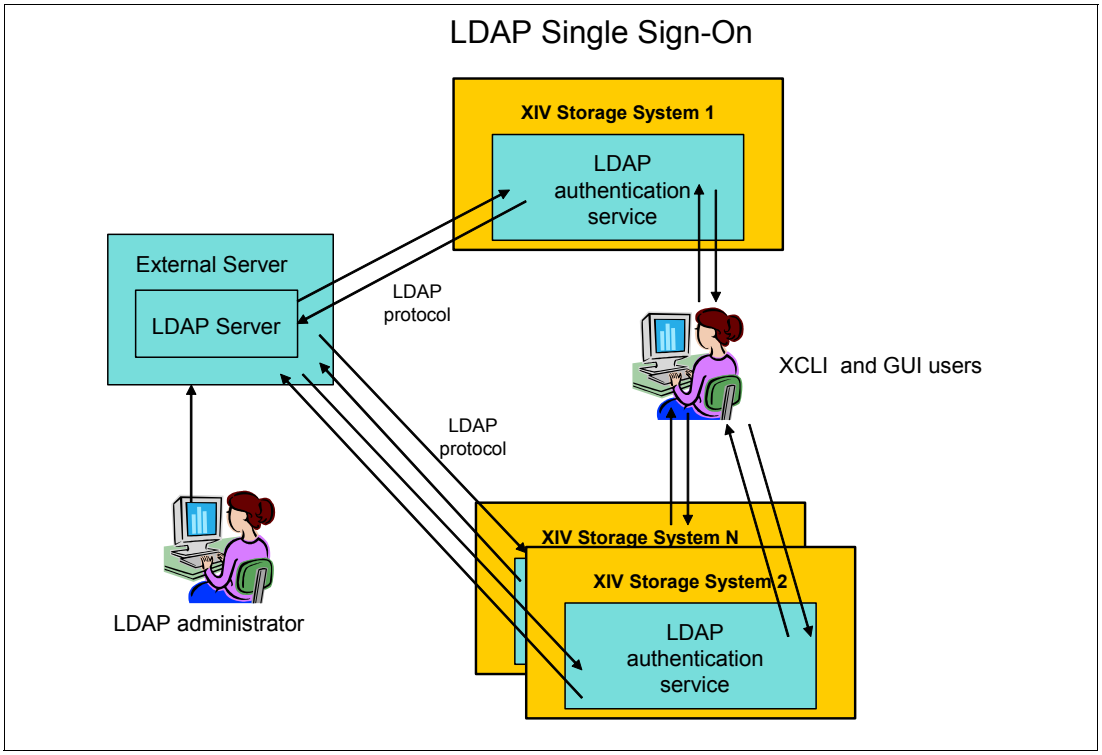


Figure 5-44 LDAP Single Sign-On

Important: To allow single sign-on in LDAP authentication mode all XIV systems should be configured to use the same set of LDAP configuration parameters for role mapping. If role mapping is setup differently on any two XIV systems, it is possible that a user can login to one but not the other XIV system.

5.4 Securing LDAP communication with SSL

In any authentication scenario, information is exchanged between the LDAP server and XIV system where access is being sought. Security Socket Layer (SSL) can used to implement secure communications between the LDAP client and server. LDAPS (LDAP over SSL, the secure version of LDAP protocol) allows secure communications between the XIV system and LDAP server with encrypted SSL connections. This allows a setup where user passwords never appear on the wire in clear text.

SSL provides methods for establishing identity using X.509 certificates and ensuring message privacy and integrity using encryption. In order to create an SSL connection, the LDAP server must have a digital certificate signed by a trusted certificate authority (CA). Companies have the choice of using a trusted third-party CA or creating their own certificate authority. In this scenario, the `xivauth.org` CA will be used for demonstration purposes.

To be operational, SSL has to be configured on both the client and the server. Server configuration includes generating a certificate request, obtaining a server certificate from a certificate authority (CA), and installing the server and CA certificates.

Refer to Appendix A, “Additional LDAP information” on page 355 for guidance on how to configure Windows Server and SUN Java Directory for SSL support. You can then proceed to “Configuring XIV to use LDAP over SSL”.

Configuring XIV to use LDAP over SSL

To be operational, SSL has to be configured on both the XIV system and the LDAP server. Client (XIV system) configuration includes uploading CA certificates to XIV and enabling SSL mode. The `cacert.pem` file is ready to be uploaded to the XIV system.

When a new LDAP server is added to the XIV system configuration, a security certificate can be entered in the optional `certificate` field. If the LDAP server was originally added without a certificate, you will need to remove that definition first and add new definition with the certificate.

Note: When defining the LDAP server with a security certificate in XIV, the fully qualified name of the LDAP server must match the “issued to name” in the client’s certificate.

For registering the LDAP server with security certificate, it might be easier to use the XIV GUI as it has file upload capability (see Figure 5-45). XCLI can also be used, but in this case you need to cut and paste a very long string containing the certificate into the XCLI session. To define the LDAP server in the XIV GUI, open the *Tools* menu at the top of the main XIV Storage Manager panel, and select **Configure** → **LDAP** → **Servers** (green ‘+’ sign on the right panel).

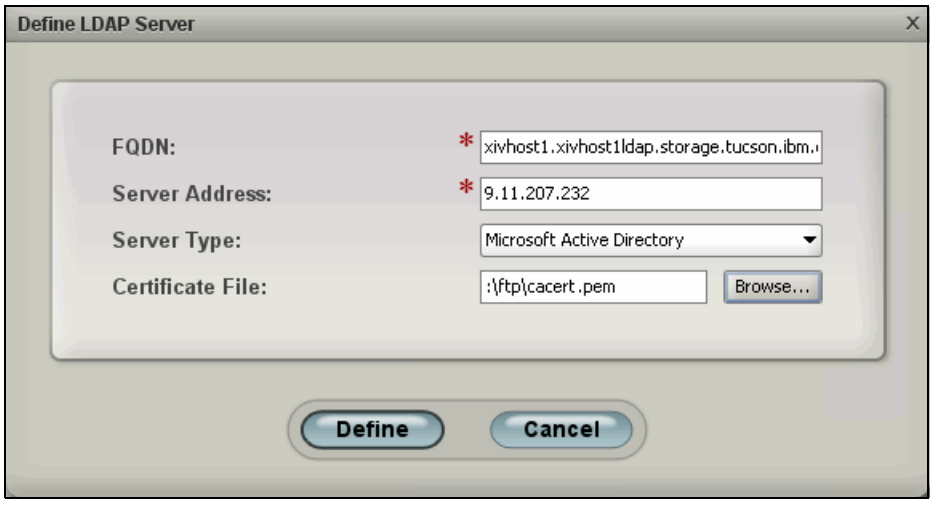


Figure 5-45 Defining Active Directory LDAP server with SSL certificate

In Figure 5-45, the Server Type selected must correspond to your specific LDAP directory, either Microsoft Active Directory as shown, or Sun Directory.

To verify that the LDAP server is defined with the correct certificate, compare the certificate expiration date as it is registered in XIV with the “Valid to” date as shown in Figure A-12 on page 373 (for Microsoft Active Directory) or Figure A-18 on page 379 (for SUN Java Directory).

To view the expiration date of the installed certificate in XIV GUI, open the *Tools* menu at the top of the main XIV Storage Manager panel, select **Configure** → **LDAP** → **Servers**, and double-click on a name of the LDAP server. In our example, the expiration date of the certificate as shown by XIV system in Figure 5-46 matches the “Valid to” date as shown in Figure A-12 on page 373 (for Microsoft Active Directory) or Figure A-18 on page 379 (for SUN Java Directory).

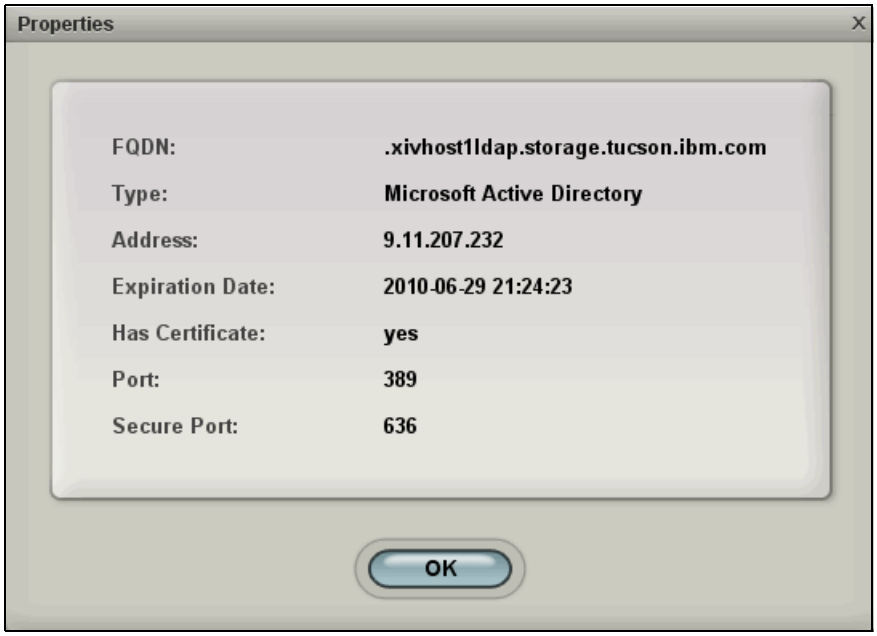


Figure 5-46 Viewing Active Directory server certificate expiration date

By default, LDAP authentication on XIV is configured to use non-SSL communication. To enable the use of SSL in the XIV GUI, open the *Tools* menu at the top of main XIV Storage Manager panel, select **Configure** → **LDAP** → **Parameters** → **Use SS**, and change it from “No” to “Yes” as shown in Figure 5-47.

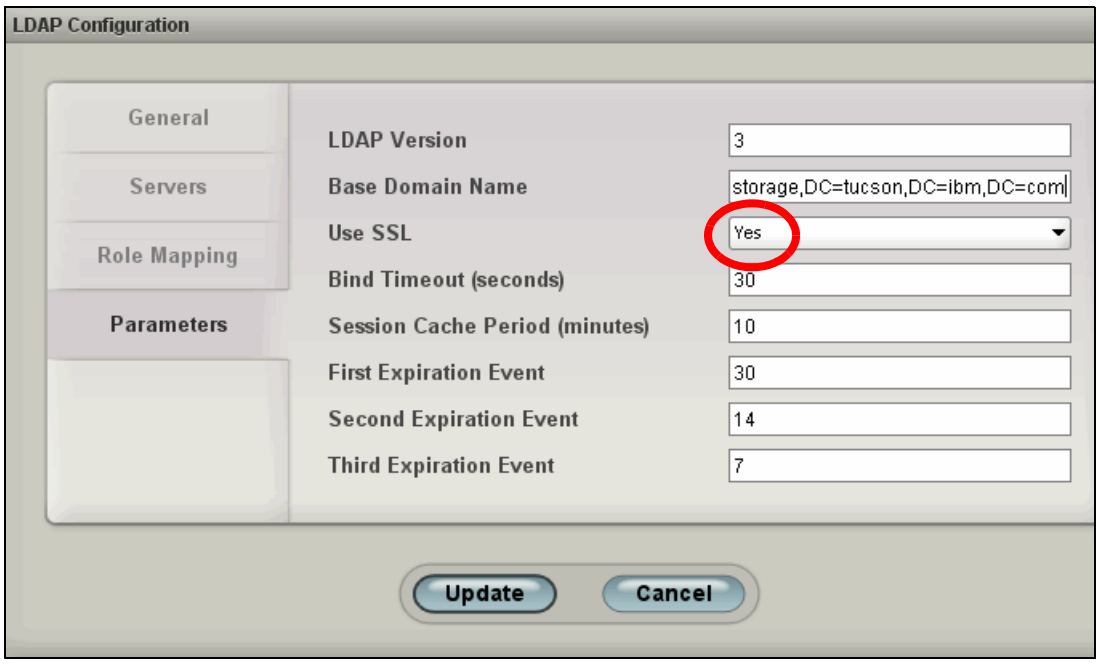


Figure 5-47 Enabling SSL for Active Directory LDAP communication

A new SSL certificate must be installed before the existing one expires. If you let your SSL certificate expire, XIV LDAP authentication will no longer be possible until you either disable SSL or install the new certificate on both the LDAP server and the XIV. Before the SSL certificate expires, XIV will issue three notification events. The first “SSL Certificate is About to Expire” event can be seen in Figure 5-48.

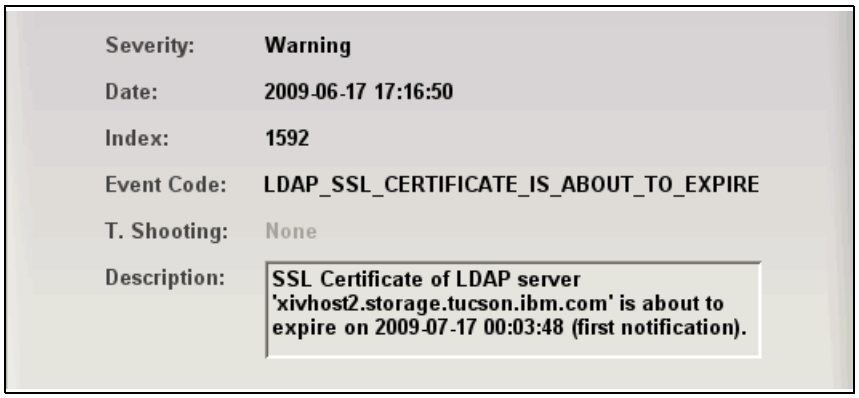


Figure 5-48 First notification of SSL Certificate of LDAP server expiration

5.5 XIV audit event logging

The XIV Storage System uses a centralized event log. For any command that has been executed that leads to a change in the system, an event entry is generated and recorded in the event log. The object creation time and the user are also as object attributes.

The event log is implemented as a circular log and is able to hold a set number of entries. When the log is full, the system wraps back to the beginning. If you need to save the log entries beyond what the system will normally hold, you can issue the XCLI command **`event_list`** and save the output to a file.

Event entries can be viewed by the GUI, XCLI commands, or via notification. A flexible system of filters and rules allows you to generate customized reports and notifications. For details about how to create customized rules, refer to 5.5.3, “Define notification rules” on page 181.

5.5.1 Viewing events in the XIV GUI

The XIV GUI provides a convenient and easy to use view of the event log. To get to the view shown in Figure 5-49, click the Monitor icon from the main GUI window and the select **Events** from the context menu.

The window is split into two sections:

- The top part contains the management tools, such as wizards in the menu bar and a series of input fields and drop-down menus that act as selection filters.
- The bottom part is a table displaying the events according to the selection criteria. Use the table tile bar or headings to enable or change sort direction.

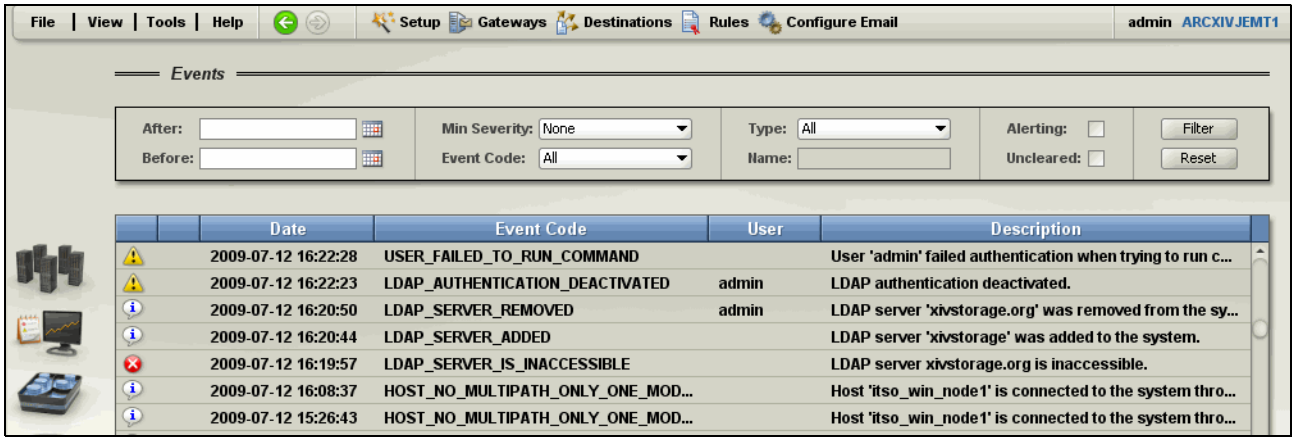


Figure 5-49 GUI events main view

The system will progressively load the events into the table. A progress indicator is visible at the bottom right of the table, as shown in Figure 5-50.

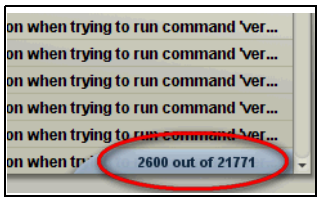


Figure 5-50 Loading events into the table

Event attributes

This section provides an overview of all available event types, event codes, and their severity levels.

Severity levels

You can select one of six possible severity levels as the minimal level to be displayed:

- **none**: Includes all severity levels
- **informational**: Changes, such as volume deletion, size changes, or host multipathing

- **warning**: Volume usage limits reach 80%, failing message sent
- **minor**: Power supply power input loss, volume usage over 90%, component TEST failed
- **major**: Component failed (disk), user system shutdown, volume and pool usage 100%, UPS on battery or Simple Mail Transfer Protocol (SMTP) gateway unreachable
- **critical**: Module failed or UPS failed

Event codes

Refer to the *XCLI Reference Guide*, GC27-2213-00, for a list of event codes.

Event types

The following event types can be used as filters (specified with the parameter **`object_type`** in the XCLI command):

- cons_group: consistency group
- destgroup:event destination group
- dest: event notification address
- dm: data migration
- host: host
- map: volume mapping
- mirror: mirroring
- pool: pool
- rule: rule
- smsgw: sms gateway
- smtpgw: smtp gateway
- target: fc/iSCSI connection
- volume: volume
- cluster:cluster
- user:user
- user_group:user group
- ip_interface: ip interface
- ldap_conf: ldap configuration

5.5.2 Viewing events in the XCLI

Table 5-6 provides a list of all the event-related commands available in the XCLI. This list covers setting up notifications and viewing the events in the system. Refer to Chapter 14, “Monitoring” on page 313 for a more in-depth discussion of system monitoring.

Table 5-6 XCLI: All event commands

Command	Description
custom_event	Generates a custom event.
dest_define	Defines a new destination for event notifications.
dest_delete	Deletes an event notification destination.
dest_list	Lists event notification destinations.
dest_rename	Renames an event notification destination.
dest_test	Sends a test message to an event notification destination.
dest_update	Updates a destination.
destgroup_add_dest	Adds an event notification destination to a destination group.

Command	Description
destgroup_create	Creates an event notification destination group.
destgroup_delete	Deletes an event notification destination group.
destgroup_list	Lists destination groups.
destgroup_remove_dest	Removes an event notification destination from a destination group.
destgroup_rename	Renames an event notification destination group.
event_clear	Clears alerting events.
event_list	Lists system events.
event_list_uncleared	Lists uncleared alerting events.
event_redefine_threshold	Redefines the threshold of a parameterized event.
smsgw_define	Defines a Short Message Service (SMS) gateway.
smsgw_delete	Deletes an SMS gateway.
smsgw_list	Lists SMS gateways.
smsgw_prioritize	Sets the priorities of the SMS gateways for sending SMS messages.
smsgw_rename	Renames an SMS gateway.
smsgw_update	Updates an SMS gateway.
smtpgw_define	Defines an SMTP gateway.
smtpgw_delete	Deletes a specified SMTP gateway.
smtpgw_list	Lists SMTP gateways.
smtpgw_prioritize	Sets the priority of which SMTP gateway to use to send e-mails.
smtpgw_rename	Renames an SMTP gateway.
smtpgw_update	Updates the configuration of an SMTP gateway.
rule_activate	Activates an event notification rule.
rule_create	Creates an event notification rule.
rule_deactivate	Deactivates an event notification rule.
rule_delete	Deletes an event notification rule.
rule_list	Lists event notification rules.
rule_rename	Renames an event notification rule.
rule_update	Updates an event notification rule.

Event_list command and parameters

The syntax of the event_list command is:

```
event_list
[max_events=MaxEventsToList]
[after=<afterTimeStamp|ALL>]
[before=<beforeTimeStamp|ALL>]
[min_severity=<informational|warning|minor|major|critical> ]
[alerting=<yes|no>]
[cleared=<yes|no>]
[code=EventCode]
[object_type=<cons_group | destgroup | dest|dm | host | map | mirror | pool | rule
smsgw | smtpgw | target | volume> ]
[ beg=BeginIndex ]
[ end=EndIndex ]
[ internal=<yes|no|all> ]
```

XCLI examples

To illustrate how the commands operates, the **`event_list`** command displays the events currently in the system. Example 5-39 shows the first few events logged in our system.

Example 5-39 XCLI viewing events

```
C:\XIV>xcli -c -c "ARCXIVJEMT1" event_list
Timestamp             Severity        Code          User  Description
2009-07-09 12:49:44   Informational   UNMAP_VOLUME ITSO   Volume with name 'Win_Vol_2' was
unmapped from cluster with name 'Win2008Cluster01'.
2009-07-09 12:49:48   Informational   UNMAP_VOLUME ITSO   Volume with name 'Win_Vol_1' was
unmapped from cluster with name 'Win2008Cluster01'.
2009-07-09 12:50:05   Informational   UNMAP_VOLUME ITSO   Volume with name 'Win_Vol_3' was
unmapped from host with name 'Win2008C1H1'.
2009-07-09 12:50:06   Informational   MAP_VOLUME   ITSO   Volume with name 'Win_Vol_3' was
mapped to LUN '1' for host with name 'Win2008C1H1'.
2009-07-09 12:50:24   Informational   UNMAP_VOLUME ITSO   Volume with name 'Win_Vol_4' was
unmapped from host with name 'Win2008C1H2'.
.......
```

Example 5-40 illustrates the command for listing all instances when the user was updated. The USER_UPDATED event is generated when a user's password, e-mail, or phone number is modified. In this example, the -t option is used to display specific fields, such as index, code, description of the event, time stamp, and user name. The description field provides the ID that was modified, and the user field is the ID of the user performing the action.

Example 5-40 View USER_UPDATED event with the XCLI

```
C:\XIV>xcli -c "ARCXIVJEMT1" -t index,code,description,timestamp,user_name event_list
code=USER_UPDATED
Index  Code           Description                              Timestamp        User
1089   USER_UPDATED User with name 'admin' was updated.       2009-07-10 21:45:50  admin
1091   USER_UPDATED User with name 'admin' was updated.       2009-07-10 21:45:51  admin
1092   USER_UPDATED User with name 'technician' was updated. 2009-07-10 21:45:52
xiv_development
1093   USER_UPDATED User with name 'technician' was updated.  2009-07-10 21:45:53
xiv_development
```

5.5.3 Define notification rules

Example 5-41 describes how to set up a rule in the XCLI to notify the storage administrator when a user's access control has changed. The rule itself has four event codes that generate a notification. The events are separated with commas with no spaces around the commas. If any of these four events are logged, the XIV Storage System uses the "relay" destination to issue the notification.

Example 5-41 Setting up an access notification rule using the XCLI

```
C:\XIV>xcli -c -c "ARCXIVJEMT1" rule_create rule=test
codes=ACCESS_OF_USER_GROUP_TO_CLUSTER_REMOVED,ACCESS_OF_USER_GROUP_TO_HOST_REMOVED
,ACCESS_TO_CLUSTER_GRANTED_TO_USER_GROUP,ACCESS_TO_HOST_GRANTED_TO_USER_GROUP
dests=relay
Command executed successfully.
```

A simpler example is setting up a rule notification for when a user account is modified. Example 5-42 creates a rule on the XIV Storage System called ESP™ that sends a notification whenever user account is modified on the system. The notification is transmitted through the relay destination.

Example 5-42 Create a rule for notification with the XCLI

```
C:\XIV>xcli -c -c "ARCXIVJEMT1" rule_create rule=user_update codes=USER_UPDATED
dests=relay
Command executed successfully.
```

The same rule can be created in the GUI. Refer to Chapter 14, "Monitoring" on page 313 for more details about configuring the system to provide notifications and setting up rules.

6

Host connectivity

This chapter discusses the host connectivity for the XIV Storage System. It addresses key aspects of host connectivity and reviews concepts and requirements for both Fibre Channel (FC) and Internet Small Computer System Interface (iSCSI) protocols.

The term *host* in this chapter refers to a server running a supported operating system such as AIX or Windows. SVC as a host has special considerations because it acts as both a host and a storage device. SVC is covered in more detail in Chapter 12, “SVC specific considerations” on page 293.

This chapter does not include attachments from a secondary XIV used for Remote Mirroring, nor does it include a legacy storage subsystem used for data migration.

This chapter covers common tasks that pertain to all hosts. For operating system-specific information regarding host attachment, refer to the corresponding subsequent chapters of this book.

6.1 Overview

The XIV Storage System can be attached to various host platforms using the following methods:

- Fibre Channel adapters for support with the Fibre Channel Protocol (FCP)
- iSCSI software initiator for support with the iSCSI protocol

This choice gives you the flexibility to start with the less expensive iSCSI implementation, using an already available Ethernet network infrastructure. Most companies have existing Ethernet connections between their locations and can use that infrastructure to implement a less expensive backup or disaster recovery setup. Imagine taking a snapshot of a critical server and being able to serve the snapshot through iSCSI to a remote data center server for backup. In this case, you can simply use the existing network resources without the need for expensive FC switches. As soon as workload and performance requirements justify it, you can progressively convert to a more expensive Fibre Channel infrastructure. From a technical standpoint and after HBAs and cabling are in place, the migration is easy. It only requires the XIV storage administrator to add the HBA definitions to the existing host configuration to make the logical unit numbers (LUNs) visible over FC paths.

As described in “Interface Module” on page 54, the XIV Storage System has six Interface Modules. Each Interface Module has four Fibre Channel ports, and three Interface Modules (Modules 7-9) also have two iSCSI ports each. These ports are used to attach hosts (as well as remote XIVs or legacy storage subsystems) to the XIV via the internal patch panel.

The patch panel simplifies cabling as the Interface Modules are pre-cabled to the patch panel so that all customer SAN and network connections are made in one central place at the back of the rack. This also helps with general cable management.

Hosts attach to the FC ports through an FC switch and to the iSCSI ports through a Gigabit Ethernet switch; Direct attach connections are not supported.

Restriction: Direct attachment between hosts and the XIV Storage System is currently not supported.

Figure 6-1 gives an example of how to connect a host through either a Storage Area Network (SAN) or an Ethernet network to the XIV Storage System; for clarity, the patch panel is not shown here.

Important: Host traffic can be served through any of the six Interface Modules. However, I/Os are not automatically balanced by the system. It is the storage administrator’s responsibility to ensure that host connections avoid single points of failure and that the host workload is adequately balanced across the connections and Interface Modules. This should be reviewed periodically or when traffic patterns change.

With XIV, all interface modules and all ports can be used concurrently to access any logical volume in the system. The only affinity is the mapping of logical volumes to host, and this simplifies storage management. Balancing traffic and zoning (for adequate performance and redundancy) is more critical, although not more complex, than with traditional storage subsystems.

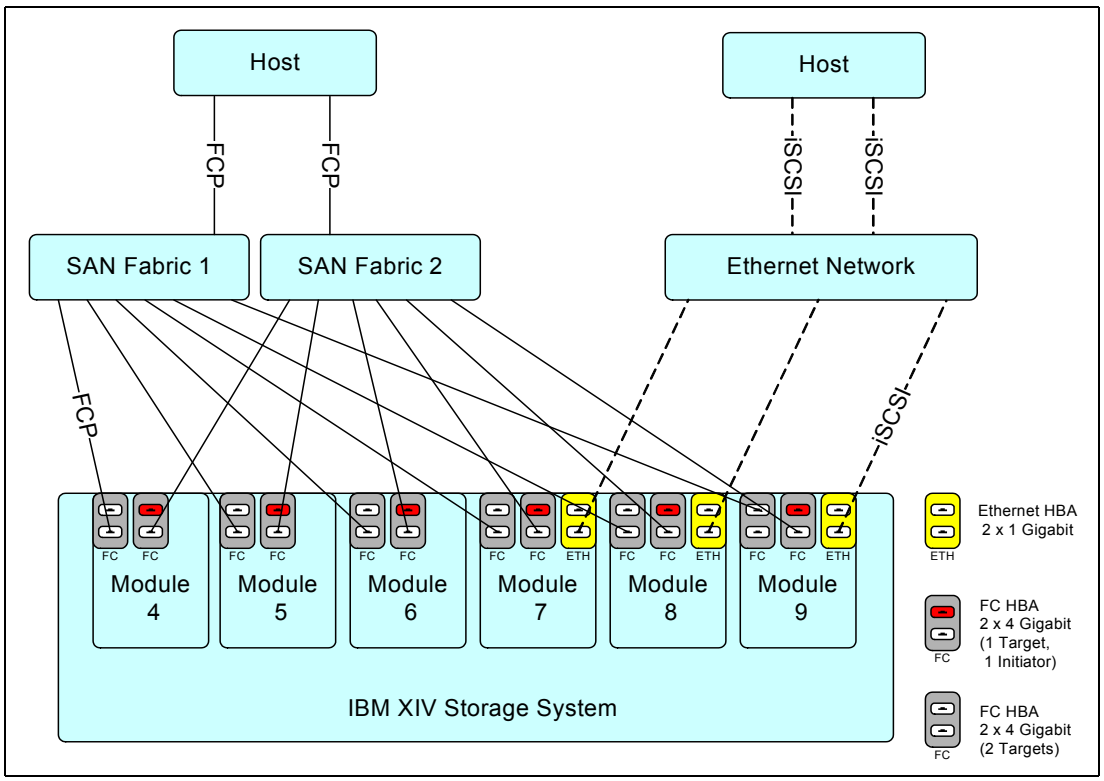


Figure 6-1 Host connectivity overview (without patch panel)

6.1.1 Module, patch panel, and host connectivity

This section presents a simplified view of the host connectivity. It is intended to explain the relationship between individual system components and how they affect host connectivity. Refer to 3.2, “IBM XIV hardware components” on page 46 for more details and an explanation of the individual components.

When connecting hosts to the XIV, there is no “one size fits all” solution that can be applied because every environment is different. However, we recommend that you use the following guidelines to ensure that there are no single points of failure and that hosts are connected to the correct ports:

- FC hosts connect to the XIV patch panel FC ports 1 and 3 on Interface Modules 4-9.
- XIV patch panel FC ports 2 and 4 should be used for mirroring to another XIV Storage System and/or for data migration from a legacy storage system. If mirroring or data migration will not be used then ports 2 and 4 can be used for additional host connections (port 4 must first be changed from its default initiator role to target). However, additional ports provide “fan out” capability and not additional throughput (see next **Note** box).
- iSCSI hosts connect to iSCSI ports 1 and 2 on Interface Modules 7-9.
- Hosts should have a connection path to separate Interface Modules to avoid a single point of failure.
- When using SVC as a host, all 12 available FC host ports on the XIV patch panel (ports 1 and 3 on Modules 4-9) should be used for SVC and nothing else. All other hosts to access the XIV through the SVC.

Note: Using the remaining 12 ports will provide the ability to manage devices on additional ports, but will not necessarily provide additional system bandwidth.

Figure 6-2 illustrates on overview of FC and iSCSI connectivity.

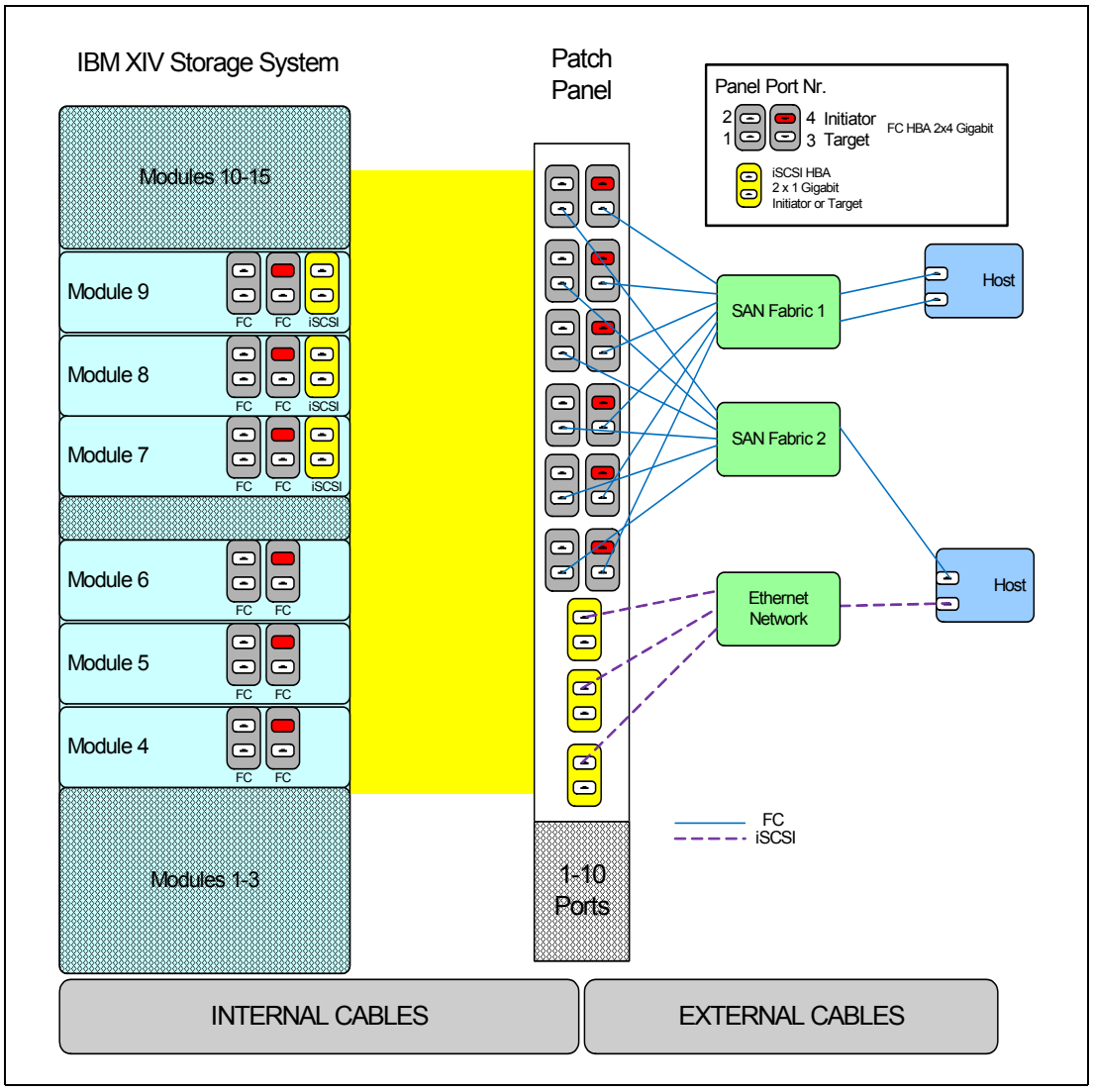


Figure 6-2 Host connectivity end-to-end view: Internal cables compared to external cables

Example 6-3 provides an XIV patch panel to FC and a patch panel to iSCSI adapter mappings. It also shows the World Wide Port Names (WWPNs) and iSCSI Qualified Names (IQNs) associated with the ports.

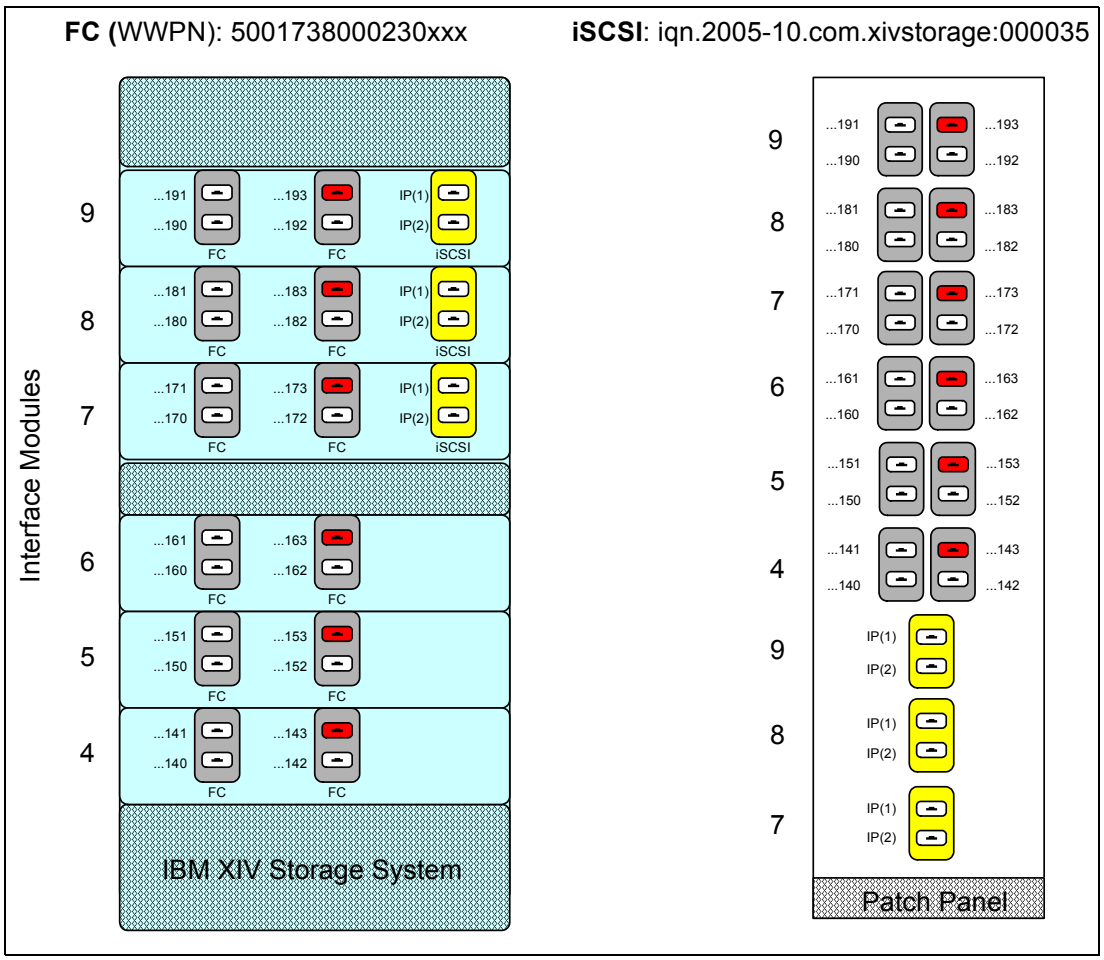


Figure 6-3 Patch panel to FC and iSCSI port mappings

A more detailed view of host connectivity and configuration options is provided in 6.2, “Fibre Channel (FC) connectivity” on page 190 and in 6.3, “iSCSI connectivity” on page 201.

6.1.2 Host operating system support

The XIV Storage System supports many operating systems, and the list is constantly growing. Here is a list of some of the supported operating systems at the time or writing:

- AIX
- ESX
- Linux (RHEL, SuSE)
- HP-UX
- VIOS (a component of Power/VM)
- Solaris
- SVC
- Windows

To get the current list when you implement your XIV, refer to the IBM System Storage Interoperation Center (SSIC) at the following Web site:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

6.1.3 Host Attachment Kits

With version 10.1.x of the XIV system software, IBM also provides updates to all of the Host Attachment Kits (version 1.1 or later). With the exception of the AIX, Host Attachment Kits (HAKs) are built on a Python framework with the intention of providing a consistent look and feel across various OS platforms. Features include these:

- Backwards compatibility with versions 9.2.x and 10.0.x of the XIV system software
- Validates patch and driver versions
- Sets up multipathing
- Adjusts system tunable parameters (if required) for performance
- Installation wizard
- Includes management utilities
- Includes support and troubleshooting utilities

Host Attachment Kits can be downloaded from the following Web site:

http://www.ibm.com/support/search.wss?q=ssg1*&tc=STJTAG+HW3EO&rs=1319&dc=D400&dtm

6.1.4 FC versus iSCSI access

XIV provides access to both FC and iSCSI hosts. The current version of XIV system software at the time of writing (10.1.x) supports iSCSI using the software initiator only.

The choice of connection protocol (iSCSI of FCP) should be considered with determination made based on application requirements. When considering IP storage based connectivity, consideration must also take into account the performance and availability of the existing corporate infrastructure.

We offer the following recommendations:

- FC hosts in a production environment should always be connected to a minimum of two separate SAN switches, in independent fabrics to provide redundancy.
- For test and development, there can be single points of failure to reduce costs. However, you will have to determine if this practice is acceptable for your environment.
- When using iSCSI use, a separate section of the IP network to isolate iSCSI traffic using either a VLAN or a physically separated section. Storage access is very susceptible to latency or interruptions in traffic flow and therefore should not be mixed with other IP traffic.

A host can connect via FC and iSCSI simultaneously, however, it should not connect to the same LUN over different protocols unless it is supported by the operating system and multipathing driver. If a LUN needs to connect over both protocols, then an alternative way of doing this is to create two hosts, one for the FC connections and one for the iSCSI connections. Figure 6-4 and Figure 6-5 illustrate these two options.

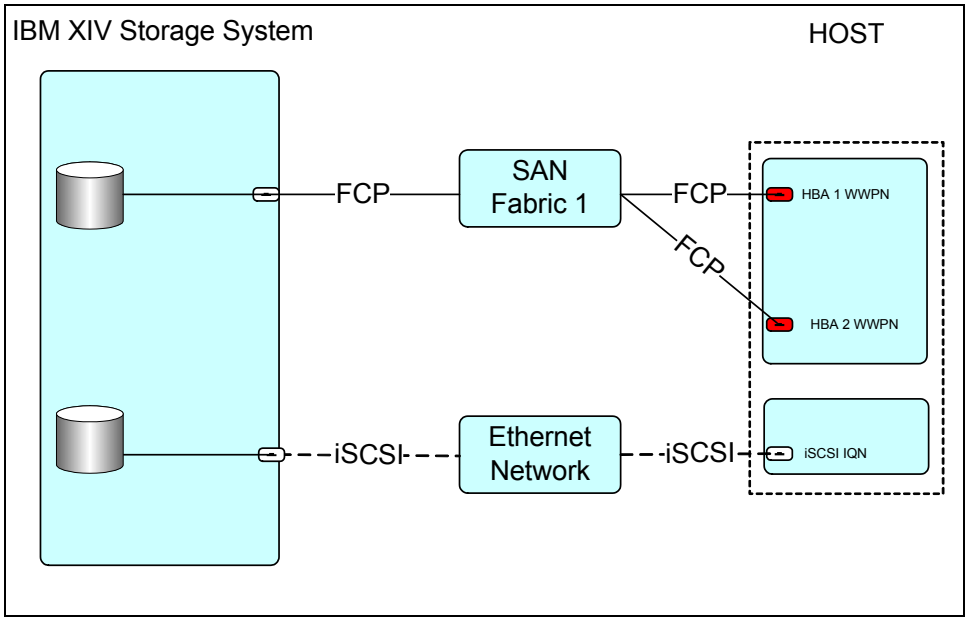


Figure 6-4 Host connectivity FCP and iSCSI simultaneously using separate host objects

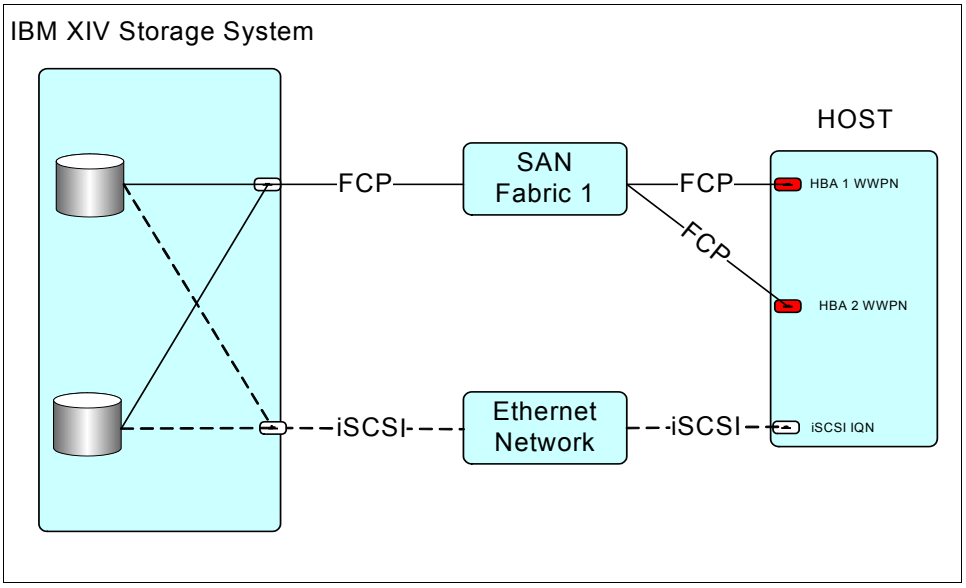


Figure 6-5 Host connectivity FCP and iSCSI simultaneously using the same host object

6.2 Fibre Channel (FC) connectivity

This section focuses on FC connectivity that applies to the XIV Storage System in general. For operating system-specific information, refer to the relevant section in the corresponding subsequent chapters of this book.

6.2.1 Preparation steps

Before you can attach an FC host to the XIV Storage System, there are a number of procedures that you must complete. Here is a list of general procedures that pertain to all hosts, however, you need to also review any procedures that pertain to your specific hardware and/or operating system.

1. Ensure that your HBA is supported. Information about supported HBAs and the recommended or required firmware and device driver levels is available at the IBM System Storage Interoperability Center (SSIC) Web site at:

   http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

   For each query, select the XIV Storage System, a host server model, an operating system, and an HBA vendor. Each query shows a list of all supported HBAs. Unless otherwise noted in SSIC, use any supported driver and firmware by the HBA vendors (the latest versions are always preferred). For HBAs in Sun systems, use Sun branded HBAs and Sun ready HBAs only.

   You should also review any documentation that comes from the HBA vendor and ensure that any additional conditions are met.

2. Check the LUN limitations for your host operating system and verify that there are enough adapters installed on the host server to manage the total number of LUNs that you want to attach.
3. Check the optimum number of paths that should be defined. This will help in determining the zoning requirements.
4. Install the latest supported HBA firmware and driver. If these are not the one that came with your HBA, they should be downloaded.

6.2.2 FC configurations

Several configurations are technically possible, and they vary in terms of their cost and the degree of flexibility, performance, and reliability that they provide.

Production environments must always have a redundant (high availability) configuration. There should be no single points of failure. Hosts should have as many HBAs as needed to support the operating system, application and overall performance requirements.

For test and development environments, a non-redundant configuration is often the only practical option due to cost or other constraints. Also, this will typically include one or more single points of failure.

Next, we review two typical FC configurations that are supported.

Redundant configuration

A redundant configuration is illustrated in Figure 6-6.

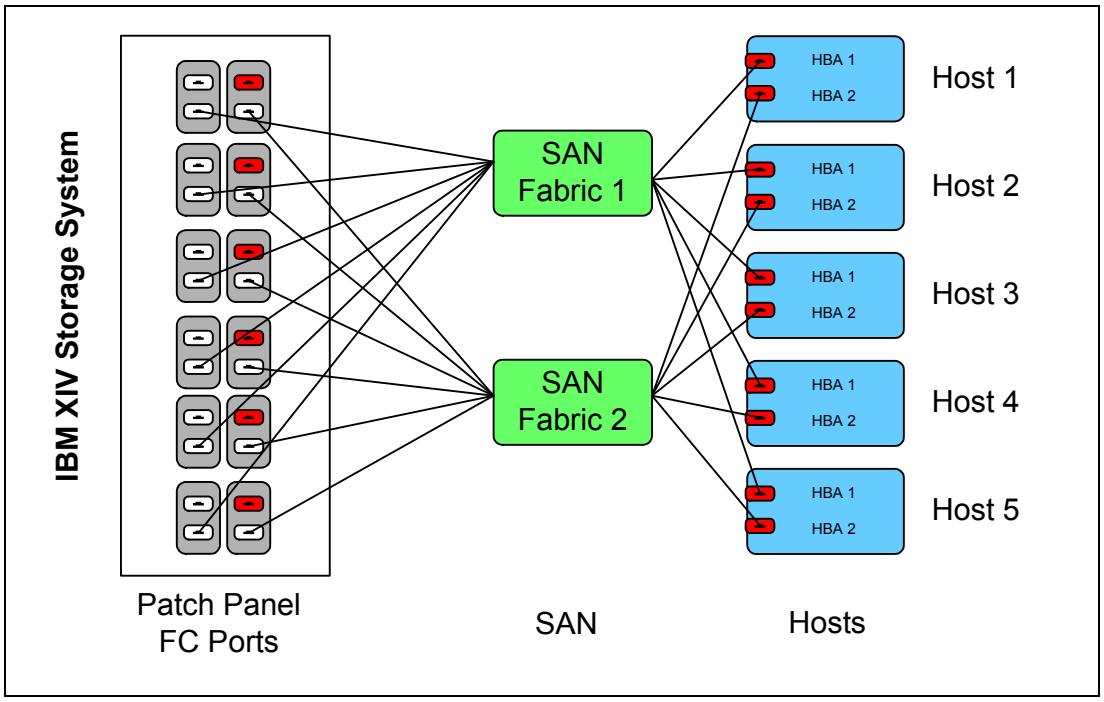


Figure 6-6 FC redundant configuration

In this configuration:

- Each host is equipped with dual HBAs. Each HBA (or HBA port) is connected to one of two FC switches.
- Each of the FC switches has a connection to a separate FC port of each of the six Interface Modules.

This configuration has no single point of failure:

- If a Module fails, each host remains connected to at least one other module. How many depends on the zoning, but it would typically be three or more other modules.
- If an FC switch fails, each host remains connected to at least one other module. How many depends on the zoning, but it would typically be two or more other modules.
- If a host HBA fails, each host remains connected to at least one other module. How many depends on the zoning, but it would typically be two or more other modules.
- If a host cable fails, each host remains connected to at least one other module. How many depends on the zoning, but it would typically be two or more other modules.

Non-redundant configurations (not recommended)

Non-redundant configurations should only be used for test and development where the risks of a single point of failure are acceptable. This is illustrated in Figure 6-7.

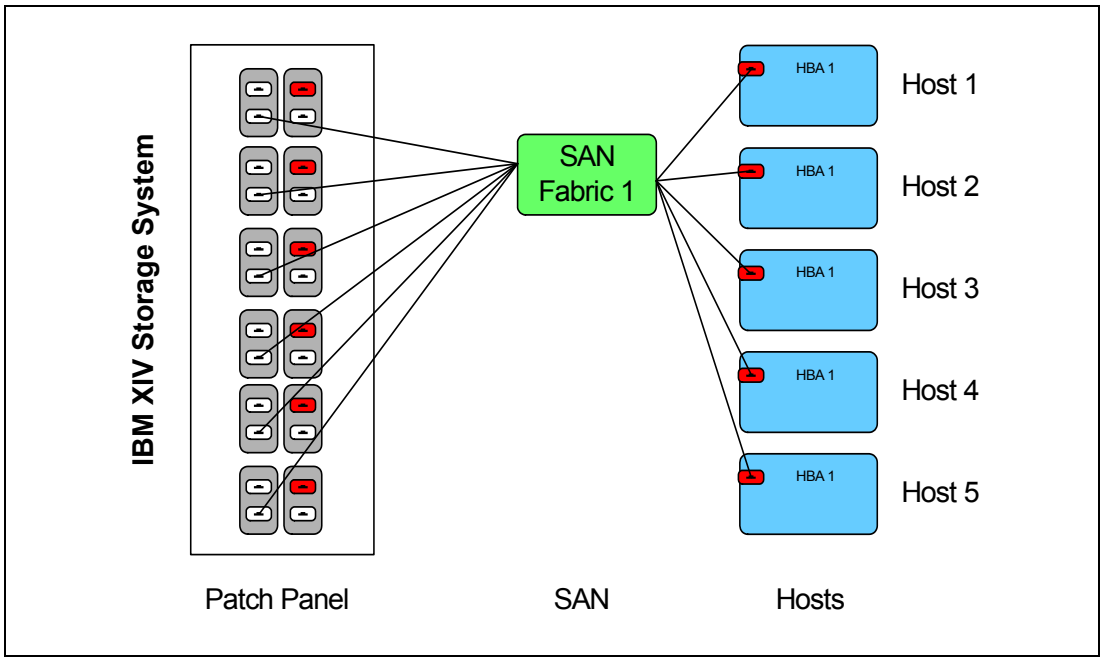


Figure 6-7 FC configurations: Single switch

6.2.3 Zoning

Zoning is mandatory when connecting FC hosts to an XIV Storage System. Zoning is configured on the SAN switch and is a boundary whose purpose is to isolate FC traffic to only those HBAs within a given zone.

A zone can be either a hard zone or a soft zone. Hard zones group HBAs depending on the physical ports they are connected to on the SAN switches. Soft zones group HBAs depending on the World Wide Port Names (WWPNs) of the HBA. Each method has its merits and you will have to determine which is right for your environment.

Correct zoning helps avoid many problems and makes it easier to trace cause of errors. Here are some examples of why correct zoning is important:

- An error from an HBA that affects the zone or zone traffic will be isolated.
- Disk and tape traffic must be in separate zones as they have different characteristics. If they are in the same zone this can cause performance problem or have other adverse affects.
- Any change in the SAN fabric, such as a change caused by a server restarting or a new product being added to the SAN, triggers a *Registered State Change Notification* (RSCN). An RSCN requires that any device that can “see” the affected or new device to acknowledge the change, interrupting its own traffic flow.

Zoning guidelines

There are many factors that affect zoning these include; host type, number of HBAs, HBA driver, operating system and applications—as such, it is not possible to provide a solution to cover every situation. The following list gives some guidelines, which can help you to avoid reliability or performance problems. However, you should also review documentation regarding your hardware and software configuration for any specific factors that need to be considered:

- Each zone (excluding those for SVC) should have one initiator HBA (the host) and multiple target HBAs (the XIV Storage System).
- Each host (excluding SVC) should have two paths per HBA unless there are other factors dictating otherwise.
- Each host should connect to ports from at least two Interface Modules.
- Do not mix disk and tape traffic on the same HBA or in the same zone.

For more in-depth information about SAN zoning, refer to section 4.7 of the IBM Redbooks publication, *Introduction to Storage Area Networks*, SG24-5470.

You can download this publication from:

`http://www.redbooks.ibm.com/redbooks/pdfs/sg245470.pdf`

An example of soft zoning using the “single initiator - multiple targets” method is illustrated in Figure 6-8.

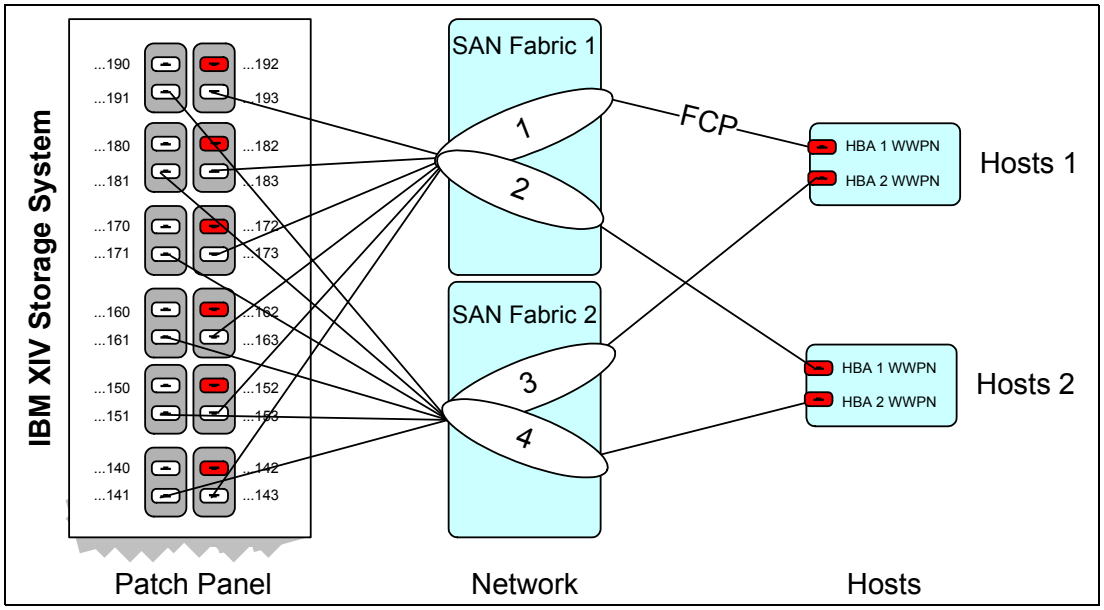


Figure 6-8 FC SAN zoning: single initiator - multiple target

Note: Use a single initiator multiple target zoning scheme. Do not share a host HBA for disk and tape access.

6.2.4 Identification of FC ports (initiator/target)

Identification of a port is required for setting up the zoning, to aid with any modifications that might be required or to assist with problem diagnosis. The unique name that identifies an FC port is called the World Wide Port Name (WWPN).

The easiest way to get a record of all the WWPNs on the XIV is to use the XCLI; However, this information is also available from the GUI. Example 6-1 shows all WWPNs for one of the XIV Storage Systems that we used in the preparation of this book. This example also shows the Extended Command Line Interface (XCLI) command to issue. Note that for clarity, some of the columns have been removed in this example.

Example 6-1 XCLI: How to get WWPN of IBM XIV Storage System

```
>> fc_port_list
Component ID   Status  Currently    WWPN               Port ID   Role
                       Functioning
1:FC_Port:4:1  OK      yes          5001738000230140   00030A00  Target
1:FC_Port:4:2  OK      yes          5001738000230141   00614113  Target
1:FC_Port:4:3  OK      yes          5001738000230142   00750029  Target
1:FC_Port:4:4  OK      yes          5001738000230143   00FFFFFF  Initiator
1:FC_Port:5:1  OK      yes          5001738000230150   00711000  Target
1:FC_Port:5:2  OK      yes          5001738000230151   0075001F  Target
1:FC_Port:5:3  OK      yes          5001738000230152   00021D00  Target
1:FC_Port:5:4  OK      yes          5001738000230153   00FFFFFF  Target
1:FC_Port:6:1  OK      yes          5001738000230160   00070A00  Target
1:FC_Port:6:2  OK      yes          5001738000230161   006D0713  Target
1:FC_Port:6:3  OK      yes          5001738000230162   00FFFFFF  Target
1:FC_Port:6:4  OK      yes          5001738000230163   00FFFFFF  Initiator
1:FC_Port:7:1  OK      yes          5001738000230170   00760000  Target
1:FC_Port:7:2  OK      yes          5001738000230171   00681813  Target
1:FC_Port:7:3  OK      yes          5001738000230172   00021F00  Target
1:FC_Port:7:4  OK      yes          5001738000230173   00021E00  Initiator
1:FC_Port:8:1  OK      yes          5001738000230180   00060219  Target
1:FC_Port:8:2  OK      yes          5001738000230181   00021C00  Target
1:FC_Port:8:3  OK      yes          5001738000230182   002D0027  Target
1:FC_Port:8:4  OK      yes          5001738000230183   002D0026  Initiator
1:FC_Port:9:1  OK      yes          5001738000230190   00FFFFFF  Target
1:FC_Port:9:2  OK      yes          5001738000230191   00FFFFFF  Target
1:FC_Port:9:3  OK      yes          5001738000230192   00021700  Target
1:FC_Port:9:4  OK      yes          5001738000230193   00021600  Initiator
```

Note that the **fc_port_list** command might not always print out the port list in the same order. When you issue the command, the rows might be ordered differently, however, all the ports will be listed.

To get the same information from the XIV GUI, select the main view of an XIV Storage System, use the arrow at the bottom (circled in red) to reveal the patch panel, and move the mouse cursor over a particular port to reveal the port details including the WWPN (refer to Figure 6-9).

Figure 6-9 GUI: How to get WWPNs of IBM XIV Storage System

Note: The WWPNs of an XIV Storage System are static. The last two digits of the WWPN indicate to which module and port the WWPN corresponds.

As shown in Figure 6-9, the WWPN is `5001738000230160`, which means that the WWPN is from module `6` port `1`. The WWPNs for the port are numbered from 0 to 3 whereas the physical the ports are numbered from 1 to 4.

The values that comprise the WWPN are shown in Example 6-2.

Example 6-2 WWPN illustration

```
If WWPN is 50:01:73:8N:NN:NN:RR:MP


5           NAA (Network Address Authority)
001738      IEEE Company ID
NNNNN       IBM XIV Serial Number in hex
RR          Rack ID   (01-ff, 0 for WWNN)
M           Module ID (1-f,   0 for WWNN)
P           Port ID   (0-7,   0 for WWNN)
```

6.2.5 FC boot from SAN

Booting from SAN opens up a number of possibilities that are not available when booting from local disks. It means that the operating systems and configuration of SAN based computers can be centrally stored and managed. This can provide advantages with regard to deploying servers, backup, and disaster recovery procedures.

To boot from SAN, you need to go into the HBA configuration mode, set the HBA BIOS to be *Enabled*, select at least one XIV target port and select a LUN to boot from. In practice you will typically configure 2-4 XIV ports as targets and you might have to enable the BIOS on two HBAs, however, this will depend on the HBA, driver and operating system. Consult the documentation that comes with you HBA and operating system.

At the time of writing, the following operating systems are fully supported using SAN boot:

- ESX 3.5
- Windows

Other operating systems—AIX, HP-UX, Linux RHEL, Linux (SusE) and Solaris—are supported via a Request for Price Quote (RPQ), a process by which IBM will test and verity a specific configuration for a customer. For further details on RPQs, contact your IBM Representative.

SAN boot in AIX is addressed in Chapter 8, “AIX host connectivity” on page 235.

Boot from SAN procedures

The procedures for setting up your server and HBA to boot from SAN will vary; this is mostly dependent on whether your server has an Emulex® or QLogic® HBA (or the OEM equivalent). The procedures in this section are for a QLogic HBA; If you have an Emulex card, the configuration panels will differ but the logical process will be the same:

1. Boot your server. During the boot process, press **CTRL-Q** when prompted to load the configuration utility and display the **Select Host Adapter** menu. See Figure 6-10.

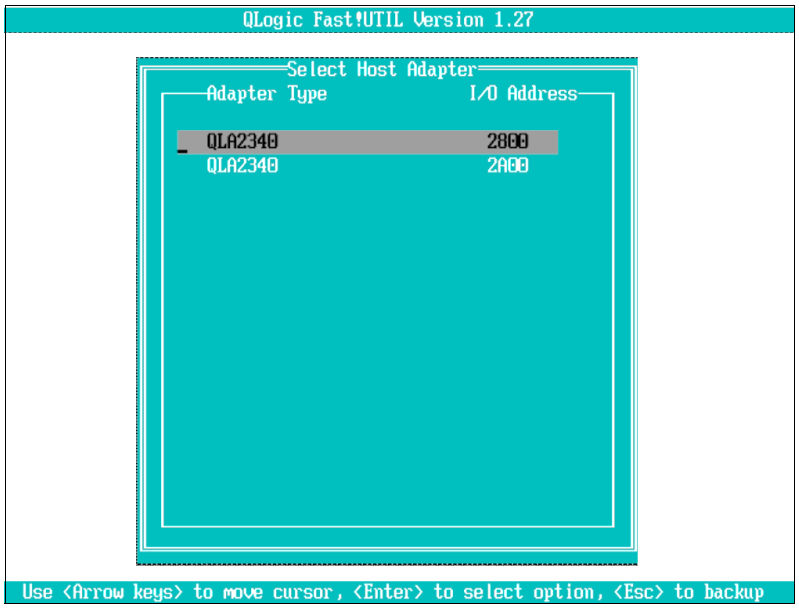


Figure 6-10 Select Host Adapter

2. You normally see one or more ports. Select a port and press Enter. This takes you to a panel as shown in Figure 6-11. Note that if you will only be enabling the BIOS on one port, then make sure to select the correct port.

   Select **Configuration Settings**.

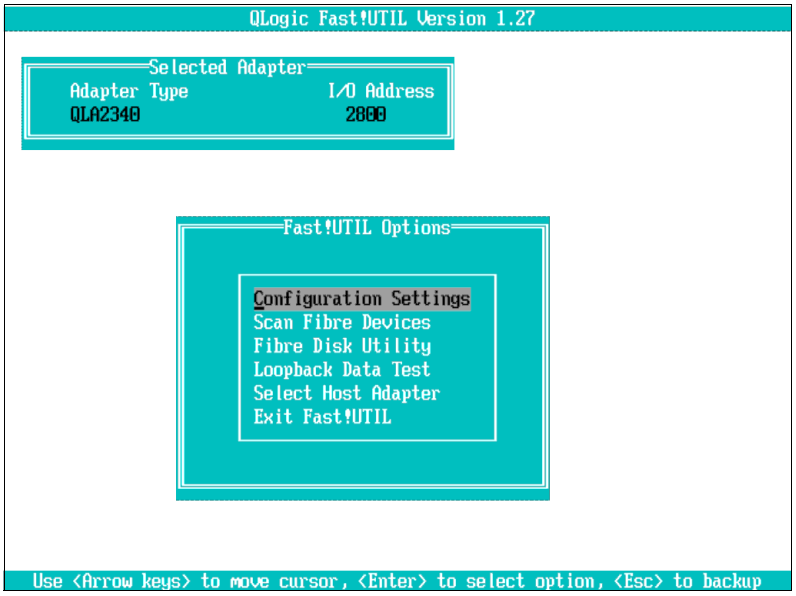


Figure 6-11 Fast!UTIL Options

3. In the panel shown in Figure 6-12, select **Adapter Settings**.

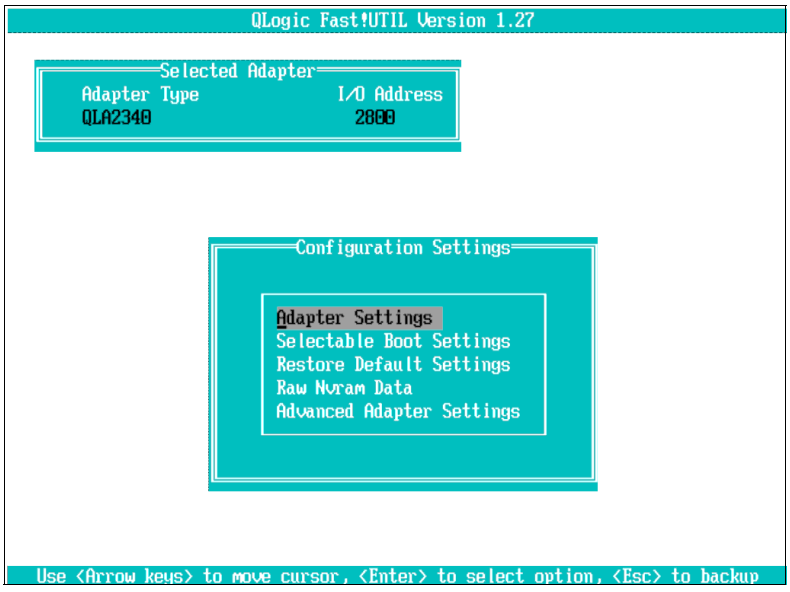


Figure 6-12 Configuration Settings

4. The **Adapter Settings** menu is displayed as shown in Figure 6-13.

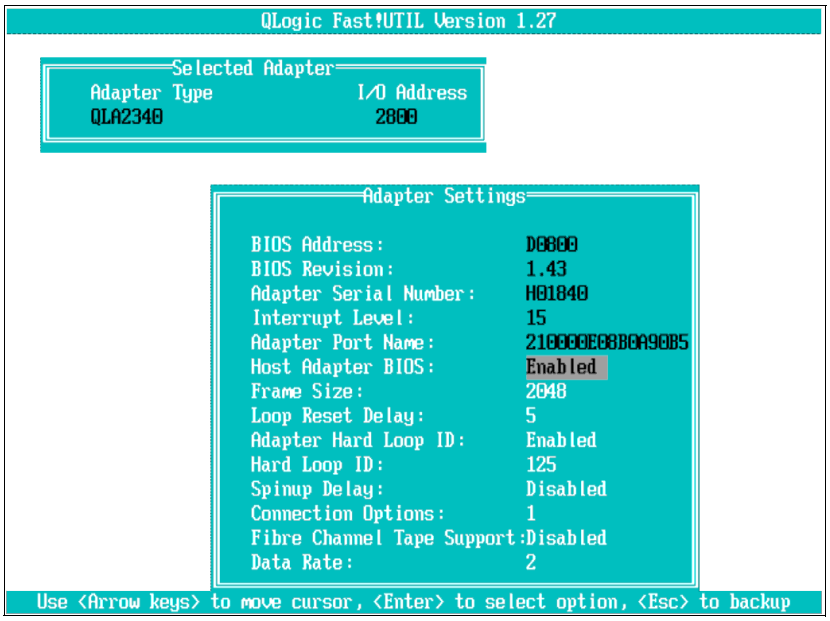


Figure 6-13 Adapter Settings

5. On the **Adapter Settings** panel, change the **Host Adapter BIOS** setting to **Enabled**, then press **Esc** to exit and go back to the **Configuration Settings** menu seen in Figure 6-12.
6. From the **Configuration Settings** menu, select **Selectable Boot Settings**, to get to the panel shown in Figure 6-14.

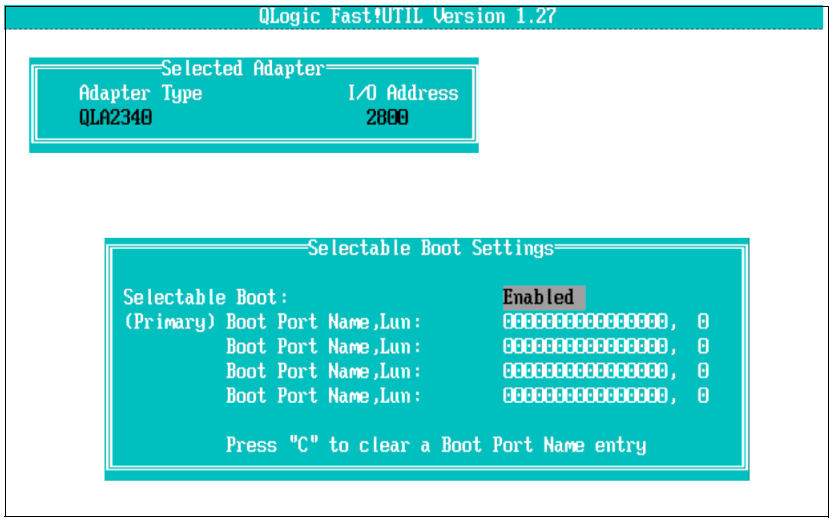


Figure 6-14 Selectable Boot Settings

7. Change **Selectable Boot** option to **Enabled**.

   Select **Boot Port Name, Lun:** and then press Enter to get the **Select Fibre Channel Device** menu, shown in Figure 6-15.

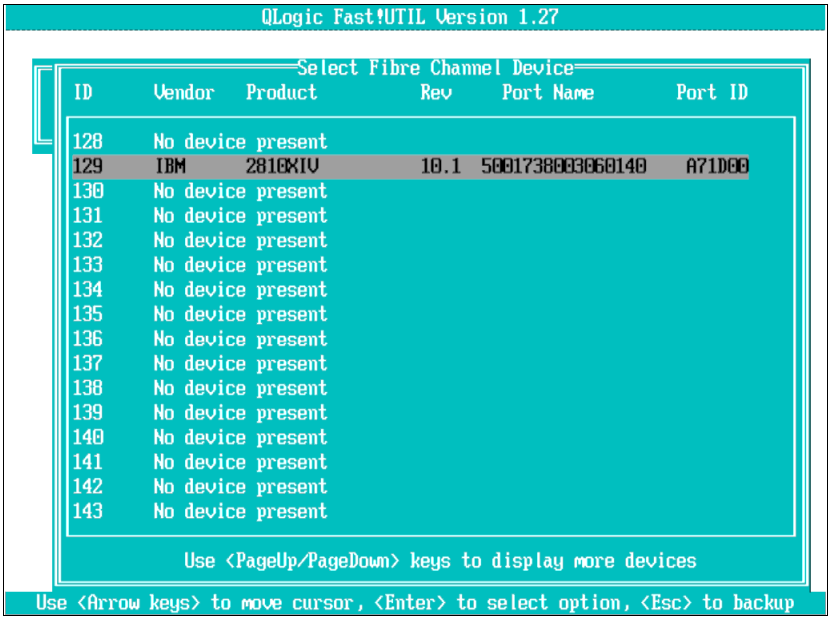


Figure 6-15 Select Fibre Channel Device

8. Select the **IBM 2810XIV** device, and press Enter, to display the **Select LUN** menu, seen in Figure 6-16.

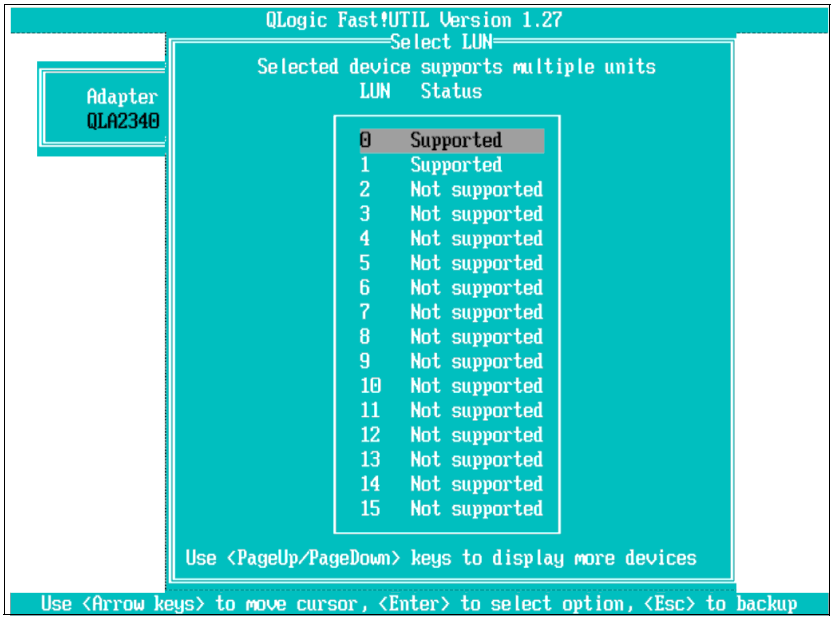


Figure 6-16 Select LUN

9. Select the boot LUN (in our case it is **LUN 0).** You are taken back to the **Selectable Boot Setting** menu and boot port with the boot LUN displayed as illustrated in Figure 6-17.

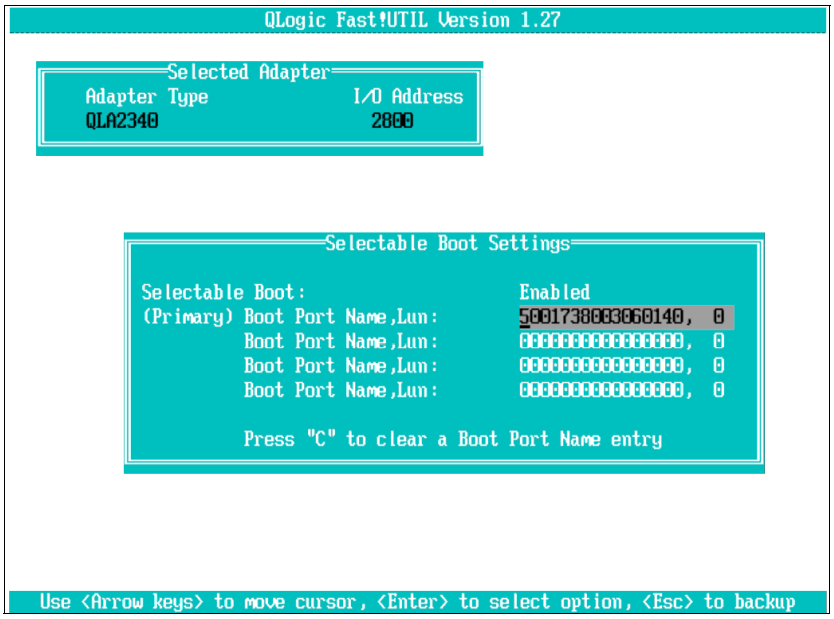


Figure 6-17 Boot Port selected

10. Repeat the steps 8-10 to add additional controllers. Note that any additional controllers must be zoned so that they point to the same boot LUN.
11. When all the controllers are added press Esc to exit (Configuration Setting panel). Press Esc again to get the **Save changes** option, as shown in Figure 6-18.

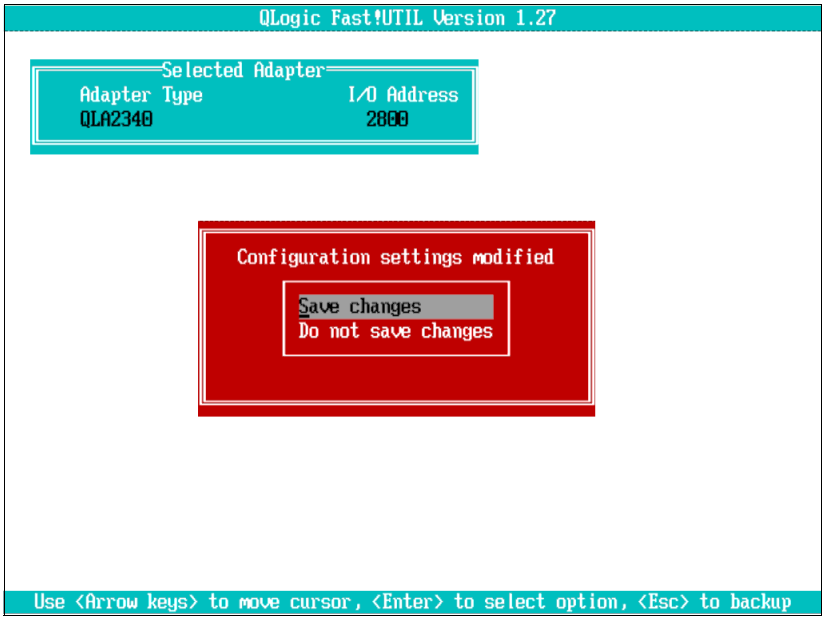


Figure 6-18 Save changes

12. Select **Save changes. T**his takes you back to the Fast!UTIL option panel. From there, select **Exit Fast!UTIL.**

13. The **Exit Fast!UTIL** menu is displayed as shown in Figure 6-19. Select **Reboot System** to reboot and boot from the newly configured SAN drive.

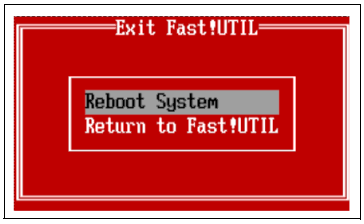


Figure 6-19 Exit Fast!UTIL

Important: Depending on your operating system and multipath drivers, you might need to configure multiple ports as “boot from SAN” ports. Consult your operating system documentation for more information.

6.3 iSCSI connectivity

This section focuses on iSCSI connectivity that applies to the XIV Storage System in general. For operating system-specific information, refer to the relevant section in the corresponding subsequent chapters of this book.

At the time of writing, with XIV system software 10.1.x iSCSI hosts are only supported using the software iSCSI initiator. Information about iSCSI software initiator support is available at the IBM System Storage Interoperability Center (SSIC) Web site at:

`http://www.ibm.com/systems/support/storage/config/ssic/index.jsp`

Table 6-1 shows some of the supported the operating systems.

Table 6-1 iSCSI supported operating systems

Operating System	Initiator
AIX	AIX iSCSI software initiator
Linux (CentOS)	Linux iSCSI software initiator Open iSCSI software initiator
Linux (RedHat)	RedHat iSCSI software initiator
Linux SuSE	Novell® iSCSI software initiator
Solaris	SUN iSCSI software initiator
Windows	Microsoft iSCSI software initiator

6.3.1 Preparation steps

Before you can attach an iSCSI host to the XIV Storage System, there are a number of procedures that you must complete. The following list describes general procedures that pertain to all hosts, however, you need to also review any procedures that pertain to your specific hardware and/or operating system:

1. Connecting host to the XIV over iSCSI is done using a standard Ethernet port on the host server. We recommend that the port you choose be dedicated to iSCSI storage traffic only. This port must also be a minimum of 1Gbps capable. This port will require an IP address, subnet mask and gateway.

   You should also review any documentation that comes with your operating system regarding iSCSI and ensure that any additional conditions are met.

2. Check the LUN limitations for your host operating system and verify that there are enough adapters installed on the host server to manage the total number of LUNs that you want to attach.

3. Check the optimum number of paths that should be defined. This will help in determining the number of physical connections that need to be made.

4. Install the latest supported adapter firmware and driver. If this is not the one that came with your operating system then it should be download.

5. Maximum Transmission Unit (MTU) configuration is required if your network supports an MTU that is larger than the default one which is 1500 bytes. Anything larger is known as a *jumbo frame*. The largest possible MTU should be specified, it is advisable to use up to 8192 bytes, if supported by the switches and routers. On the XIV, the MTU default value is 4500 bytes and the maximum value is 8192 bytes.

6. Any device using iSCSI requires an iSCSI Qualified Name (IQN) in our case it is the XIV Storage System and an attached host. The IQN uniquely identifies different iSCSI devices. The IQN for the XIV Storage System is configured when the system is delivered and must not be changed. Contact IBM technical support if a change is required.

   Our XIV Storage System's name was `iqn.2005-10.com.xivstorage:000035`.

6.3.2 iSCSI configurations

Several configurations are technically possible, and they vary in terms of their cost and the degree of flexibility, performance and reliability that they provide.

In the XIV Storage System, each iSCSI port is defined with its own IP address. Ports cannot be bonded.

Important: Link aggregation is not supported. Ports cannot be bonded

By default, there are six predefined iSCSI target ports on the XIV Storage System to serve hosts through iSCSI.

Redundant configurations

A redundant configuration is illustrated in Figure 6-20.

In this configuration:

- Each host is equipped with dual Ethernet interfaces. Each interface (or interface port) is connected to one of two Ethernet switches.
- Each of the Ethernet switches has a connection to a separate iSCSI port of each of Interface Modules 7-9.

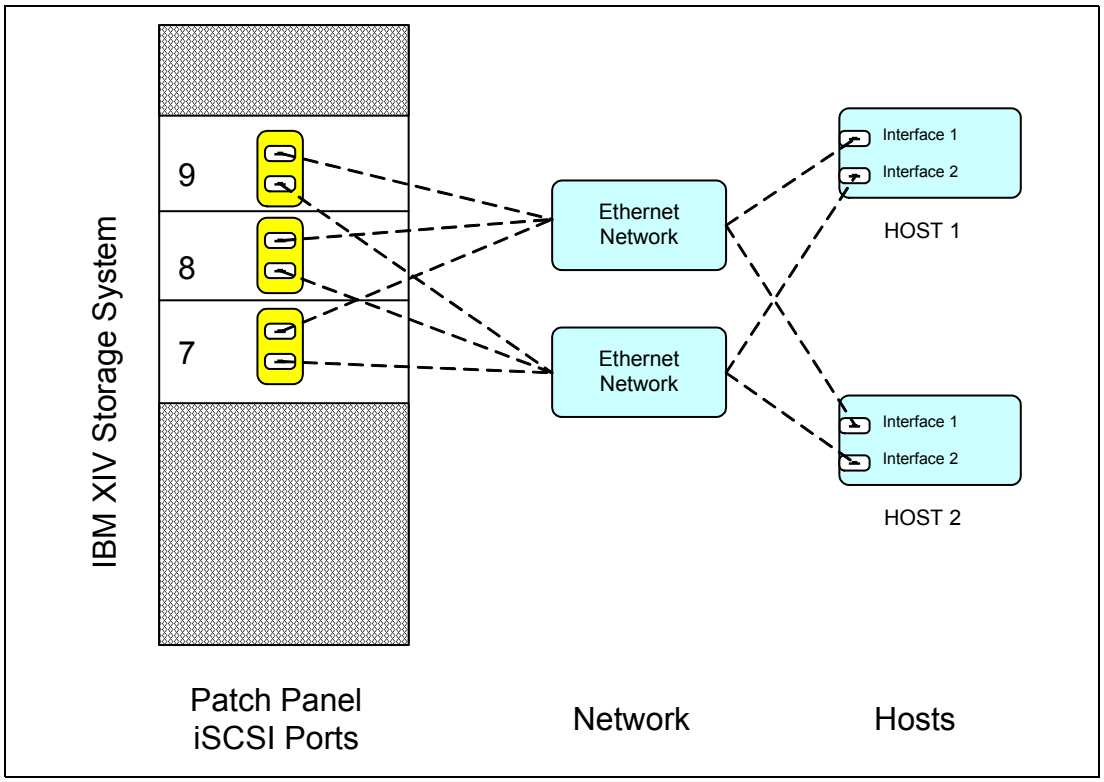


Figure 6-20 iSCSI configurations: redundant solution

This configuration has no single point of failure:

- If a module fails, each host remains connected to at least one other module. How many depends on the host configuration, but it would typically be one or two other modules.
- If an Ethernet switch fails, each host remains connected to at least one other module. How many depends on the host configuration, but it would typically be one or two or more other modules through the second Ethernet switch.
- If a host Ethernet interface fails, the host remains connected to at least one other module. How many depends on the host configuration, but it would typically be one or two other modules through the second Ethernet interface.
- If a host Ethernet cable fails, the host remains connected to at least one other module. How many depends on the host configuration, but it would typically be one or two other modules through the second Ethernet interface.

Note: For the best performance, use a dedicated iSCSI network infrastructure.

Non-redundant configurations

Non-redundant configurations should only be used where the risks of a single point of failure are acceptable. This is typically the case for test and development environments.

Figure 6-21 illustrates a non-redundant configuration.

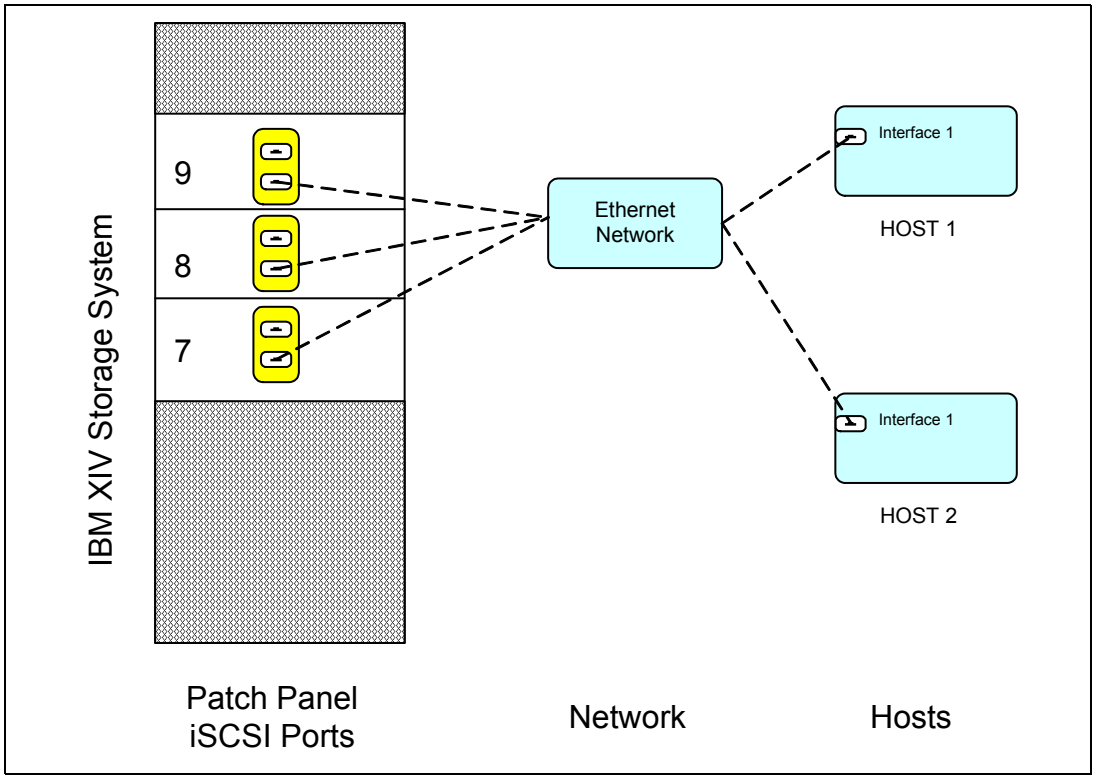


Figure 6-21 iSCSI configurations: Single switch

6.3.3 Link aggregation

Link aggregation is not supported with XIV system software 10.1.x. If you are currently using this feature with 10.0.x system software, the links should be removed and separate connections configured before upgrading.

6.3.4 Network configuration

Disk access is very susceptible to network latency. Latency can cause time-outs, delayed writes and/or possible data loss. In order to realize the best performance from iSCSI, all iSCSI IP traffic should reside on a dedicated network. This network should be a minimum of 1 Gbps and hosts should have interfaces dedicated to iSCSI only. For such configurations, additional host Ethernet ports might need to be purchased. Physical switches or VLANs should be used to provide a dedicated network.

6.3.5 IBM XIV Storage System iSCSI setup

Initially, no iSCSI connections are configured in the XIV Storage System. The configuration process is simple but requires more steps when compared to an FC connection setup.

Getting the XIV iSCSI Qualified Name (IQN)

Every XIV Storage System has a unique iSCSI Qualified Name (IQN). The format of the IQN is simple and includes a fixed text string followed by the last digits of the XIV Storage System serial number.

Important: Do not attempt to change the IQN. If a change is required, you must engage IBM support.

The IQN is visible as part of the XIV Storage System. From the XIV GUI, from the opening GUI panel (with all the systems) right-click on a system and select **Properties**. The System Properties dialog box is displayed, as shown in Figure 6-22.

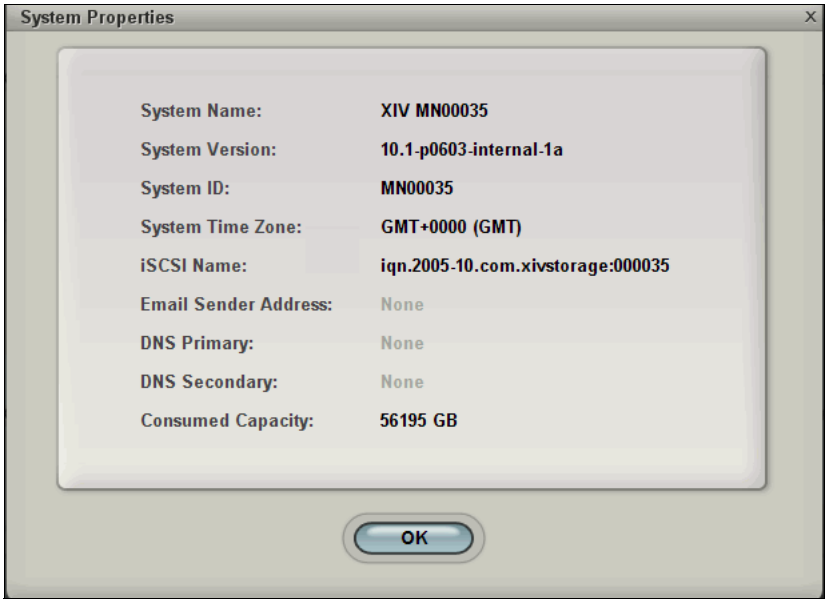


Figure 6-22 iSCSI: Use XIV GUI to get iSCSI name (IQN)

To show the same information in the XCLI, run the XCLI **config_get** command as shown in Example 6-3.

Example 6-3 iSCSI: use XCLI to get iSCSI name (IQN)

```
>> config_get
Name                        Value
dns_primary
dns_secondary
email_reply_to_address
email_sender_address
email_subject_format        {severity}: {description}
iscsi_name                  iqn.2005-10.com.xivstorage:000035
machine_model               A14
machine_serial_number       MN00035
machine_type                2810
ntp_server
snmp_community              XIV
snmp_contact                Unknown
snmp_location               Unknown
snmp_trap_community         XIV
support_center_port_type    Management
system_id                   35
system_name                 XIV MN00035
```

iSCSI XIV port configuration using the GUI

To set up the iSCSI port using the GUI:

1. Log on to the XIV GUI, select the XIV Storage System to be configured, move your mouse over the **Hosts and Clusters** icon. Select **iSCSI Connectivity** (refer to Figure 6-23).

Figure 6-23 GUI: iSCSI Connectivity menu option

2. The **iSCSI Connectivity** window opens. Click the Define icon at the top of the window (refer to Figure 6-24) to open the Define IP interface dialog.

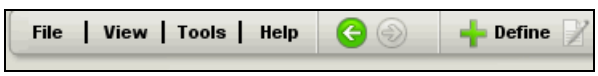


Figure 6-24 GUI: iSCSI Define interface icon

3. Enter the following information (refer to Figure 6-25):
   - Name: This is a name you define for this interface.
   - Address, netmask, and gateway: These are the standard IP address details.
   - MTU: The default is 4500. All devices in a network must use the same MTU. If in doubt, set MTU to 1500, because 1500 is the default value for Gigabit Ethernet. Performance might be impacted if the MTU is set incorrectly.
   - Module: Select the module to configure.
   - Port number: Select the port to configure .

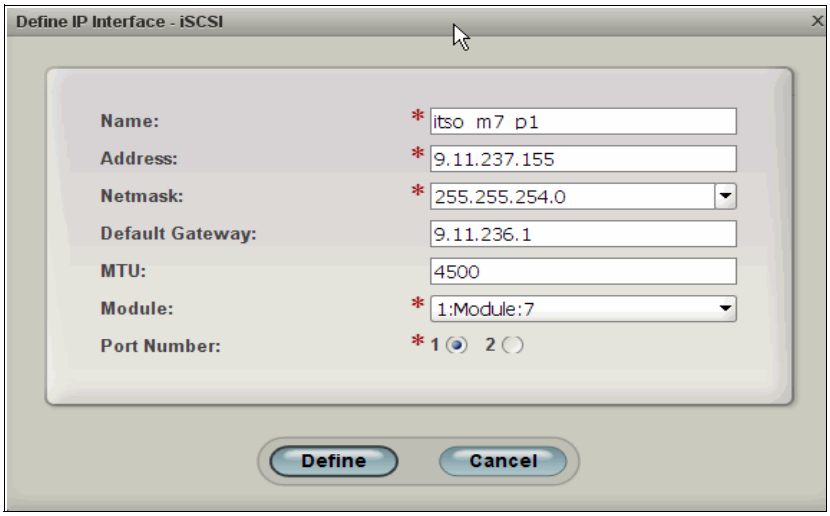


Figure 6-25 Define IP Interface: iSCSI setup window

4. Click **Define** to conclude defining the IP interface and iSCSI setup.

iSCSI XIV port configuration using the XCLI

Open an XCLI session tool, and use the *ipinterface_create* command; see Example 6-4.

Example 6-4 XCLI: iSCSI setup

```
>> ipinterface_create ipinterface=itso_m7_p1 address=9.11.237.155
netmask=255.255.254.0 gateway=9.11.236.1 module=1:module:7 ports=1
Command executed successfully.
```

6.3.6 Identifying iSCSI ports

iSCSI ports can easily be identified and configured in the XIV Storage System. Use either the GUI or an XCLI command to display current settings.

Viewing iSCSI configuration using the GUI

Log on to the XIV GUI, select the XIV Storage System to be configured and move the mouse over the **Hosts and Clusters** icon. Select **iSCSI connectivity** (refer to Figure 6-23 on page 206).

The iSCSI connectivity panel is displayed, this is shown in Figure 6-26. Right-click the port and select **Edit** from the context menu to make changes.

Note that in our example, only two of the six iSCSI ports are configured. Non-configured ports do not show up in the GUI.

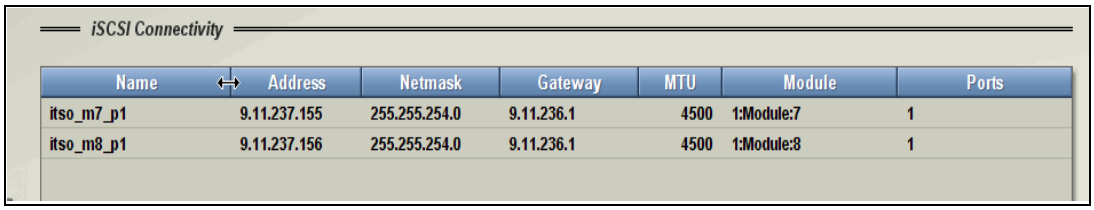


Figure 6-26 iSCSI connectivity

View iSCSI configuration using the XCLI

The **ipinterface_list** command illustrated in Example 6-5 can be used to display configured network ports only.

Example 6-5 XCLI to list iSCSI ports with ipinterface_list command

```
>> ipinterface_list
Name          Type         IP Address    Network Mask    Default Gateway MTU   Module
Ports
itso_m8_p1    iSCSI         9.11.237.156 255.255.254.0  9.11.236.1      4500  1:Module:8   1
itso_m7_p1    iSCSI         9.11.237.155 255.255.254.0  9.11.236.1      4500  1:Module:7   1
management    Management    9.11.237.109 255.255.254.0  9.11.236.1      1500  1:Module:4
management    Management    9.11.237.107 255.255.254.0  9.11.236.1      1500  1:Module:5
management    Management    9.11.237.108 255.255.254.0  9.11.236.1      1500  1:Module:6
VPN           VPN           0.0.0.0      255.0.0.0      0.0.0.0         1500  1:Module:4
VPN           VPN           0.0.0.0      255.0.0.0      0.0.0.0         1500  1:Module:6
```

Note that when you type this command, the rows might be displayed in a different order.

To see a complete list of IP interfaces, use the command **ipinterface_list_ports**. Example 6-6 shows an example of the result of running this command.

Example 6-6 XCLI to list iSCSI ports with ipinterface_list_ports command

```
>>ipinterface_list_ports
Index   Role                    IP Interface   Connected  Link Up?   Negotiated   Full Duplex?   Module
                                               Component             Speed (MB/s) Duplex
1       Management                                        yes        1000         yes            1:Module:4
1       Component                              1:UPS:1    yes        100          no             1:Module:4
1       Laptop                                            no         0            no             1:Module:4
1       VPN                                               no         0            no             1:Module:4
1       Management                                        yes        1000         yes            1:Module:5
1       Component                              1:UPS:2    yes        100          no             1:Module:5
1       Laptop                                            no         0            no             1:Module:5
1       Remote_Support_Module                             yes        1000         yes            1:Module:5
1       Management                                        yes        1000         yes            1:Module:6
1       Component                              1:UPS:3    yes        100          no             1:Module:6
1       VPN                                               no         0            no             1:Module:6
1       Remote_Support_Module                             yes        1000         yes            1:Module:6
1       iSCSI                                             unknown    N/A          unknown        1:Module:9
2       iSCSI                                             unknown    N/A          unknown        1:Module:9
1       iSCSI                  itso_m8_p1                 yes        1000         yes            1:Module:8
2       iSCSI                                             unknown    N/A          unknown        1:Module:8
1       iSCSI                  itso_m7_p1                 yes        1000         yes            1:Module:7
2       iSCSI                                             unknown    N/A          unknown        1:Module:7
```

6.3.7 iSCSI boot from SAN

It is not possible to boot from an iSCSI software initiator because it only becomes operational after the operating system is loaded.

6.4 Logical configuration for host connectivity

This section shows the tasks required to define a volume (LUN) and assign it to a host. The following sequence of steps is generic and intended to be operating system independent. The exact procedures for your server and operating system might differ somewhat:

1. Gather information on hosts and storage systems (WWPN and/or IQN).
2. Create SAN Zoning for the FC connections.
3. Create a Storage Pool.
4. Create a volume within the Storage Pool.
5. Define a host.
6. Add ports to the host (FC and/or iSCSI).
7. Map the volume to the host.
8. Check host connectivity at the XIV Storage System.
9. Complete and operating system-specific tasks.
10. If the server is going to SAN boot, the operating system will need installing.
11. Install mulitpath drivers if required. For information installing multi-path drivers, refer to the corresponding section in the host specific chapters of this book.
12. Reboot the host server or scan new disks.

Important: For the host system to effectively see and use the LUN, additional and operating system-specific configuration tasks are required. The tasks are described in subsequent chapters of this book according to the operating system of the host that is being configured.

6.4.1 Host configuration preparation

We use the environment shown in Figure 6-27 to illustrate the configuration tasks. In our example, we have two hosts: one host using FC connectivity and the other host using iSCSI. The diagram also shows the unique names of components, which are also used in the configuration steps.

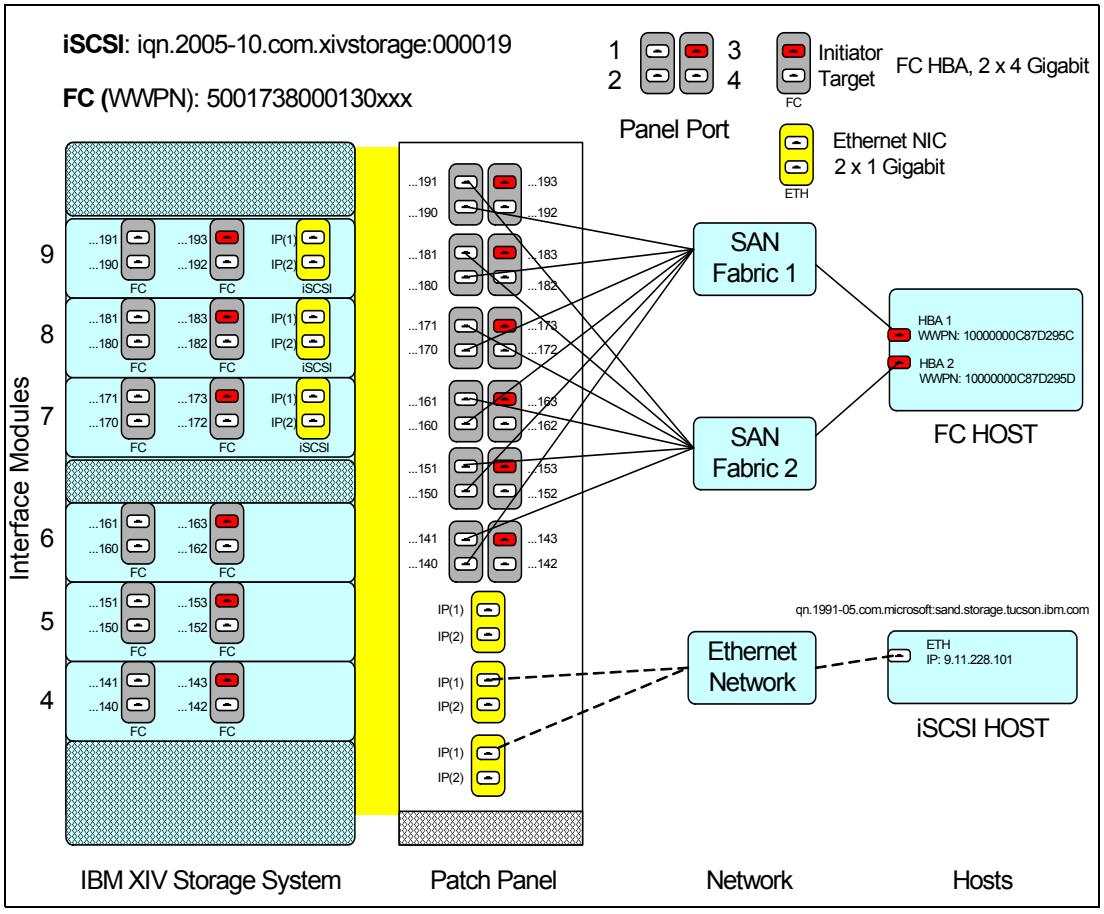


Figure 6-27 Example: Host connectivity overview of base setup

The following assumptions are made for the scenario shown in Figure 6-27:

- One host is set up with an FC connection; it has two HBAs and a multi-path driver installed.
- One host is set up with an iSCSI connection; it has one connection, it has the software initiator loaded and configured.

Hardware information

We recommend writing down the component names and IDs because this saves time during the implementation. An example is illustrated in Figure 6-2 for our particular scenario.

Table 6-2 Example: Required component information

Component	FC environment	iSCSI environment
IBM XIV FC HBAs	WWPN: 5001738000130*nnn* *nnn* for Fabric1: 140, 150, 160, 170, 180, and 190 *nnn* for Fabric2: 142, 152, 162, 172, 182, and 192	N/A
Host HBAs	HBA1 WWPN: 10000000C87D295C HBA2 WWPN: 10000000C87D295D	N/A
IBM XIV iSCSI IPs	N/A	Module7 Port1: 9.11.237.155 Module8 Port1: 9.11.237.156
IBM XIV iSCSI IQN (*do not change*)	N/A	iqn.2005-10.com.xivstorage:000019
Host IPs	N/A	9.11.228.101
Host iSCSI IQN	N/A	iqn.1991-05.com.microsoft:sand.storage.tucson.ibm.com
OS Type	Default	Default

Note: The OS Type is *default* for all hosts except HP-UX, in which case the type is *hpux*.

FC host specific tasks

It is preferable to first configure the SAN (Fabrics 1 and 2) and power on the host server, this will populate the XIV Storage System with a list of WWPNs from the host. This method is less prone to error when adding the ports in subsequent procedures.

For procedures showing how to configure zoning, refer to your FC switch manual. Here is an example of what the zoning details might look like for a typical server HBA zone. (Note that if using SVC as a host, there will be additional requirements, which are not discussed here.)

Fabric 1 HBA 1 zone

1. Log on to the Fabric 1 SAN switch and create a host zone:

```
zone: prime_sand_1
   prime_4_1; prime_5_3; prime_6_1;
   prime_7_3; sand_1
```

Fabric 2 HBA 2 zone

2. Log on to the Fabric 2 SAN switch and create a host zone:

```
zone: prime_sand_2
   prime_4_1; prime_5_3; prime_6_1;
   prime_7_3; sand_2
```

In the foregoing examples, aliases are used:

- sand is the name of the server, sand_1 is the name of HBA1, and sand_2 is the name of HBA2.
- prime_sand_1 is the zone name of fabric 1, and prime_sand_2 is the zone name of fabric 2.
- The other names are the aliases for the XIV patch panel ports.

iSCSI host specific tasks

For iSCSI connectivity, ensure that any configurations such as VLAN membership or port configuration are completed to allow the hosts and the XIV to communicate over IP.

6.4.2 Assigning LUNs to a host using the GUI

There are a number of steps required in order to a define new host and assign LUNs to it. Prerequisites are that volumes have been created in a Storage.

Defining a host

To define a host, follow these steps:

1. In the XIV Storage System main GUI window, move the mouse cursor over the **Hosts and Clusters** icon and select **Hosts and Clusters** (refer to Figure 6-28).

Figure 6-28 Hosts and Clusters menu

2. The Hosts window is displayed showing a list of hosts (if any) that are already defined. To add a new host or cluster, click either the **Add Host** or **Add Cluster** in the menu bar (refer to Figure 6-29). In our example, we select **Add Host**. The difference between the two is that *Add Host* is for a single host that will be assigned a LUN or multiple LUNs, whereas *Add Cluster* is for a group of hosts that will share a LUN or multiple LUNs.

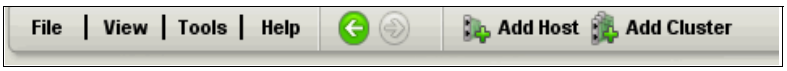


Figure 6-29 Add new host

3. The **Add Host** dialog is displayed as shown in Figure 6-30. Enter a name for the host. If a cluster definition was created in the previous step, it is available in the cluster drop-down list box. To add a server to a cluster, select a cluster name. Because we do not create a cluster in our example, we select **None**.

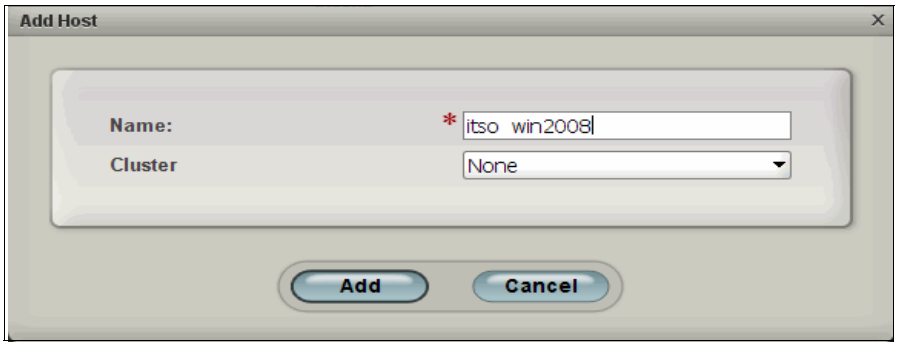


Figure 6-30 Add host details

4. Repeat steps 4 and 5 to create additional hosts. In our scenario we add another host called itso_win2008_iscsi
5. Host access to LUNs is granted depending on the host adapter ID. For an FC connection, the host adapter ID is the FC HBA WWPN, for an iSCSI connection, the host adapter ID is the host IQN. To add a WWPN or IQN to a host definition, right-click the host and select **Add Port** from the context menu (refer to Figure 6-31).

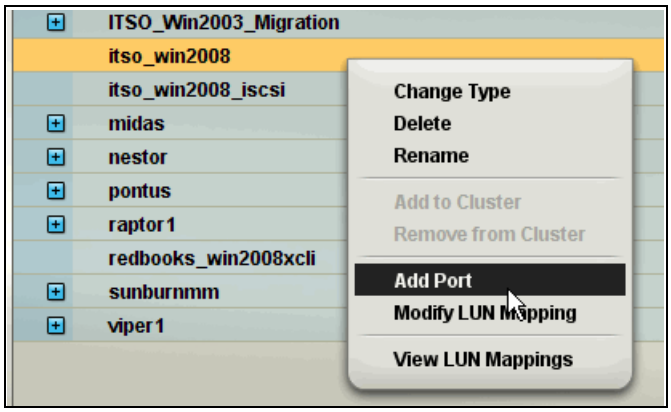


Figure 6-31 GUI example: Add port to host definition

6. The Add Port dialog is displayed as shown in Figure 6-32. Select port type FC or iSCSI.

   In this example, an FC host is defined. Add the WWPN for HBA1 as listed in Table 6-2 on page 211. If the host is correctly connected and has done a port login to the SAN switch at least once, the WWPN is shown in the drop-down list box. Otherwise, you can manually enter the WWPN. Adding ports from the drop-down list is less prone to error and is the recommended method. However, if hosts have not yet been connected to the SAN or zoned, then manually adding the WWPNs is the only option.

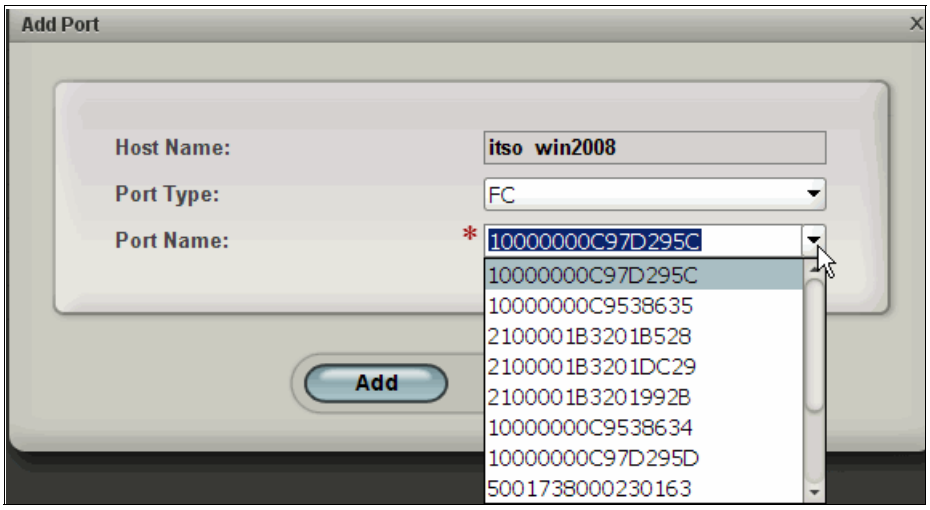


Figure 6-32 GUI example: Add FC port WWPN

Repeat steps 5 and 6 to add the second HBA WWPN; ports can be added in any order.

7. To add an iSCSI host, in the **Add Port** dialog, specify the port type as iSCSI and enter the IQN of the HBA as the iSCSI Name. Refer to Figure 6-33.

Figure 6-33 GUI example: Add iSCSI port

8. The host will appear with its ports in the Hosts dialog box as shown in Figure 6-34.

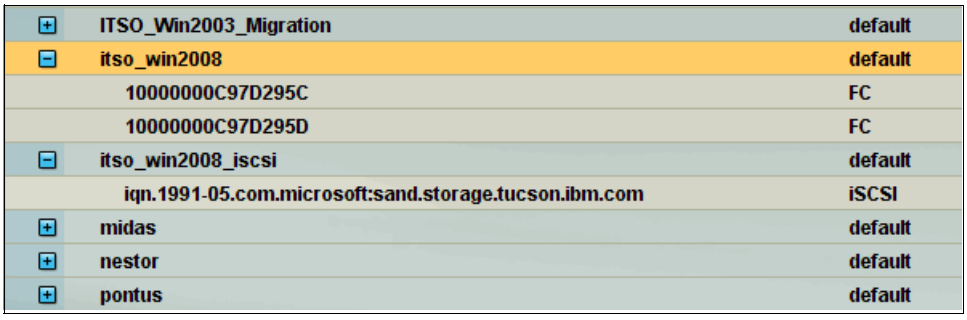


Figure 6-34 List of hosts and ports

In this example, the hosts *itso_win2008* and *itso_win2008_iscsi* are in fact the same physical host, however, they have been entered as separate entities so that when mapping LUNs, the FC and iSCSI protocols do not access the same LUNs.

Mapping LUNs to a host

The final configuration step is to map LUNs to the host. To do this, follow these steps:

1. While still in the **Hosts and Clusters** configuration pane, right-click the host to which the volume is to be mapped and select **Modify LUN Mappings** from the context menu (refer to Figure 6-35).

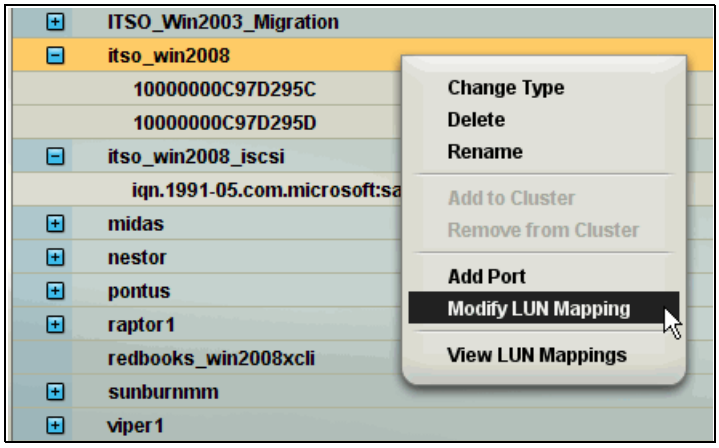


Figure 6-35 Map LUN to host

2. The **Volume to LUN Mapping** window opens as shown in Figure 6-36.

- Select an available volume from the left pane.
- The GUI will suggest a LUN ID to which to map the volume, however, this can be changed to meet your requirements.
- Click **Map** and the volume is assigned immediately.

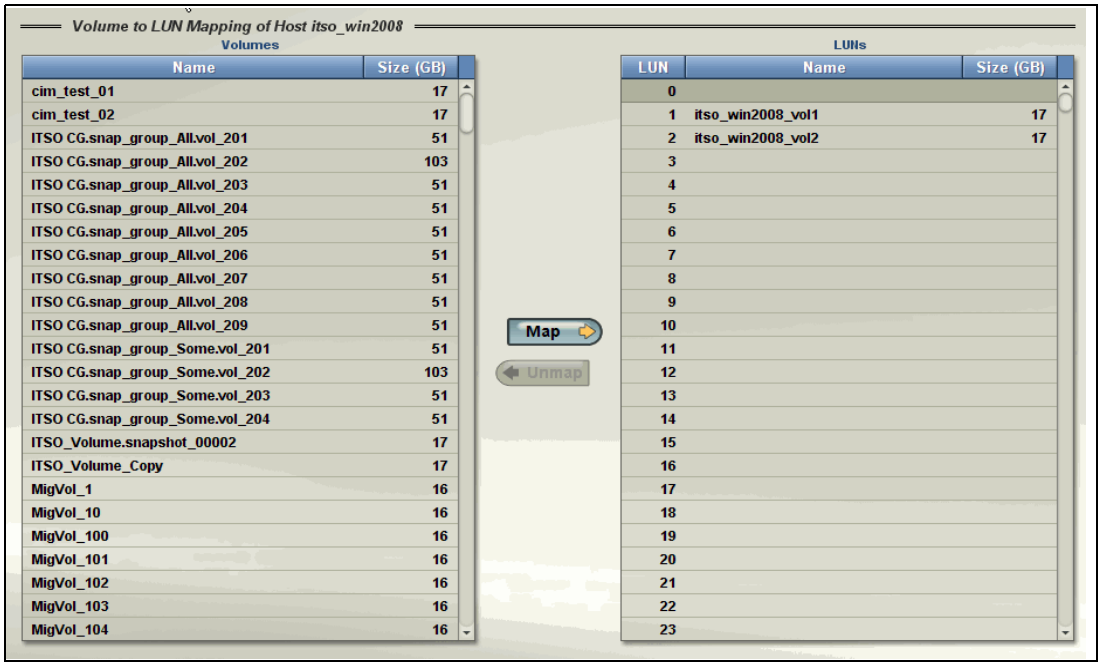


Figure 6-36 Map FC volume to FC host

There is no difference in mapping a volume to an FC or iSCSI host in the XIV GUI **Volume to LUN Mapping** view.

3. To complete this example, power up the host server and check connectivity. The XIV Storage System has a real-time connectivity status overview. Select **Hosts Connectivity** from the **Hosts and Clusters** menu to access the connectivity status. See Figure 6-37.

.

Figure 6-37 Hosts Connectivity

4. The host connectivity window is displayed. In our example, the `ExampleFChost` was expected to have dual path connectivity to every module. However, only two modules (5 and 6) show as connected (refer to Figure 6-38), and the iSCSI host has no connection to module 9.

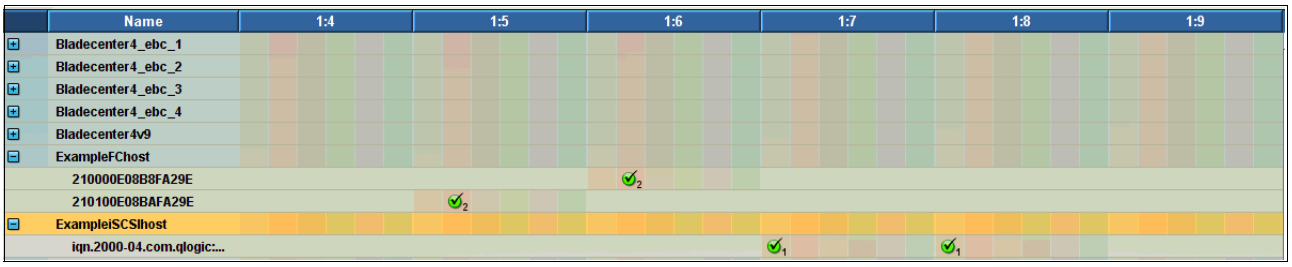


Figure 6-38 GUI example: Host connectivity matrix

5. The setup of the new FC and/or iSCSI hosts on the XIV Storage System is complete.

   At this stage there might be operating system dependent steps that need to be performed, these are described in the operating system chapters.

6.4.3 Assigning LUNs to a host using the XCLI

There are a number of steps required in order to define a new host and assign LUNs to it. Prerequisites are that volumes have been created in a Storage Pool.

Defining a new host

Follow these steps to use the XCLI to prepare for a new host:

1. Create a host definition for your FC and iSCSI hosts, using the **`host_define`** command. Refer to Example 6-7.

   Example 6-7 XCLI example: Create host definition

```
>>host_define host=itso_win2008
Command executed successfully.

>>host_define host=itso_win2008_iscsi
Command executed successfully.
```

2. Host access to LUNs is granted depending on the host adapter ID. For an FC connection, the host adapter ID is the FC HBA WWPN. For an iSCSI connection, the host adapter ID is the IQN of the host.

   In Example 6-8, the WWPN of the FC host for HBA1 and HBA2 is added with the **host_add_port** command and by specifying an **fcaddress**.

   Example 6-8 Create FC port and add to host definition

```
>> host_add_port host=itso_win2008 fcaddress=10000000c97d295c
Command executed successfully.

>> host_add_port host=itso_win2008 fcaddress=10000000c97d295d
Command executed successfully.
```

   In Example 6-9, the IQN of the iSCSI host is added. Note this is the same **host_add_por**t command, but with the **iscsi_name** parameter.

   Example 6-9 Create iSCSI port and add to the host definition

```
>> host_add_port host=itso_win2008_iscsi
iscsi_name=iqn.1991-05.com.microsoft:sand.storage.tucson.ibm.com
Command executed successfully
```

Mapping LUNs to a host

To map the LUNs, follow these steps:

1. The final configuration step is to map LUNs to the host definition. Note that for a cluster, the volumes are mapped to the cluster host definition. There is no difference for FC or iSCSI mapping to a host. Both commands are shown in Example 6-10.

   Example 6-10 XCLI example: Map volumes to hosts

```
>> map_vol host=itso_win2008 vol=itso_win2008_vol1 lun=1
Command executed successfully.

>> map_vol host=itso_win2008 vol=itso_win2008_vol2 lun=2
Command executed successfully.

>> map_vol host=itso_win2008_iscsi vol=itso_win2008_vol3 lun=1
Command executed successfully.
```

2. To complete the example, power up the server and check the host connectivity status from the XIV Storage System point of view. Example 6-11 shows the output for both hosts.

   Example 6-11 XCLI example: Check host connectivity

```
>> host_connectivity_list host=itso_win2008
Host           Host Port          Module       Local FC port   Type
itso_win2008   10000000C97D295C   1:Module:6   1:FC_Port:6:1   FC
itso_win2008   10000000C97D295C   1:Module:4   1:FC_Port:4:1   FC
itso_win2008   10000000C97D295D   1:Module:5   1:FC_Port:5:3   FC
itso_win2008   10000000C97D295D   1:Module:7   1:FC_Port:7:3   FC

>> host_connectivity_list host=itso_win2008_iscsi
Host               Host Port                                              Module       Local FC port   Type
itso_win2008_iscsi iqn.1991-05.com.microsoft:sand.storage.tucson.ibm.com  1:Module:8                   iSCSI
itso_win2008_iscsi iqn.1991-05.com.microsoft:sand.storage.tucson.ibm.com  1:Module:7                   iSCSI
```

In Example 6-11 on page 217, there are two paths per host FC HBA and two paths for the single Ethernet port that was configured.

3. The setup of the new FC and/or iSCSI hosts on the XIV Storage System is now complete.

   At this stage there might be operating system dependent steps that need to be performed, these steps are described in the operating system chapters.

6.4.4 HBA queue depth

The HBA queue depth is a performance tuning parameter and refers to the amount of data that can be “in-flight” on the HBA. Most HBAs will default to around 32 and typically range from 1-254. The optimum figure is normally between 16-64 but will depend on the operating system, driver, application, and storage system that the server is attached to. Refer to your HBA and OS documentation for guidance. However, you might also need to run tests to determine the correct figure.

Each XIV port can handle a queue depth of 1400, however, this does not mean that the server HBA should automatically be increased based on the number of hosts per port, because a higher than required figured can have a negative impact.

Different HBAs and operating systems will have their own procedures for configuring queue depth; refer to your documentation for more information. Figure 6-39 shows an example of the Emulex HBAnyware® utility used on a Windows server to change queue depth.

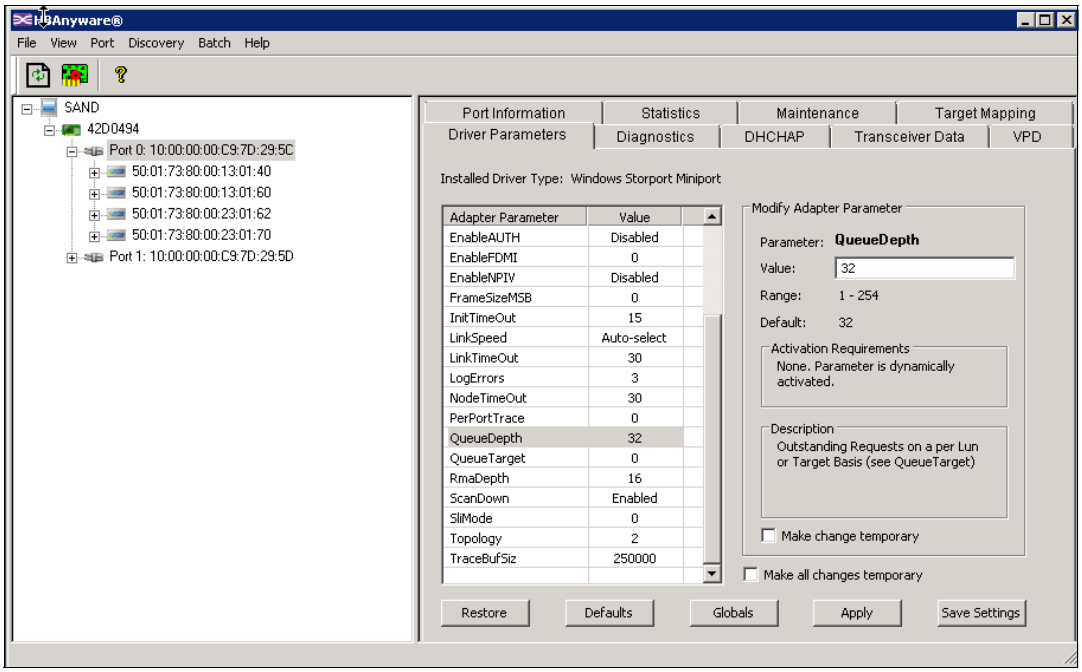


Figure 6-39 Emulex queue depth

6.4.5 Troubleshooting

Troubleshooting connectivity problems can be difficult. However, the XIV Storage System does have some in-built tools to assist with this. Table 6-3 contains a list of some of the built-in tools. For further information, refer to the XCLI manual, which can be downloaded from the XIV Information Center at:

http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp

Table 6-3 XIV in-built tools

Tool	Description
fc_connectivity_list	Discovers FC hosts and targets on the FC network
fc_port_list	Lists all FC ports, their configuration, and their status
ipinterface_list_ports	Lists all Ethernet ports, their configuration, and their status
ipinterface_run_arp	Prints the ARP database of a specified IP address
ipinterface_run_traceroute	Tests connectivity to a remote IP address
host_connectivity_list	Lists FC and iSCSI connectivity to hosts

7

Windows Server 2008 host connectivity

This chapter explains specific considerations for attaching the XIV to a Microsoft Windows Server 2008 host.

7.1 Attaching a Microsoft Windows 2008 host to XIV

This section discusses specific instructions for Fibre Channel (FC) and Internet Small Computer System Interface (iSCSI) connections. All the information here relates to Windows Server 2008 (and not other versions of Windows) unless otherwise specified.

The procedures and instructions given here are based on code that was available at the time of writing this book. For the latest information about XIV OS support, refer to the System Storage Interoperability Center (SSIC) at:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

Also, refer to the XIV Storage System *Host System Attachment Guide for Windows - Installation Guide*, which is available at:

http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp

Prerequisites

To successfully attach a Windows host to XIV and access storage, a number of prerequisites need to be met. Here is a generic list. However, your environment might have additional requirements:

- Complete the cabling.
- Complete the zoning.
- Install Service Pack 1 or later.
- Install any other updates if required.
- Install hot fix KB958912.
- Install hot fix KB932755 if required.
- Refer to KB957316 if booting from SAN.
- Create volumes to be assigned to the host.

Supported versions of Windows

At the time of writing, the following versions of Windows (including cluster configurations) are supported:

- Windows Server 2008 SP1 and above (x86, x64)
- Windows Server 2003 SP1 and above (x86, x64)
- Windows 2000 Server SP4 (x86) available via RPQ

Supported FC HBAs

Supported FC HBAs are available from IBM, Emulex and QLogic. Further details on driver versions are available from SSIC at the following Web site:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

Unless otherwise noted in SSIC, use any supported driver and firmware by the HBA vendors (the latest versions are always preferred). For HBAs in Sun systems, use Sun branded HBAs and Sun ready HBAs only.

Multi-path support

Microsoft provides a multi-path framework and development kit called the Microsoft Multi-path I/O (MPIO). The driver development kit allows storage vendors to create Device Specific Modules (DSMs) for MPIO and to build interoperable multi-path solutions that integrate tightly with the Microsoft Windows family of products.

MPIO allows the host HBAs to establish multiple sessions with the same target LUN but present it to Windows as a single LUN. The Windows MPIO drivers enables a true active/active path policy allowing I/O over multiple paths simultaneously.

Further information about Microsoft MPIO support is available at the following Web site:

http://download.microsoft.com/download/3/0/4/304083f1-11e7-44d9-92b9-2f3cdbf01048/mpio.doc

Boot from SAN: Support

SAN boot is supported (over FC only) in the following configurations:

- Windows 2008 with MSDSM
- Windows 2003 with XIVDSM

7.1.1 Windows host FC configuration

This section describes attaching to XIV over Fibre Channel and provides detailed descriptions and installation instructions for the various software components required.

Installing HBA drivers

Windows 2008 includes drivers for many HBAs, however, it is likely that they will not be the latest version for your HBA. You should install the latest available driver that is supported. HBAs drivers are available from IBM, Emulex and QLogic Web sites. They will come with instructions that should be followed to complete the installation.

With Windows operating systems, the queue depth settings are specified as part of the host adapter's configuration through the BIOS settings or using a specific software provided by the HBA vendor.

The XIV Storage System can handle a queue depth of 1400 per FC host port and 256 per volume.

Optimize your environment by trying to evenly spread the I/O load across all available ports, taking into account the load on a particular server, its queue depth, and the number of volumes.

Installing Multi-Path I/O (MPIO) feature

MPIO is provided as an in-built feature of Windows 2008. Follow these steps to install it:

1. Using *Server Manager*, select **Features Summary**, then right-click and select **Add Features**. In the *Select Feature* page, select **Multi-Path I/O**. See Figure 7-1.
2. Follow the instructions on the panel to complete the installation. This might require a reboot.

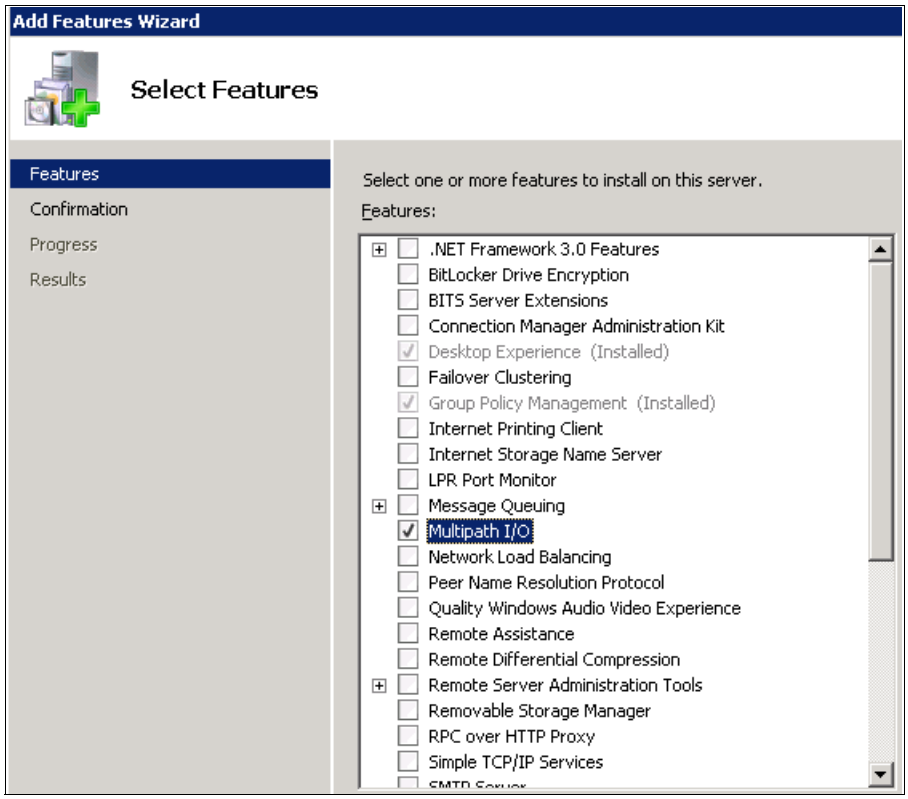


Figure 7-1 Selecting the Multipath I/O feature

3. To check that the driver has been installed correctly, load **Device Manager** and verify that it now includes **Microsoft Multi-Path Bus Driver** as illustrated in Figure 7-2.

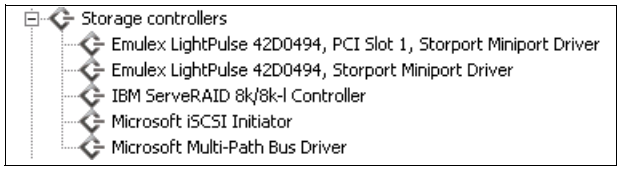


Figure 7-2 Microsoft Multi-Path Bus Driver

Windows Host Attachment Kit installation

The Windows 2008 Host Attachment Kit must be installed to gain access to XIV storage. Note that there are different versions of the Host Attachment Kit for different versions of Windows, and this is further sub-divided into 32-bit and 64-bit versions. The Host Attachment Kit can be downloaded from the following Web site:

`http://www.ibm.com/support/search.wss?q=ssg1*&tc=STJTAG+HW3E0&rs=1319&dc=D400&dtm`

The following instructions are based on the installation performed at the time of writing. You should also refer to the instructions in the *Windows Host Attachment Guide* because these instructions are subject to change over time. The instructions included here show the GUI installation; for command line instructions, refer to the *Windows Host Attachment Guide*.

Before installing the Host Attachment Kit, any other multipathing software that was eventually previously installed must be removed. Failure to do so can lead to unpredictable behavior or even loss of data.

First you need to install the Python engine, which is now used in all of the XIV HAKs and is a mandatory installation:

1. Run the **XPyV.msi** file. The *Welcome* panel shown in Figure 7-3 is displayed. Follow the instructions on the panel to complete the installation. This might require a reboot when finished.

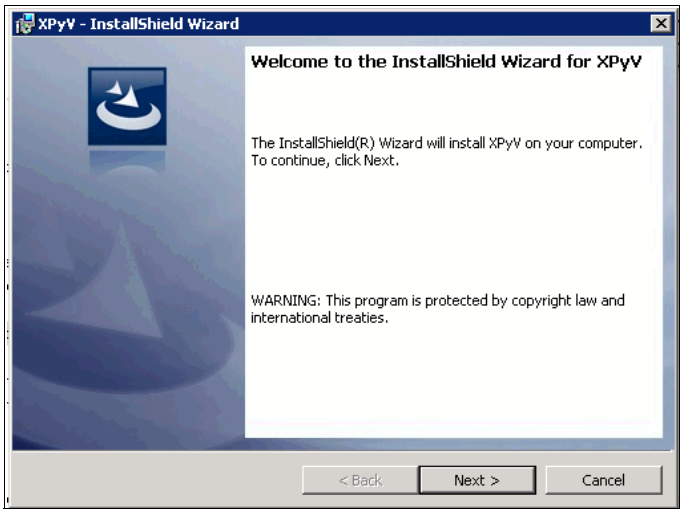


Figure 7-3 XPyV welcome panel

2. When the XPyV installation has completed, run the installation file for your own version of Windows. In our installation, it was *XIV_host_attachment_windows-1.1-x64_SLT.msi.* Figure 7-4 and Figure 7-5 show the start of the installation. Simply follow the instructions on the panel to complete the installation.

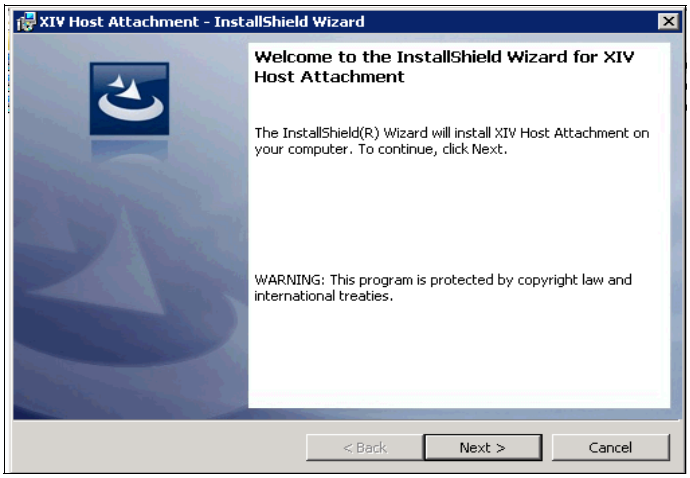


Figure 7-4 Host Attachment Welcome panel

Figure 7-5 Setup Type

3. The Windows Security dialog shown in Figure 7-6 might be displayed. If this is the case, select **Install**, and follow the instructions to complete the installation.

Figure 7-6 Windows Security dialog

4. When the installation has finished, a reboot will be required.

At this point your Windows host should have all the required software to successfully attach to the XIV Storage System.

Scanning for new LUNs

Before you can scan for new LUNs, your host needs to be created, configured, and have LUNs assigned. See Chapter 6., “Host connectivity” on page 183 for information on how to do this. The following instructions assume that these operations have been completed.

1. Go to **Server Manager** → **Device Manager** → select **Action** → **Scan for hardware changes**. In the Device Manger tree under **Disk Drives**, your XIV LUNs will appear as shown in Figure 7-7.

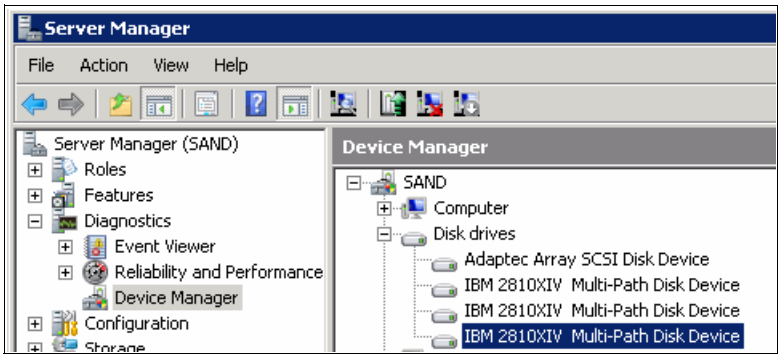


Figure 7-7 Multi-Path disk devices in Device Manager

The number of objects named **IBM 2810XIV SCSI Disk Device** will depend on the number of LUNs mapped to the host.

2. Right-clicking on one of the I**BM 2810XIV SCSI Device** object and selecting **Properties** and then the **MPIO** tab will allow the load balancing to be changed as shown in Figure 7-8:

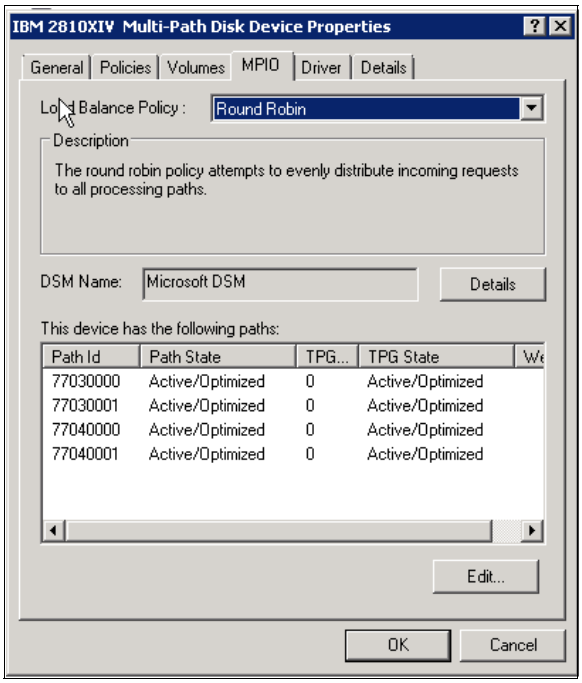


Figure 7-8 MPIO load balancing

The default setting here should be **Round Robin**. Change this only if you are confident that another option is better suited to your environment.

The possible options are:

- Fail Over Only
- Round Robin (default)
- Round Robin With Subset
- Least Queue Depth
- Weighted Paths

3. The mapped LUNs on the host can be seen in **Disk Management** as illustrated in Figure 7-9.

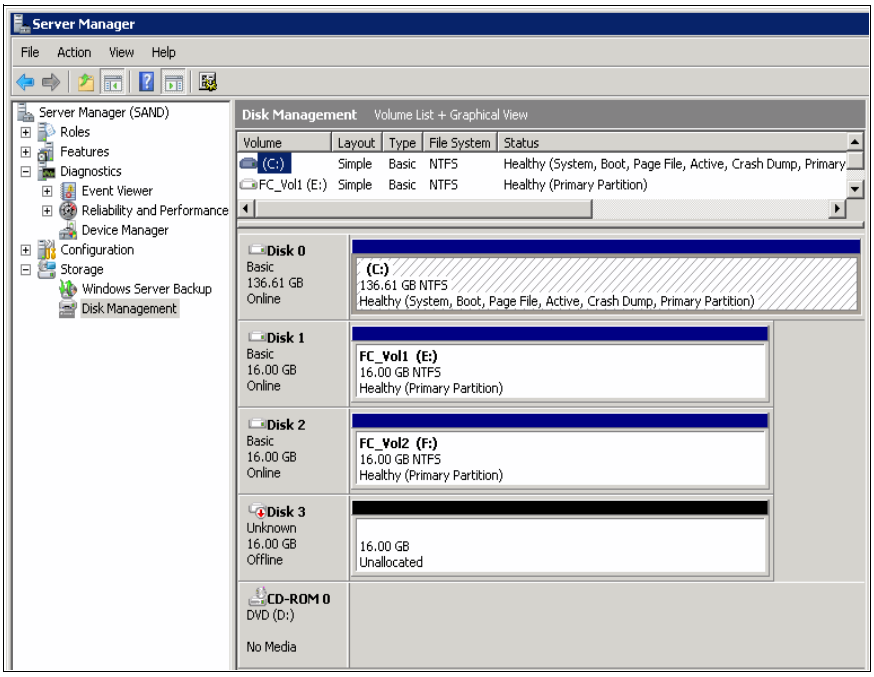


Figure 7-9 Mapped LUNs appear in Disk Management

7.1.2 Host Attachment Kit utilities

The Host Attachment Kit (HAK) now includes the following utilities:

- xiv_devlist
- xiv_diag
- wfetch

xiv_devlist

This utility requires Administrator privileges. The utility lists the XIV volumes available to the host; non-XIV volumes are also listed separately. To run it, go to a command prompt and enter **`xiv_devlist`**, as shown in Example 7-1.

Example 7-1 xiv_devlist

```
C:\Users\Administrator.SAND>xiv_devlist
executing: xpyv.exe "C:\Program Files\XIV\host_attach\lib\python\xiv_devlist\xiv
_devlist.py"

XIV devices
===========
Device               Vol Name     XIV Host     Size    Paths XIV ID   Vol ID
-------------------------------------------------------------------------------
PHYSICALDRIVE3       itso_...vol3 itso_win2008 17.2GB  4/4   MN00013  2746
PHYSICALDRIVE2       itso_...vol1 itso_win2008 17.2GB  4/4   MN00013  194
PHYSICALDRIVE1       itso_...vol2 itso_win2008 17.2GB  4/4   MN00013  195

Non-XIV devices
===============
Device               Size    Paths
-----------------------------------
PHYSICALDRIVE0       146.7GB 1/1
```

xiv_diag

This requires Administrator privileges. The utility gathers diagnostic information from the operating system. The resulting zip file can then be sent to IBM-XIV support teams for review and analysis. To run, go to a command prompt and enter `xiv_diag`, as shown in Example 7-2.

Example 7-2 xiv_diag

```
C:\Users\Administrator.SAND>xiv_diag
executing: xpyv.exe "C:\Program Files\XIV\host_attach\lib\python\xiv_diag\xiv_diag.py"
Please type in a directory in which to place the xiv_diag file [default: C:\Windows\Temp]:
Creating xiv_diag zip file C:\Windows\Temp\xiv_diag-results_2009-7-1_15-38-53.zip

INFO: Copying Memory dumps to temporary directory... DONE
INFO: Gathering System Information (1/2)... DONE
INFO: Gathering System Information (2/2)... DONE
INFO: Gathering System Event Log... DONE
INFO: Gathering Application Event Log... DONE
INFO: Gathering Cluster (2003) Log... SKIPPED
INFO: Gathering Cluster (2008) Log Generator... SKIPPED
INFO: Gathering Cluster (2008) Logs (1/5)... SKIPPED
INFO: Gathering Cluster (2008) Logs (2/5)...
.
.
.
INFO: Gathering Windows Memory Dump... SKIPPED
INFO: Gathering Windows Setup API (1/2)... DONE
INFO: Gathering Windows Setup API (2/2)... DONE
INFO: Gathering Hardware Registry Subtree... DONE
INFO: Gathering xiv_devlist... SKIPPED

Deleting temporary directory...
DONE
INFO: Gathering is now complete.
INFO: You can now send C:\Windows\Temp\xiv_diag-results_2009-7-1_15-38-53.zip to IBM-XIV
for review.
INFO: Exiting.
```

wfetch

This is a simple CLI utility for downloading files from HTTP, HTTPS and FTP sites. It runs on most UNIX, Linux, and Windows operating systems.

7.1.3 Installation for Windows 2003

The installation for Windows 2003 follows a set of procedures similar to that of Windows 2008 with the exception that Windows 2003 does not have native MPIO support.

MPIO support for Windows 2003 is installed as part of the Host Attachment Kit.

Review the prerequisites and requirements outlined in the XIV Host Attachment Kit.

7.2 Attaching a Microsoft Windows 2003 Cluster to XIV

This section discusses the attachment of Microsoft Windows 2003 cluster nodes to the XIV Storage System.

The procedures and instructions given here are based on code that was available at the time of writing this book. For the latest information about XIV Storage Management software compatibility, refer to the System Storage Interoperability Center (SSIC) at:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

Also, refer to the XIV Storage System *Host System Attachment Guide for Windows - Installation Guide*, which is available at:

http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp

This section only focuses on the implementation of a two node Windows 2003 Cluster using FC connectivity and assumes that all of the following prerequisites have been completed.

7.2.1 Prerequisites

To successfully attach a Windows cluster node to XIV and access storage, a number of prerequisites need to be met. Here is a generic list; however, your environment might have additional requirements:

- Complete the cabling.
- Configure the zoning.
- Install Windows Service Pack 2 or later.
- Install any other updates if required.
- Install hot fix KB932755 if required.
- Install the Host Attachment Kit.
- Ensure that all nodes are part of the same domain.
- Create volumes to be assigned to the nodes.

Supported versions of Windows Cluster Server

At the time of writing, the following versions of Windows Cluster Server are supported:

- Windows Server 2008
- Windows Server 2003 SP2

Supported configurations of Windows Cluster Server

Windows Cluster Server is supported in the following configurations:

- 2 node: All versions
- 4 node: Windows 2003 x64
- 4 node: Windows 2008 x86

If other configurations are required, you will need a Request for Price Quote (RPQ). This is a process by which IBM will test a specific customer configuration to determine if it can be certified and supported. Contact your IBM representative for more information.

Supported FC HBAs

Supported FC HBAs are available from IBM, Emulex and QLogic. Further details on driver versions are available from SSIC at the following Web site:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

Unless otherwise noted in SSIC, use any supported driver and firmware by the HBA vendors (the latest versions are always preferred). For HBAs in Sun systems, use Sun branded HBAs and Sun ready HBAs only.

Multi-path support

Microsoft provides a multi-path framework and development kit called the Microsoft Multi-path I/O (MPIO). The driver development kit allows storage vendors to create Device Specific Modules (DSM) for MPIO and to build interoperable multi-path solutions that integrate tightly with the Microsoft Windows family of products.

MPIO allows the host HBAs to establish multiple sessions with the same target LUN but present it to Windows as a single LUN. The Windows MPIO drivers enable a true active/active path policy allowing I/O over multiple paths simultaneously.

MPIO support for Windows 2003 is installed as part of the Windows Host Attachment Kit.

Further information on Microsoft MPIO support is available at the following Web site:

http://download.microsoft.com/download/3/0/4/304083f1-11e7-44d9-92b9-2f3cdbf01048/mpio.doc

7.2.2 Installing Cluster Services

In our scenario described next, we install a two node Windows 2003 Cluster. Our procedures assume that you are familiar with Windows 2003 Cluster and focus on specific requirements for attaching to XIV. For further details about installing a Windows 2003 Cluster, refer to the following Web site:

http://www.microsoft.com/downloads/details.aspx?familyid=96F76ED7-9634-4300-9159-89638F4B4EF7&displaylang=en

To install the cluster, follow these steps:

1. Set up a cluster specific configuration. This includes:
   - Public network connectivity
   - Private (Heartbeat) network connectivity
   - Cluster Service account
2. Before continuing, ensure that at all times, only one node can access the shared disks until the cluster service has been installed on the first node. To do this, turn off all nodes except the first one (Node 1) that will be installed.
3. On the XIV system, select the **Hosts and Clusters** menu, then select the **Hosts and Clusters** menu item. Create a cluster and put both nodes into the cluster as depicted in Figure 7-10.

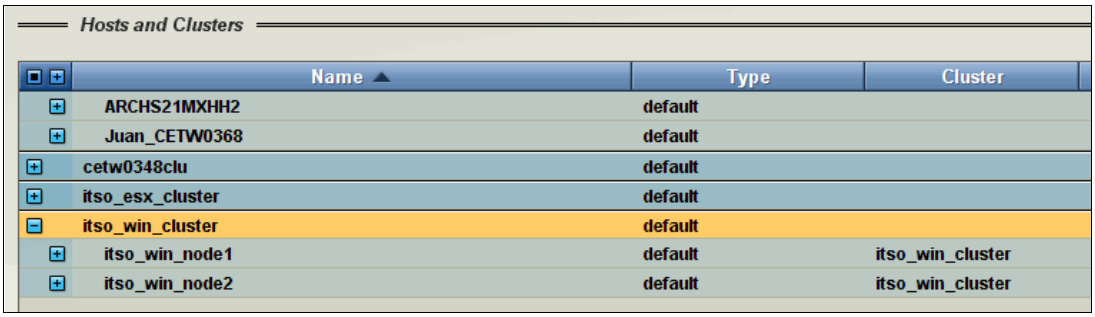


Figure 7-10 XIV cluster with Node 1

You can see that an XIV cluster named *itso_win_cluster* has been created and both nodes have been put in. Node2 must be turned off.

4. Map the quorum and data LUNs to the cluster as shown in Figure 7-11.

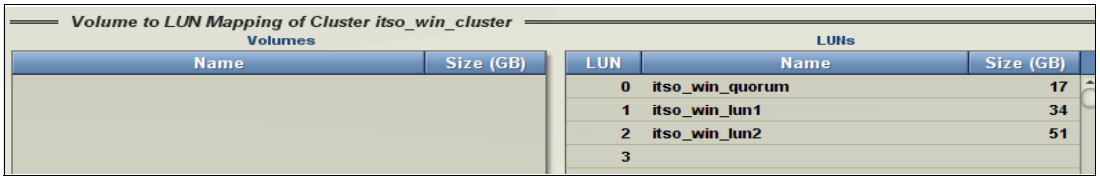


Figure 7-11 Mapped LUNs

You can see here that three LUNs have been mapped to the XIV cluster (and not to the individual nodes).

5. On Node1, scan for new disks, then initialize, partition, and format them with NTFS.

   Microsoft has some best practices for drive letter usage and drive naming. For more information, refer to the following document.

   http://support.microsoft.com/?id=318534

   For our scenario, we use the following values:

   – Quorum drive letter = Q
   – Quorum drive name = DriveQ
   – Data drive 1 letter = R
   – Data drive 1 name = DriveR
   – Data drive 2 letter = S
   – Data drive 2 name = DriveS

   The following requirements are for shared cluster disks:

   – These disks must be basic disks.
   – For 64-bit versions of Windows 2003, they must be MBR disks.

   Refer to Figure 7-12 for what this would look like on Node1.

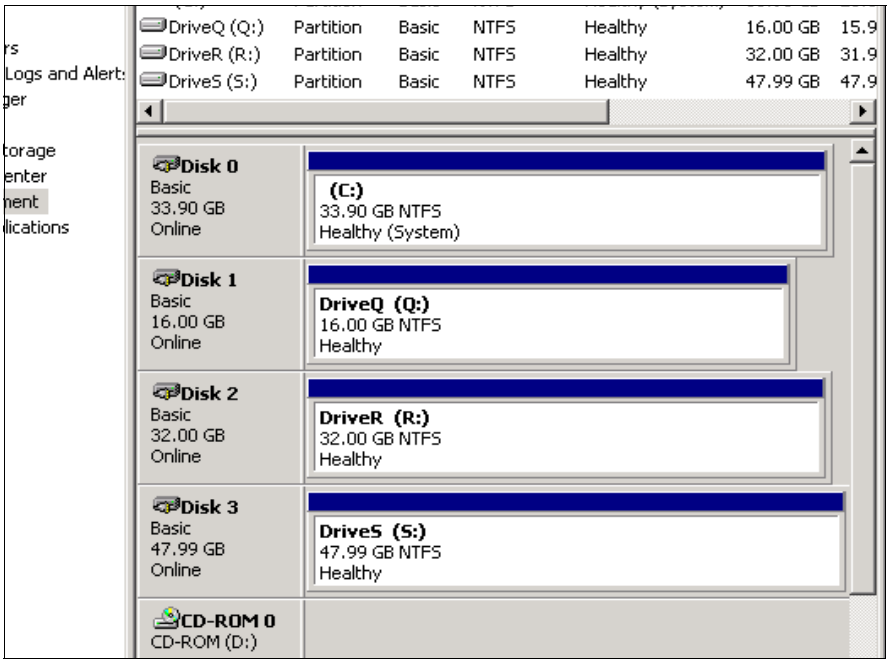


Figure 7-12 Initialized, partitioned and formatted disks

6. Check access to at least one of the shared drives by creating a document. For example, create a text file on one of them, and then turn Node1 off.
7. Turn on Node2 and scan for new disks. All the disks should appear, in our case, three disks. They will already be initialized and partitioned. However, they might need formatting again. You will still have to set drive letters and drive names, and these must identical to those set in step 4.
8. Check access to at least one of the shared drives by creating a document. For example, create a text file on one of them, then turn Node2 off.
9. Turn Node1 back on, launch Cluster Administrator, and create a new cluster. Refer to documentation from Microsoft if necessary for help with this task.
10. After the cluster service is installed on Node1, turn on Node2. Launch **Cluster Administrator** on Node2 and install Node2 into the cluster.
11. Change the boot delay time on the nodes so that Node2 boots one minute after Node1. If you have more nodes, then continue this pattern; for instance, Node3 boots one minute after Node2, and so on. The reason for this is that if all the nodes boot at once and try to attach to the quorum resource, the cluster service might fail to initialize.
12. At this stage the configuration is complete with regard to the cluster attaching to the XIV system, however, there might be some post-installation tasks to complete. Refer to the Microsoft documentation for more information. Figure 7-13 shows resources split between the two nodes.

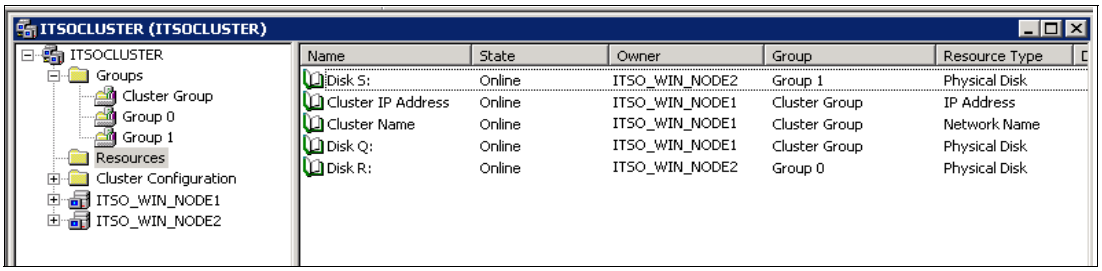


Figure 7-13 Cluster resources shared between nodes

8

AIX host connectivity

This chapter explains specific considerations for host connectivity and describes the host attachment-related tasks for the AIX operating system platform.

8.1 Attaching AIX hosts to XIV

This section provides information and procedures for attaching the XIV Storage System to AIX on an IBM POWER® platform. The Fibre Channel connectivity is discussed first, then iSCSI attachment.

Interoperability

The XIV Storage System supports different versions of the AIX operating system, either via Fibre Channel (FC) or iSCSI connectivity.

Details about supported versions of AIX and associated hardware and software requirements can found in the System Storage Interoperation Center (SSIC) at:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

Prerequisites

If the current AIX operating system level installed on your system is not a level that is compatible with XIV, you must upgrade prior to attaching the XIV storage. To determine the maintenance package or technology level currently installed on your system, use the **oslevel** command as shown in Example 8-1.

Example 8-1 AIX: Determine current AIX version and maintenance level

```
# oslevel -s
5300-10-01-0921
```

In our example, the system is running AIX 5.3.0.0 technology level 10 (53TL10). Use this information in conjunction with the SSIC to ensure that the attachment will be a supported IBM configuration.

In the event that AIX maintenance items are needed, consult the IBM Fix Central Web site to download fixes and updates for your systems software, hardware, and operating system at:

http://www.ibm.com/eserver/support/fixes/fixcentral/main/pseries/aix

Before further configuring your host system or the XIV Storage System, make sure that the physical connectivity between the XIV and the POWER system is properly established. In addition to proper cabling, if using FC switched connections, you must ensure that you have a correct zoning (using the WWPN numbers of the AIX host). Refer to 6.2, “Fibre Channel (FC) connectivity” on page 190 for the recommended cabling and zoning setup.

8.1.1 AIX host FC configuration

Attaching the XIV Storage System to an AIX host using Fibre Channel involves the following activities from the host side:

- Identify the Fibre Channel host bus adapters (HBAs) and determine their WWPN values.
- Install XIV-specific AIX Host Attachment Kit.
- Configure multipathing.

Identifying FC adapters and attributes

In order to allocate XIV volumes to an AIX host, the first step is to identify the Fibre Channel adapters on the AIX server. Use the **lsdev** command to list all the FC adapter ports in your system, as shown in Example 8-2.

Example 8-2 AIX: Listing FC adapters

```
# lsdev -Cc adapter | grep fcs
fcs5    Available 1n-08    FC Adapter
fcs6    Available 1n-09    FC Adapter
```

This example shows that, in our case, we have two FC ports.

Another useful command that is shown in Example 8-3 returns not just the ports, but also where the Fibre Channel adapters reside in the system (in which PCI slot). This command can be used to physically identify in what slot a specific adapter is placed.

Example 8-3 AIX: Locating FC adapters

```
# lsslot -c pci | grep fcs
U0.1-P2-I6  PCI-X capable, 64 bit, 133MHz slot  fcs5 fcs6
```

To obtain the Worldwide Port Name (WWPN) of each of the POWER system FC adapters, you can use the **lscfg** command, as shown in Example 8-4.

Example 8-4 AIX: Finding Fibre Channel adapter WWN

```
# lscfg -vl fcs0
fcs5             U0.1-P2-I6/Q1  FC Adapter

        Part Number.................03N5029
        EC Level....................A
        Serial Number...............1F5510C069
        Manufacturer................001F
        Customer Card ID Number.....5759
        FRU Number.................. 03N5029
        Device Specific.(ZM)........3
        Network Address.............10000000C9509F8A
        ROS Level and ID............02C82114
        Device Specific.(Z0)........1036406D
        Device Specific.(Z1)........00000000
        Device Specific.(Z2)........00000000
        Device Specific.(Z3)........03000909
        Device Specific.(Z4)........FFC01154
        Device Specific.(Z5)........02C82114
        Device Specific.(Z6)........06C12114
        Device Specific.(Z7)........07C12114
        Device Specific.(Z8)........20000000C9509F8A
        Device Specific.(Z9)........BS2.10A4
        Device Specific.(ZA)........B1F2.10A4
        Device Specific.(ZB)........B2F2.10A4
        Device Specific.(ZC)........00000000
        Hardware Location Code......U0.1-P2-I6/Q1
```

You can also print the WWPN of an HBA directly by issuing this command:

`lscfg -vl <fcs#> | grep Network`

Note: In the foregoing command, *<fcs#>* stands for an instance of a FC HBA to query.

At this point, you can define the AIX host system on the XIV Storage System and assign FC ports for the WWPNs. If the FC connection was correctly done, the zoning enabled, and the FC adapters are in an available state on the host, these ports will be selectable from the drop-down list in the Add Port window of the XIV Graphical User Interface.

For the detailed description of host definition and volume mapping, refer to 4.5, “Host definition and mappings” on page 118.

Installing the XIV-specific package for AIX

For AIX to recognize the disks mapped from the XIV Storage System as *IBM 2810XIV Fibre Channel Disk*, a specific fileset known as the *XIV Host Attachment Package for AIX* is required on the AIX system. This fileset will also enable multipathing. The fileset can be downloaded from:

http://www.ibm.com/support/search.wss?q=ssg1*&tc=STJTAG+HW3E0&rs=1319&dc=D400&dtm

Important: Use this package for a clean installation or to upgrade from previous versions. IBM supports a connection by AIX hosts to the IBM XIV Storage System only when this package is installed. Installing previous versions might render your server unbootable.

To install the fileset, follow these steps:

1. Download or copy the downloaded fileset to your AIX system.
2. From the AIX prompt, change to the directory where your XIV package is located and execute the **`inutoc .`** command to create the table of contents file.
3. Use the AIX **`installp`** command or SMITTY (**smitty** → **Software Installation and Maintenance** → **Install and Update Software** → **Install Software**) to install the XIV disk package. Complete the parameters as shown in Example 8-5 if installing via command line or Figure 8-1 when using SMITTY.

Example 8-5 AIX: Manual installation

```
# installp -aXY -d . XIV_host_attach-1.1.0.1-aix.bff
```

```
                                Install Software

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* INPUT device / directory for software               .
* SOFTWARE to install                                [+ 1.1.0.1  IBM 2810XI] +
  PREVIEW only? (install operation will NOT occur)    no                      +
  COMMIT software updates?                            yes                     +
  SAVE replaced files?                                no                      +
  AUTOMATICALLY install requisite software?           yes                     +
  EXTEND file systems if space needed?                yes                     +
  OVERWRITE same or newer versions?                   no                      +
  VERIFY install and check file sizes?                no                      +
  Include corresponding LANGUAGE filesets?            yes                     +
  DETAILED output?                                    no                      +
  Process multiple volumes?                           yes                     +
  ACCEPT new license agreements?                      yes                     +
  Preview new LICENSE agreements?                     no                      +

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-1 Smitty install

4. Complete the installation by rebooting the server to put the installation into effect. Use this command:

   `shutdown -Fr`

When the reboot has completed, listing the disks should display the correct number of disks seen from the XIV storage. They are labeled as XIV disks, as illustrated in Example 8-6.

Example 8-6 AIX: XIV labeled FC disks

```
# lsdev -Cc disk
hdisk0  Available 1Z-08-00-8,0  16 Bit LVD SCSI Disk Drive
hdisk1  Available 1n-08-02      IBM 2810XIV Fibre Channel Disk
hdisk2  Available 1n-08-02      IBM 2810XIV Fibre Channel Disk
```

AIX Multi-path I/O (MPIO)

AIX MPIO is an enhancement to the base OS environment that provides native support for multi-path Fibre Channel storage attachment. MPIO automatically discovers, configures, and makes available every storage device path. The storage device paths are managed to provide high availability and load balancing for storage I/O. MPIO is part of the base AIX kernel and is available with the current supported AIX levels.

The MPIO base functionality of MPIO limited. It provides an interface for vendor-specific Path Control Modules (PCMs) that allow for implementation of advanced algorithms.

For basic information about MPIO, refer to the online guide *AIX 5L System Management Concepts: Operating System and Devices* from the AIX documentation Web site at:

http://publib.boulder.ibm.com/pseries/en_US/aixbman/admnconc/hotplug_mgmt.htm#mpioconcepts

The management of MPIO devices is described in the online guide *System Management Guide: Operating System and Devices for AIX 5L* from the AIX documentation Web site at:

http://publib.boulder.ibm.com/pseries/en_US/aixbman/baseadmn/manage_mpio.htm

Configuring XIV devices as MPIO or non-MPIO devices

Configuring XIV devices as MPIO provides the optimum solution. In some cases, you could be using a third party multipathing solution for managing other storage devices and wants to manage the XIV 2810 with the same solution. This usually requires the XIV devices to be configured as non-MPIO devices.

AIX provides a command to migrate a device between MPIO and non-MPIO. The manage_disk_drivers command can be used to change how the XIV device is configured (MPIO or non-MPIO). The command cause all XIV disks to be converted. It is not possible to convert one XIV disk to MPIO and another XIV disk non-MPIO.

- To migrate XIV 2810 devices from MPIO to non-MPIO, run the following command:

  ```
  manage_disk_drivers -o AIX_non_MPIO  -d 2810XIV
  ```

- To migrate XIV 2810 devices from non-MPIO to MPIO, run the following command:

  ```
  manage_disk_drivers -o AIX_AAPCM  -d 2810XIV
  ```

After running either of the foregoing commands, the system will need to be rebooted in order for the configuration change to take effect.

To display the present settings, run the following command:

```
manage_disk_drivers -l
```

Disk behavior algorithms and queue depth settings

In a multi-path environment using the XIV Storage System, you can change the disk behavior algorithm from round_robin to fail_over mode or from fail_over to round_robin mode. The default disk behavior mode is round_robin, with a queue depth setting of 32. To check the disk behavior algorithm and queue depth setting, see Example 8-7.

Example 8-7 AIX: Viewing disk behavior and queue depth

```
# lsattr -El hdisk2 | grep -e algorithm -e queue_depth
algorithm       round_robin              Algorithm           True
queue_depth     32                       Queue DEPTH         True
```

With regard to queue depth settings, the initial release of the XIV Storage System (release 10.0.x) had limited support when using round_robin in that the queue depth could only be set to one. More importantly, *note that this queue depth restriction in round_robin is lifted with the introduction AIX 53TL10 and AIX 61TL3*. No such limitations exist when employing the fail_over disk behavior algorithm.

See Table 8-1 and Table 8-2 for minimum level service packs and an associated APAR list to determine the exact specification based on the AIX version installed on the host system.

Table 8-1 AIX 5.3 minimum level service packs and APARS

AIX level	APAR	Queue depth
53TL7 SP6	IZ28969	Queue depth restriction
53TL8 SP4	IZ28970	Queue depth restriction
53TL9	IZ28047	Queue depth restriction
53TL10	IZ42730	No queue depth restriction

Table 8-2 AIX 6.1 minimum level service packs and APARS.

AIX level	APAR	Queue depth
61TL0 SP6	IZ28002	Queue depth restriction
61TL1 SP2	IZ28004	Queue depth restriction
61TL2	IZ28079	Queue depth restriction
61TK3	IZ42898	No queue depth restriction

As noted earlier, the default disk behavior algorithm is round_robin with a queue depth of 32. If the appropriate AIX levels and APAR list has been met, then the queue depth restriction is lifted and the settings can be adjusted. To adjust the disk behavior algorithm and queue depth setting, see Example 8-8.

Example 8-8 AIX: Change disk behavior algorithm and queue depth command

```
# chdev -a algorithm=round_robin -a queue_depth=32 -l <hdisk#>
```

Note: In the foregoing command, *<hdisk#>* stands for a particular instance of an hdisk.

If you want the fail_over disk behavior algorithm, after making the changes in Example 8-8, load balance the I/O across the FC adapters and paths by setting the path priority attribute for each LUN so that $1/n^{th}$ of the LUNs are assigned to each of the *n* FC paths.

Useful MPIO commands

There are commands to change priority attributes for paths that can specify a preference for the path used for I/O. The effect of the priority attribute depends on whether the disk behavior algorithm attribute is set to fail_over or round_robin:

- For algorithm=fail_over, the path with the higher priority value handles all the I/Os unless there is a path failure, then the other path will be used. After a path failure and recovery, if you have IY79741 installed, I/O will be redirected down the path with the highest priority; otherwise, if you want the I/O to go down the primary path, you will have to use **chpath** to disable the secondary path, and then re-enable it. If the priority attribute is the same for all paths, the first path listed with **lspath -Hl *<hdisk>*** will be the primary path. So, you can set the primary path to be used by setting its priority value to 1, and the next path's priority (in case of path failure) to 2, and so on.
- For algorithm=round_robin, and if the priority attributes are the same, I/O goes down each path equally. If you set pathA's priority to 1 and pathB's to 255, for every I/O going down pathA, there will be 255 I/Os sent down pathB.

To change the path priority of an MPIO device, use the **chpath** command. (An example of this is shown as part of a procedure in Example 8-11.)

Initially, use the **lspath** command to display the operational status for the paths to the devices, as shown here in Example 8-9.

Example 8-9 AIX: The lspath command shows the paths for hdisk2

```
# lspath -l hdisk2 -F status:name:parent:path_id:connection
Enabled:hdisk2:fscsi5:0:5001738000130140,2000000000000
Enabled:hdisk2:fscsi5:1:5001738000130160,2000000000000
Enabled:hdisk2:fscsi6:2:5001738000130140,2000000000000
Enabled:hdisk2:fscsi6:3:5001738000130160,2000000000000
```

It can also be used to read the attributes of a given path to an MPIO-capable device, as shown in Example 8-10.

It is also good to know that the <*connection*> info is either "<*SCSI ID*>, <*LUN ID*>" for SCSI, (for example, "5,0") or "<*WWN*>, <*LUN ID*>" for FC devices.

Example 8-10 AIX: The lspath command reads attributes of the 0 path for hdisk2

```
# lspath -AHE -l hdisk2 -p fscsi5 -w "5001738000130140,2000000000000"
attribute value              description  user_settable

scsi_id   0x30a00            N/A          False
node_name 0x5001738000130000 FC Node Name False
priority  1                  Priority     True
```

As just noted, the **chpath** command is used to perform change operations on a specific path. It can either change the operational status or tunable attributes associated with a path. It cannot perform both types of operations in a single invocation.

Example 8-11 illustrates the use of the **chpath** command with an XIV Storage System, which sets the primary path to fscsi5 using the first path listed (there are two paths from the switch to the storage for this adapter). Then for the next disk, we set the priorities to 4,1,2,3 respectively. If we are in fail-over mode and assuming the I/Os are relatively balanced across the hdisks, this setting will balance the I/Os evenly across the paths.

Example 8-11 AIX: The chpath command

```
# lspath -l hdisk2 -F"status parent connection"
Enabled fscsi5 5001738000130140,2000000000000
Enabled fscsi5 5001738000130160,2000000000000
Enabled fscsi6 5001738000130140,2000000000000
Enabled fscsi6 5001738000130160,2000000000000

# chpath -l hdisk2 -p fscsi5 -w 5001738000130160,2000000000000 -a priority=2 path Changed

# chpath -l hdisk2 -p fscsi6 -w 5001738000130140,2000000000000 -a priority=3 path Changed

# chpath -l hdisk2 -p fscsi6 -w 5001738000130160,2000000000000 -a priority=4 path Changed
```

The **rmpath** command unconfigures or undefines, or both, one or more paths to a target device. It is not possible to unconfigure (undefine) the last path to a target device using the **rmpath** command. The only way to unconfigure (undefine) the last path to a target device is to unconfigure the device itself (for example, use the **rmdev** command).

8.1.2 AIX host iSCSI configuration

At the time of writing, AIX 5.3 and AIX 6.1 operating systems are supported for iSCSI connectivity with XIV (only when using the iSCSI software initiator).

To make sure that your system is equipped with the required filesets, run the **lslpp** command as shown in Example 8-12. We used the AIX Version 5.3 operating system with Technology Level 10 in our examples.

Example 8-12 Verifying installed iSCSI filesets in AIX

```
# lslpp -la "*.iscsi*"
  Fileset                      Level  State      Description
  ----------------------------------------------------------------------------
Path: /usr/lib/objrepos
  devices.common.IBM.iscsi.rte
                              5.3.9.0  COMMITTED  Common iSCSI Files
                             5.3.10.0  COMMITTED  Common iSCSI Files
  devices.iscsi.disk.rte     5.3.0.30  COMMITTED  iSCSI Disk Software
                              5.3.7.0  COMMITTED  iSCSI Disk Software
  devices.iscsi.tape.rte     5.3.0.30  COMMITTED  iSCSI Tape Software
  devices.iscsi_sw.rte        5.3.9.0  COMMITTED  iSCSI Software Device Driver
                             5.3.10.0  COMMITTED  iSCSI Software Device Driver

Path: /etc/objrepos
  devices.common.IBM.iscsi.rte
                              5.3.9.0  COMMITTED  Common iSCSI Files
                             5.3.10.0  COMMITTED  Common iSCSI Files
  devices.iscsi_sw.rte        5.3.9.0  COMMITTED  iSCSI Software Device Driver
                             5.3.10.0  COMMITTED  iSCSI Software Device Driver
```

At the time of writing this book, only AIX iSCSI software initiator is supported for connecting to the XIV Storage System.

Current limitations when using iSCSI software initiator

The code available at the time of preparing this book had limitations when using the iSCSI software initiator in AIX. These restrictions will be lifted over time:

- Single path only is supported.
- Remote boot is not supported.
- The maximum number of configured LUNs tested using the iSCSI software initiator is 128 per iSCSI target. The software initiator uses a single TCP connection for each iSCSI target (one connection per iSCSI session). This TCP connection is shared among all LUNs that are configured for a target. The software initiator's TCP socket send and receive space are both set to the system socket buffer maximum. The maximum is set by the sb_max network option. The default is 1 MB.

Volume Groups

To avoid configuration problems and error log entries when you create Volume Groups using iSCSI devices, follow these guidelines:

- Configure Volume Groups that are created using iSCSI devices to be in an inactive state after reboot. After the iSCSI devices are configured, manually activate the iSCSI-backed Volume Groups. Then, mount any associated file systems.

  Note: Volume Groups are activated during a different boot phase than the iSCSI software driver. For this reason, it is not possible to activate iSCSI Volume Groups during the boot process

- Do not span Volume Groups across non-iSCSI devices.

I/O failures

To avoid I/O failures, consider these recommendations:

- If connectivity to iSCSI target devices is lost, I/O failures occur. To prevent I/O failures and file system corruption, stop all I/O activity and unmount iSCSI-backed file systems before doing anything that will cause long term loss of connectivity to the active iSCSI targets.
- If a loss of connectivity to iSCSI targets occurs while applications are attempting I/O activities with iSCSI devices, I/O errors will eventually occur. It might not be possible to unmount iSCSI-backed file systems, because the underlying iSCSI device stays busy.
- File system maintenance must be performed if I/O failures occur due to loss of connectivity to active iSCSI targets. To do file system maintenance, run the **`fsck`** command against the effected file systems.

Configuring the iSCSI software initiator

The software initiator is configured using System Management Interface Tool (SMIT) as shown in this procedure:

1. Select **Devices**.
2. Select **iSCSI**.
3. Select **iSCSI Protocol Device**.
4. Select **Change / Show Characteristics of an iSCSI Protocol Device**.
5. After selecting the desired device, verify that the iSCSI Initiator Name value. The Initiator Name value is used by the iSCSI Target during login.

   Note: A default initiator name is assigned when the software is installed. This initiator name can be changed by the user to match local network naming conventions.

   You can issue the **`lsattr`** command as well to verify the initiator_name parameter as shown in Example 8-13.

Example 8-13 Check initiator name

```
# lsattr -El iscsi0 | grep initiator_name
initiator_name iqn.com.ibm.tucson.storage:midas iSCSI Initiator Name
```

6. The Maximum Targets Allowed field corresponds to the maximum number of iSCSI targets that can be configured. If you reduce this number, you also reduce the amount of network memory pre-allocated for the iSCSI protocol driver during configuration.

After the software initiator is configured, define iSCSI targets that will be accessed by the iSCSI software initiator. To specify those targets:

1. First, determine your iSCSI IP addresses in the XIV Storage System. To get that information, select iSCSI Connectivity from the Host and LUNs menu as shown in Figure 8-2.

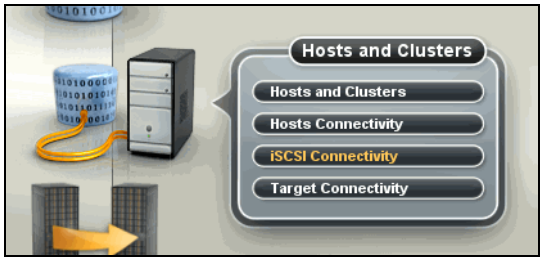


Figure 8-2 iSCSI Connectivity

Or just issue the command in Example 8-14 in the XCLI.

Example 8-14 List iSCSI interfaces

```
>> ipinterface_list
Name            Type    IP Address    Network Mask   Default Gateway   MTU   Module       Ports
Midas_iSCSI7-1  iSCSI   9.11.245.87   255.255.254.0  9.11.244.1        1500  1:Module:7   1
```

2. The next step is find the iSCSI name (IQN) of the XIV Storage. To get this information, navigate to the basic system view in the XIV GUI and right-click the XIV Storage box itself and select **Properties**. The System Properties window appears as shown in Figure 8-3.

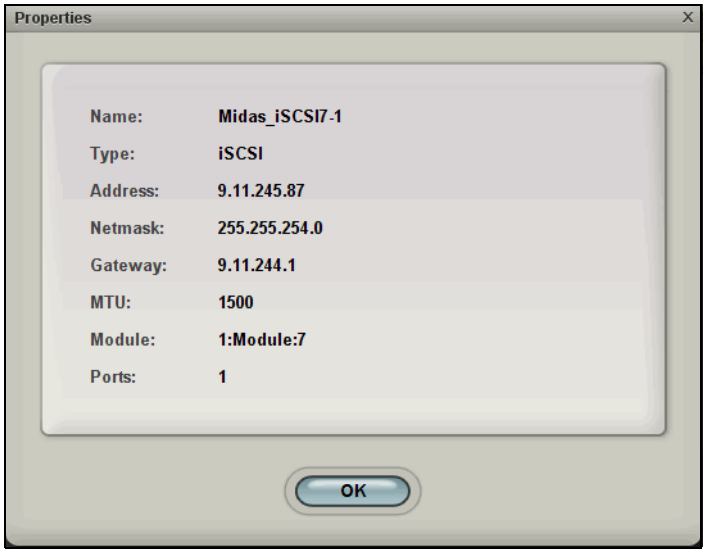


Figure 8-3 Verifying iSCSI name in XIV Storage System

If you are using XCLI, issue the **config_get** command. Refer to Example 8-15.

Example 8-15 The config_get command in XCLI

```
>> config_get
Name                          Value
dns_primary                   9.11.224.114
dns_secondary                 9.11.224.130
email_reply_to_address
email_sender_address          XIVbox@us.ibm.com
email_subject_format          {severity}:{description}
iscsi_name                    iqn.2005-10.com.xivstorage:000035
machine_model                 A14
machine_serial_number         MN00035
machine_type                  2810
ntp_server                    9.11.224.116
snmp_community                XIV
snmp_contact                  Unknown
snmp_location                 Unknown
snmp_trap_community           XIV
support_center_port_type      Management
system_id                     35
system_name                   XIV MN00035
```

3. Go back to the AIX system and edit the /etc/iscsi/targets file to include the iSCSI targets needed during device configuration:

Note: The iSCSI targets file defines the name and location of the iSCSI targets that the iSCSI software initiator will attempt to access. This file is read any time that the iSCSI software initiator driver is loaded.

– Each uncommented line in the file represents an iSCSI target.

– iSCSI device configuration requires that the iSCSI targets can be reached through a properly configured network interface. Although the iSCSI software initiator can work using a 10/100 Ethernet LAN, it is designed for use with a gigabit Ethernet network that is separate from other network traffic.

Include your specific connection information in the targets file as shown in Example 8-16. Insert a HostName PortNumber and iSCSIName similar to what is shown in this example.

Example 8-16 Inserting connection information into /etc/iscsi/targets file in AIX operating system

```
9.11.245.87 3260 iqn.2005-10.com.xivstorage:000209
```

4. After editing the /etc/iscsi/targets file, enter the following command at the AIX prompt:

   ```
   cfgmgr -l iscsi0
   ```

   This command will reconfigure the software initiator driver, and this command causes the driver to attempt to communicate with the targets listed in the /etc/iscsi/targets file, and to define a new hdisk for each LUN found on the targets.

   Note: If the appropriate disks are not defined, review the configuration of the initiator, the target, and any iSCSI gateways to ensure correctness. Then, rerun the **cfgmgr** command.

iSCSI performance considerations

To ensure the best performance, enable the TCP Large Send, TCP send and receive flow control, and Jumbo Frame features of the AIX Gigabit Ethernet Adapter and the iSCSI Target interface.

Tune network options and interface parameters for maximum iSCSI I/O throughput on the AIX system:

- Enable the RFC 1323 network option.
- Set up the tcp_sendspace, tcp_recvspace, sb_max, and mtu_size network options and network interface options to appropriate values:
  - The iSCSI software initiator's maximum transfer size is 256 KB. Assuming that the system maximums for tcp_sendspace and tcp_recvspace are set to 262144 bytes, an **ifconfig** command used to configure a gigabit Ethernet interface might look like:

    ```
    ifconfig en2 10.1.2.216 mtu 9000 tcp_sendspace 262144 tcp_recvspace 262144
    ```
- Set the sb_max network option to at least 524288, and preferably 1048576.
- Set the mtu_size to 9000.
- For certain iSCSI targets, the TCP Nagle algorithm must be disabled for best performance. Use the **no** command to set the tcp_nagle_limit parameter to 0, which will disable the Nagle algorithm.

8.1.3 Management volume LUN 0

According to the SCSI standard, XIV Storage System maps itself in every map to LUN 0. This LUN serves as the "well known LUN" for that map, and the host can issue SCSI commands to that LUN that are not related to any specific volume. This device appears as a normal hdisk in the AIX operating system, and because it is not recognized by Windows by default, it appears with an unknown device's question mark next to it.

Exchange management of LUN 0 to a real volume

You might want to eliminate this management LUN on your system, or you have to assign the LUN 0 number to a specific volume. In that case, all you need to do is just map your volume to the first place in the mapping view and it will replace the management LUN to your volume and assign the zero value to it. To see the mapping method, refer to 6.4, “Logical configuration for host connectivity” on page 209.

8.2 SAN boot in AIX

This section contains a step-by-step illustration of SAN boot implementation for the IBM POWER System (formerly System p®) in an AIX v5.3 environment. Similar steps can be followed for an AIX v6.1 environment.

There are various possible implementations of SAN boot with AIX:

- To implement SAN boot on a system with an already installed AIX operating system, you can do this by mirroring of the rootvg volume to the SAN disk.
- To implement SAN boot for a new system, you can start the AIX installation from a bootable AIX CD install package or use the Network Installation Manager (NIM).

The method known as *mirroring* is simpler to implement than the more complete and more sophisticated method using the Network Installation Manager.

8.2.1 Creating a SAN boot disk by mirroring

The mirroring method requires that you have access to an AIX system that is up and running. If it is not already available, you must locate an available system where you can install AIX on an internal SCSI disk.

To create a boot disk on the XIV system:

1. Select a logical drive that is the same size or larger than the size of rootvg that currently resides on the internal SCSI disk. Ensure that your AIX system can see the new disk. You can verify this with the **lspv** command. Verify the size with **bootinfo** and use **lsdev** to make sure that you are using an XIV (external) disk
2. Add the new disk to the rootvg volume group with **smitty vg** → **Set Characteristics of a Volume Group** → **Add a Physical Volume from a Volume Group** (see Figure 8-4).

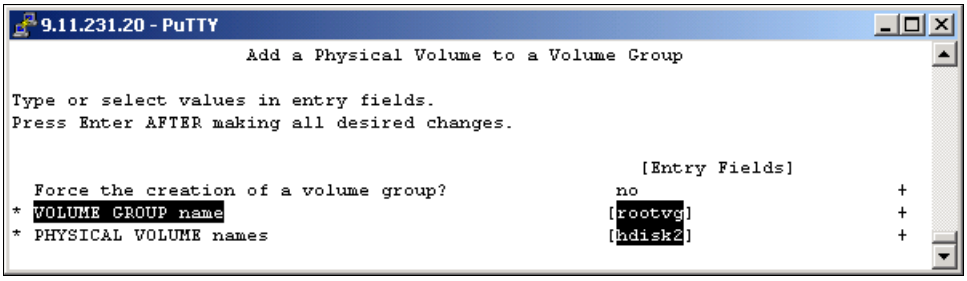


Figure 8-4 Add the disk to the rootvg

3. Create the mirror of rootvg. If the rootvg is already mirrored you can create a third copy on the new disk with **smitty vg**-> **Mirror a Volume Group**, then select the rootvg and the new hdisk.

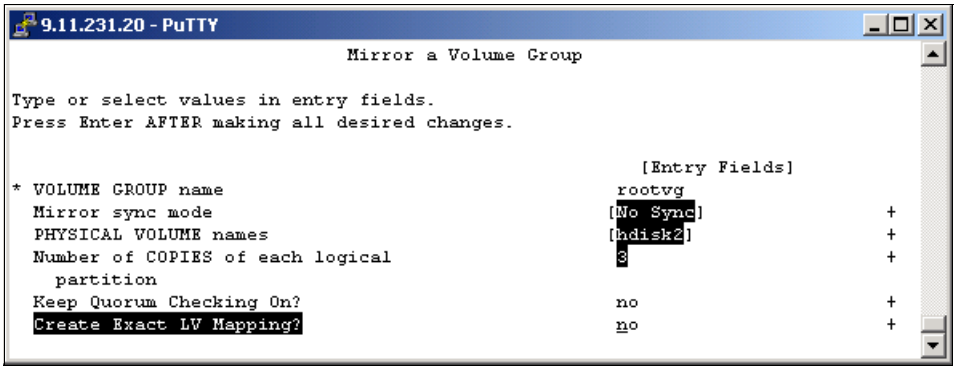


Figure 8-5 Create a rootvg mirror

4. Verify that all partitions are mirrored (Figure 8-6) with **`lsvg -l rootvg`**, recreate the boot logical drive, and change the normal boot list with the following commands:

   ```
   bosboot -ad hdiskx
   bootlist -m normal hdiskx
   ```

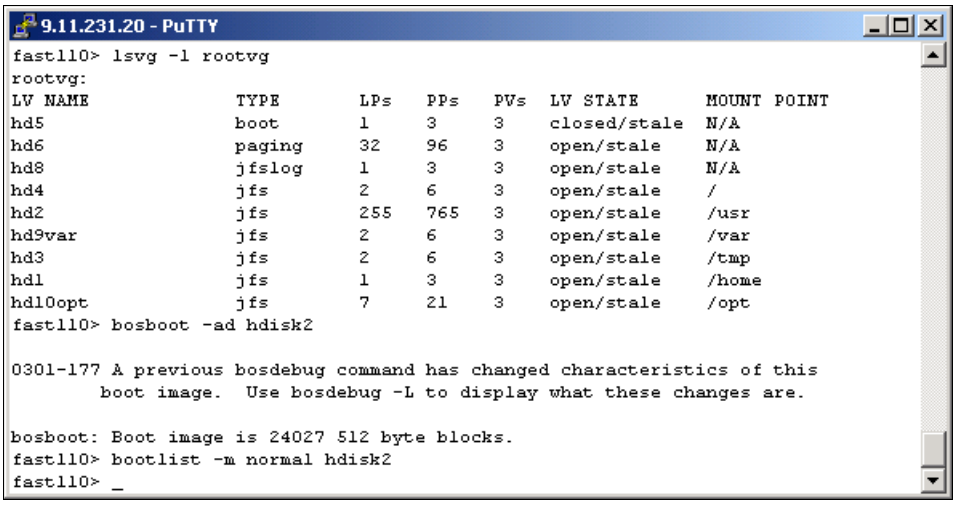


Figure 8-6 Verify that all partitions are mirrored

5. The next step is to remove the original mirror copy with **smitty vg-> Unmirror a Volume Group**. Choose the rootvg volume group, then the disks that you want to remove from mirror and run the command.
6. Remove the disk from the volume group rootvg with **smitty vg**-> **Set Characteristics of a Volume Group**-> **Remove a Physical Volume from a Volume Group**, select rootvg for the volume group name ROOTVG and the internal SCSI disk you want to remove, and run the command.
7. We recommend that you execute the following commands again (see step 4):

   ```
   bosboot -ad hdiskx
   bootlist -m normal hdiskx
   ```

At this stage, the creation of a bootable disk on the XIV is completed. Restarting the system makes it boot from the SAN (XIV) disk.

8.2.2 Installation on external storage from bootable AIX CD-ROM

To install AIX on XIV System disks, make the following preparations:

1. Update the Fibre Channel (FC) adapter (HBA) microcode to the latest supported level.
2. Make sure that you have an appropriate SAN configuration: The host is properly connected to the SAN, the zoning configuration is updated, and at least one LUN is mapped to the host.

Note: If the system cannot see the SAN fabric at login, you can configure the HBAs at the server open firmware prompt.

Because by nature, a SAN allows access to a large number of devices, identifying the hdisk to install to can be difficult. We recommend the following method to facilitate the discovery of the lun_id to hdisk correlation:

1. If possible, zone the switch or disk array such that the machine being installed can only discover the disks to be installed to. After the installation has completed, you can then reopen the zoning so the machine can discover all necessary devices.
2. If more than one disk is assigned to the host, make sure that you are using the correct one, as follows:
   - If possible, assign Physical Volume Identifiers (PVIDs) to all disks from an already installed AIX system that can access the disks. This can be done using the command:

     ```
     chdev -a pv=yes -l hdiskX
     ```

     Where *X* is the appropriate disk number. Create a table mapping PVIDs to physical disks. The PVIDs will be visible from the install menus by selecting option **77 display more disk info** (AIX 5.3 install) when selecting a disk to install to. Or you could use the PVIDs to do an unprompted Network Installation Management (NIM) install.
   - Another way to ensure the selection of the correct disk is to use Object Data Manager (ODM) commands. Boot from the AIX installation CD-ROM and from the main install menu, then select **Start Maintenance Mode for System Recovery** → **Access Advanced Maintenance Functions** → **Enter the Limited Function Maintenance Shell**. At the prompt, issue the command:

     ```
     odmget -q "attribute=lun_id AND value=OxNN..N" CuAt
     ```

     or

     ```
     odmget -q "attribute=lun_id" CuAt (list every stanza with lun_id attribute)
     ```

     Where *OxNN..N* is the lun_id that you are looking for. This command prints out the ODM stanzas for the hdisks that have this lun_id. Enter `Exit` to return to the installation menus.

The Open Firmware implementation can only boot from lun_ids O through 7. The firmware on the Fibre Channel adapter (HBA) promotes this lun_id to an 8-byte FC lun-id by adding a byte of zeroes to the front and 6 bytes of zeroes to the end. For example, lun_id 2 becomes 0x0002000000000000. Note that usually the lun_id will be displayed without the leading zeroes. Care must be taken when installing because the installation procedure will allow installation to lun_ids outside of this range.

Installation procedure

Follow these steps:

1. Insert an AIX CD that has a bootable image into the CD-ROM drive.
2. Select **CD-ROM** as the install device to make the system boot from the CD. The way to change the bootlist varies model by model. In most System p models, this can be done by using the System Management Services (SMS) menu. Refer to the user's guide for your model.
3. Let the system boot from the AIX CD image after you have left the SMS menu.
4. After a few minutes the console should display a window that directs you to press the specified key on the device to be used as the system console.
5. A window is displayed that prompts you to select an installation language.
6. The Welcome to the Base Operating System Installation and Maintenance window is displayed. Change the installation and system settings that have been set for this machine in order to select a Fibre Channel-attached disk as a target disk. Type `2` and press Enter.
7. At the Installation and Settings window you should enter `1` to change the system settings and choose the **New and Complete Overwrite** option.
8. You are presented with the Change (the destination) Disk window. Here you can select the Fibre Channel disks that are mapped to your system. To make sure and get more information, type `77` to display the detailed information window. The system shows the PVID. Type `77` again to show WWPN and LUN_ID information. Type the number, but do not press Enter, for each disk that you choose. Typing the number of a selected disk deselects the device. Be sure to choose an XIV disk.
9. After you have selected Fibre Channel-attached disks, the Installation and Settings window is displayed with the selected disks. Verify the installation settings. If everything looks okay, type `0` and press Enter and the installation process begins.

   Important: Be sure that you have made the correct selection for root volume group because the existing data in the destination root volume group will be destroyed during BOS installation.

10. When the system reboots, a window message displays the address of the device from which the system is reading the boot image.

8.2.3 AIX SAN installation with NIM

Network Installation Manager (NIM) is a client server infrastructure and service that allows remote install of the operating system, manages software updates, and can be configured to install and update third-party applications. Although both the NIM server and client file sets are part of the operating system, a separate NIM server has to be configured, which keeps the configuration data and the installable product file sets.

We assume that the NIM environment is deployed and all of the necessary configurations on the NIM master are already done:

- The NIM server is properly configured as the NIM master and the basic NIM resources have been defined.
- The Fibre Channel Adapters are already installed on the machine onto which AIX is to be installed.
- The Fibre Channel Adapters are connected to a SAN and on the XIV system have at least one logical volume (LUN) mapped to the host.

- The target machine (NIM client) currently has no operating system installed and is configured to boot from the NIM server.

For more information about how to configure a NIM server, refer to the *AIX 5L Version 5.3: Installing AIX* reference, SC23-4887-02.

Installation procedure

Prior the installation, you should modify the bosinst.data file, where the installation control is stored.

Insert your appropriate values at the following stanza:

```
SAN_DISKID
```

This specifies the worldwide port name and a logical unit ID for Fibre Channel-attached disks. The worldwide port name and logical unit ID are in the format returned by the **lsattr** command (that is, 0x followed by 1–16 hexadecimal digits). The ww_name and lun_id are separated by two slashes (//).

```
SAN_DISKID = <worldwide_portname//lun_id>
```

For example:

```
SAN_DISKID = 0x0123456789FEDCBA//0x2000000000000
```

Or you can specify PVID (example with internal disk):

```
target_disk_data:
PVID = 000c224a004a07fa
SAN_DISKID =
CONNECTION = scsi0//10,0
LOCATION = 10-60-00-10,0
SIZE_MB = 34715
HDISKNAME = hdisk0
```

To install:

1. Enter the command:

   ```
   # smit nim_bosinst
   ```

2. Select the **lpp_source** resource for the BOS installation.
3. Select the **SPOT** resource for the BOS installation.
4. Select the **BOSINST_DATA to use during installation** option, and select a bosinst_data resource that is capable of performing a non prompted BOS installation.
5. Select the **RESOLV_CONF to use for network configuration** option, and select a resolv_conf resource.
6. Select the **Accept New License Agreements** option, and select **Yes**. Accept the default values for the remaining menu options.
7. Press Enter to confirm and begin the NIM client installation.
8. To check the status of the NIM client installation, enter:

   ```
   # lsnim -l va09
   ```

9

Linux host connectivity

This chapter explains specific considerations for attaching the XIV system to a Linux host.

9.1 Attaching a Linux host to XIV

Linux is different from the other proprietary operating systems in many ways:

- There is no one person or organization that can be held responsible or called for support.
- Depending on the target group, the distributions differ largely in the kind of support that is available.
- Linux is available for almost all computer architectures.
- Linux is rapidly changing.

All these factors make it difficult to promise and provide generic support for Linux. As a consequence, IBM has decided on a support strategy that limits the uncertainty and the amount of testing.

IBM supports the major Linux distributions that are targeted at enterprise clients:

- Red Hat Enterprise Linux
- SUSE® Linux Enterprise Server

These distributions have release cycles of about one year, are maintained for five years, and require you to sign a support contract with the distributor. They also have a schedule for regular updates. These factors mitigate the issues listed previously. The limited number of supported distributions also allows IBM to work closely with the vendors to ensure interoperability and support.

Details about the supported Linux distributions and supported SAN boot environments can be found in the System Storage Interoperability Center (SSIC):

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

9.2 Linux host FC configuration

This section describes attaching a Linux host to XIV over Fibre Channel and provides detailed descriptions and installation instructions for the various software components required.

9.2.1 Installing supported Qlogic device driver

Download a supported driver version for the QLA2340. Unless otherwise noted in SSIC, use any supported driver and firmware by the HBA vendors (the latest versions are always preferred). For HBAs in Sun systems, use Sun branded HBAs and Sun ready HBAs only. The SSIC is at:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

Install the driver as shown in Example 9-1.

Example 9-1 QLogic driver installation and setup

```
[root@x345-tic-30 ~]# tar -xvzf qla2xxx-v8.02.14_01-dist.tgz
qlogic/
qlogic/drvrsetup
qlogic/libinstall
qlogic/libremove
```

```
qlogic/qla2xxx-src-v8.02.14_01.tar.gz
qlogic/qlapi-v4.00build12-rel.tgz
qlogic/README.qla2xxx
[~]# cd qlogic/
[qlogic]# ./drvrsetup
Extracting QLogic driver source...
Done.
[qlogic]# cd qla2xxx-8.02.14/
[qla2xxx-8.02.14]# ./extras/build.sh install

QLA2XXX -- Building the qla2xxx driver, please wait...
Installing intermodule.ko in /lib/modules/2.6.18-128.1.6.el5/kernel/kernel/
QLA2XXX -- Build done.

QLA2XXX -- Installing the qla2xxx modules to
/lib/modules/2.6.18-128.1.6.el5/kernel/drivers/scsi/qla2xxx/...
```

Set the queue depth to 64, disable the failover mode for the driver, and set the time-out for a PORT-DOWN status before returning I/O back to the OS to"1" in the /etc/modprobe.conf. Refer to Example 9-2 for details.

Example 9-2 Modification of /etc/modprobe.conf for the XIV

```
[qla2xxx-8.02.14]# cat >> /etc/modprobe.conf << EOF
> options qla2xxx qlport_down_retry=1
> options qla2xxx ql2xfailover=0
> options qla2xxx ql2xmaxqdepth=64
> EOF
[qla2xxx-8.02.14]# cat /etc/modprobe.conf
alias eth0 tg3
alias eth1 tg3
alias scsi_hostadapter mptbase
alias scsi_hostadapter1 mptspi
alias scsi_hostadapter2 qla2xxx
options qla2xxx qlport_down_retry=1
options qla2xxx ql2xfailover=0
options qla2xxx ql2xmaxqdepth=64
install qla2xxx /sbin/modprobe qla2xxx_conf; /sbin/modprobe --ignore-install
qla2xxx
remove qla2xxx /sbin/modprobe -r --first-time --ignore-remove qla2xxx && {
/sbin/modprobe -r --ignore-remove qla2xxx_conf; }
alias qla2100 qla2xxx
alias qla2200 qla2xxx
alias qla2300 qla2xxx
alias qla2322 qla2xxx
alias qla2400 qla2xxx
options qla2xxx qlport_down_retry=1
options qla2xxx ql2xfailover=0
```

We now have to build a new RAM disk image, so that the driver will be loaded by the operating system loader after a boot. Next, we reboot the Linux host as shown in Example 9-3.

Example 9-3 Build a new ram disk image

```
[qla2xxx-8.02.14]# cd /boot/
```

```
[root@x345-tic-30 boot]# cp -f initrd-2.6.18-128.1.6.el5.img
initrd-2.6.18-92.el5.img.bak
[boot]# mkinitrd -f initrd-2.6.18-128.1.6.el5.img 2.6.18-128.1.6.el5
[boot]# reboot

Broadcast message from root (pts/1) (Tue June 30 13:57:28 2009):

The system is going down for reboot NOW!
```

9.2.2 Linux configuration changes

In this step, we make changes to the Linux configuration to support the XIV Storage System. Disable Security-enhanced Linux in the /etc/selinux/config file, according to Example 9-4.

Example 9-4 Modification of /etc/selinux/config

```
[/] # cat /etc/selinux/config
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#       enforcing - SELinux security policy is enforced.
#       permissive - SELinux prints warnings instead of enforcing.
#       disabled - SELinux is fully disabled.
SELINUX=disabled
# SELINUXTYPE= type of policy in use. Possible values are:
#       targeted - Only targeted network daemons are protected.
#       strict - Full SELinux protection.
SELINUXTYPE=targeted
```

9.2.3 Obtain WWPN for XIV volume mapping

To map the volumes to the Linux host, you must know the World Wide Port Names (WWPNs) of the HBAs. WWPNs can be found in the SYSFS. Refer to Example 9-5 for details.

Example 9-5 WWPNs of the HBAs

```
# cat /sys/class/fc_host/host1/port_name
0x210000e08b13d6bb
# cat /sys/class/fc_host/host2/port_name
0x210000e08b13f3c1
```

Create and map new volumes to the Linux host, as described in 4.5, “Host definition and mappings” on page 118.

9.2.4 Installing the Host Attachment Kit

This section explains how to install the Host Attachment Kit on a Linux server:

Download Linux **rpm** packages to the server. Regardless of the type of connectivity you are going to implement (FC or iSCSI), the following **rpm** packages are mandatory:

```
host_attach-<package_version>.noarch.rpm
xpyv.<package_version>.<glibc_version>.<linux-version>.rpm
```

The **rpm** packages for the Host Attachment Kit packages are dependent on several software packages that are needed on the host machine. The following software packages are generally required to be installed on the system:

```
device-mapper-multipath
sg3-utils
python
```

These software packages are supplied on the installation media of the supported Linux distributions. If one of more required software packages are missing on your host, the installation of the Host Attachment Kit package will stop, and you will be notified of package names required to be installed prior to installing the Host Attachment Kit package.

To install the HAK, open a terminal session and make current the directory where the package was downloaded. Execute the following command to extract the archive:

```
# gunzip –c XIV_host_attach-1.1.*-*.tar.gz | tar xvf –
```

Go to the newly created directory and invoke the Host Attachment Kit installer:

```
# cd XIV_host_attach-1.1.*-<platform>
# /bin/sh ./install.sh
```

Follow the prompts.

After running the installation script, review the installation log file `install.log` residing in the same directory.

9.2.5 Configuring the host

Use the utilities provided in the Host Attachment Kit to configure the Linux host. Host Attach Kit packages are installed in `/opt/xiv/host_attach` directory.

Note: You must be logged in as root or with root privileges to use the Host Attachment Kit.

The main executable file the is used for fiber channel host attachments is:

`/opt/xiv/host_attach/bin/`**`xiv_attach`**

Refer to Example 9-6 for illustration.

Example 9-6 Fiber channel host attachment configuration

```
# /opt/xiv/host_attach/bin/xiv_attach
-------------------------------------------------------------------------------
Welcome to the XIV host attachment wizard. This wizard will
guide you through the process of attaching your host to the XIV system.

Are you ready to configure this host for the XIV system? [default: no]: y
-------------------------------------------------------------------------------
Please wait while the wizard validates your existing configuration...
The wizard will now configure the host for the XIV system
Press [Enter] to proceed
Please wait while the host is configured...
The host is now configured for the XIV system
-------------------------------------------------------------------------------
Would you like to discover new XIV Storage devices now? [default: yes]:
```

```
------------------------------------------------------------------------------
Would you like to discover fiber (fc) or iscsi devices? fc
------------------------------------------------------------------------------
Please open the console and define the host with the relevant fiber channel WWPNs:
21:00:00:e0:8b:13:d6:bb:
        [QLOGIC]:       N/A
21:00:00:e0:8b:13:f3:c1:
        [QLOGIC]:       N/A
Press [Enter] to proceed
Would you like to rescan for fiber channel storage devices now? [default: yes]:
Please wait while rescanning for storage devices...
------------------------------------------------------------------------------
The XIV host attachment wizard successfuly configured this host
```

9.3 Linux host iSCSI configuration

Follow these steps to configure the Linux host for iSCSI attachment with multipathing:

1. Install the iSCSI initiator package.
2. Installing Host Attachment Kit.
3. Configuring iSCSI connectivity with Host Attachment Kit
4. Verifying iSCSI targets and multipathing

Our environment to prepare the examples that we present in the remainder of this section consisted of an IBM System x server x345, running Red Hat Enterprise Linux 5.2 with the iSCSI software initiator.

9.3.1 Install the iSCSI initiator package

Download a supported iSCSI driver version according to the System Storage Interoperability Center (SSIC) at:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

Install the iSCSI initiator and change the configuration, so that iSCSI driver will be automatically loaded by the OS loader, as shown in Example 9-7.

Example 9-7 Installation of the iSCSI initiator

```
[/]# rpm -ivh iscsi-initiator-utils-6.2.0.868-0.7.el5.i386.rpm
warning: iscsi-initiator-utils-6.2.0.868-0.7.el5.i386.rpm: Header V3 DSA
signature: NOKEY, key ID 37017186
Preparing...                ########################################### [100%]
   1:iscsi-initiator-utils  ########################################### [100%]
[/]# chkconfig --add iscsi
[/]# chkconfig --level 2345 iscsi on
```

For additional information about iSCSI in Linux environments, refer to:

http://open-iscsi.org/

9.3.2 Installing the Host Attachment Kit

Download Linux RPM packages to the server. Regardless of the type of connectivity you are going to implement (FC or iSCSI), the following RPM packages are mandatory:

```
host_attach-<package_version>.noarch.rpm
xpyv.<package_version>.<glibc_version>.<linux-version>.rpm
```

The rpm packages for the Host Attachment Kit packages are dependent on several software packages that are needed on the host machine. The following software packages are generally required to be installed on the system:

```
device-mapper-multipath
sg3-utils
python
```

These software packages are supplied on the installation media of the supported Linux distributions. If one or more required software packages are missing on your host, the installation of the Host Attachment Kit package will stop, and you will be notified of package names required be installed prior installing the Host Attachment Kit package.

To install Host Attachment Kit packages, run the installation script `install.sh` in the directory where the downloaded package is extracted. After running the installation script, review the installation log file `install.log` residing in the same directory.

9.3.3 Configuring iSCSI connectivity with Host Attachment Kit

Host Attach Kit packages are installed in `/opt/xiv/host_attach` directory.

Note: You must be logged in as root or with root privileges to use the Host Attachment Kit.

The main utility that is used for configuring iscsi host attachments is:

```
/opt/xiv/host_attach/bin/xiv_attach
```

See Example 9-8 for an illustration.

Example 9-8 XIV host attachment wizard

```
# /opt/xiv/host_attach/bin/xiv_attach
-------------------------------------------------------------------------------
Welcome to the XIV host attachment wizard. This wizard will
guide you through the process of attaching your host to the XIV system.

Are you ready to configure this host for the XIV system? [default: no]: y
-------------------------------------------------------------------------------
Please wait while the wizard validates your existing configuration...
This host is already configured for the XIV system
-------------------------------------------------------------------------------
Would you like to discover new XIV Storage devices now? [default: yes]:
-------------------------------------------------------------------------------
Would you like to discover fiber (fc) or iscsi devices? iscsi
-------------------------------------------------------------------------------
Please open the console and define the host with the following initiator name or
initiator alias:
Initiator name: iqn.1994-05.com.redhat:c0349525ce9b
Initator alias: -
```

```
Press [Enter] to proceed
Would you like to discover a new iSCSI target? [default: yes]:
Enter an XIV iSCSI discovery address: 9.11.237.208
Would you like to discover a new iSCSI target? [default: yes]:
Enter an XIV iSCSI discovery address: 9.11.237.209
Would you like to discover a new iSCSI target? [default: yes]: no
Would you like to rescan for iSCSI storage devices now? [default: yes]:
-------------------------------------------------------------------------------
The XIV host attachment wizard successfuly configured this host
```

Create and map volumes to the Linux host iscsi initiator, as described in 4.5, “Host definition and mappings” on page 118. Run rescanning of the iscsi connection (with **-s** option of the **host_attach_iscsi** command). You should see mpath devices as illustrated in Example 9-9.

Example 9-9 Rescanning and verifying mapped XIV LUNs

```
[/]# /opt/xiv/host_attach/bin/host_attach_iscsi -s
INFO: rescanning iSCSI connections for devices, this may take a while...
INFO: refreshing iSCSI connections
INFO: updating multipath maps, this may take a while...

[/]# /opt/xiv/host_attach/bin/xiv_devlist
XIV devices
===========
Device               Vol Name     XIV Host     Size    Paths XIV ID   Vol ID
-------------------------------------------------------------------------------
mpath9               orcah...l_01 orcak...scsi 17.2GB  2/2   MN00021  39
mpath11              orcah...l_03 orcak...scsi 17.2GB  2/2   MN00021  41
mpath10              orcah...l_02 orcak...scsi 17.2GB  2/2   MN00021  40

Non-XIV devices
===============
Device               Size    Paths
-----------------------------------
```

9.3.4 Verifying iSCSI targets and multipathing

To discover the iSCSI targets, use the **iscsiadm** command. Refer to Example 9-10 and Example 9-11 for details.

Example 9-10 iSCSI target discovery

```
[/]# iscsiadm -m discovery -t sendtargets -p  9.11.237.208
9.11.237.208:3260,1 iqn.2005-10.com.xivstorage:000033
9.11.237.209:3260,2 iqn.2005-10.com.xivstorage:000033
[/]# iscsiadm -m discovery -t sendtargets -p  9.11.237.209
9.11.237.208:3260,1 iqn.2005-10.com.xivstorage:000033
9.11.237.209:3260,2 iqn.2005-10.com.xivstorage:000033
[/]# cat /etc/iscsi/initiatorname.iscsi
InitiatorName=iqn.1994-05.com.redhat:c0349525ce9b
```

Example 9-11 iSCSI multipathing output

```
[/]# multipathd -k"show topo"
mpath10 (20017380000210028) dm-2  IBM,2810XIV
```

```
[size=16G ][features=1 queue_if_no_path][hwhandler=0        ][rw        ]
\_ round-robin 0 [prio=2][active]
 \_ 16:0:0:2 sdd 8:48  [active][ready]
 \_ 15:0:0:2 sdc 8:32  [active][ready]
mpath9  (20017380000210027) dm-3  IBM,2810XIV
[size=16G ][features=1 queue_if_no_path][hwhandler=0        ][rw        ]
\_ round-robin 0 [prio=2][enabled]
 \_ 16:0:0:1 sdb 8:16  [active][ready]
 \_ 15:0:0:1 sda 8:0   [active][ready]
mpath11 (20017380000210029) dm-4  IBM,2810XIV
[size=16G ][features=1 queue_if_no_path][hwhandler=0        ][rw        ]
\_ round-robin 0 [prio=2][active]
 \_ 15:0:0:3 sde 8:64  [active][ready]
 \_ 16:0:0:3 sdf 8:80  [active][ready]

[/]# multipathd -k"list paths"
hcil     dev dev_t pri dm_st   chk_st  next_check
16:0:0:1 sdb 8:16  1   [active][ready] XXXXX..... 10/20
15:0:0:1 sda 8:0   1   [active][ready] XXXXX..... 10/20
15:0:0:2 sdc 8:32  1   [active][ready] XXXXX..... 10/20
16:0:0:2 sdd 8:48  1   [active][ready] XXXXX..... 10/20
16:0:0:3 sdf 8:80  1   [active][ready] XXXXX..... 10/20
15:0:0:3 sde 8:64  1   [active][ready] XXXXX..... 10/20
[/]# multipathd -k"list maps status"
name    failback queueing paths dm-st  write_prot
mpath10 -        5 chk    2     active rw
mpath9  -        5 chk    2     active rw
mpath11 -        5 chk    2     active rw
```

9.4 Linux Host Attachment Kit utilities

The Host Attachment Kit (HAK) now includes the following utilities:

- **xiv_devlist**

  xiv_devlist is the command allowing validation of the attachment configuration. This command generates a list of multipathed devices available to the operating system. An illustration is given in Example 9-12.

Example 9-12 List of multipathed IBM-XIV devices

```
# /opt/xiv/host_attach/bin/xiv_devlist
XIV devices
===========
Device               Vol Name     XIV Host     Size    Paths XIV ID   Vol ID
-------------------------------------------------------------------------------
mpath2               orcah...l_10 orcakpvhd97  17.2GB  4/4   MN00021  48
mpath1               orcah...l_09 orcakpvhd97  17.2GB  4/4   MN00021  47
mpath4               orcah...l_12 orcakpvhd97  17.2GB  4/4   MN00021  50
mpath0               orcah...l_08 orcakpvhd97  17.2GB  4/4   MN00021  46
mpath3               orcah...l_11 orcakpvhd97  17.2GB  4/4   MN00021  49
mpath5               orcah...l_13 orcakpvhd97  17.2GB  4/4   MN00021  51
```

- **xiv_diag**

  The utility gathers diagnostic information from the operating system. The resulting zip file can then be sent to IBM-XIV support teams for review and analysis. To run, go to a command prompt and enter `xiv_diag`. See the illustration in Example 9-13.

Example 9-13 xiv_diag command

```
xiv_diag
Please type in a directory in which to place the xiv_diag file [default:
/tmp]:
Creating xiv_diag zip file /tmp/xiv_diag-results_2009-6-24_15-7-45.zip
...
...
INFO: Closing xiv_diag zip file /tmp/xiv_diag-results_2009-6-24_19-18-
4.zip
Deleting temporary directory...
DONE
INFO: Gathering is now complete.
INFO: You can now send /tmp/xiv_diag-results_2009-6-24_19-18-4.zip to IBMXIV
for review.
INFO: Exiting.
```

- **wfetch**

  This is a simple CLI utility for downloading files from HTTP, HTTPS, and FTP sites. It runs on most UNIX, Linux, and Windows operating systems.

9.5 Partitions and filesystems

This section illustrates the creation and use of partition and filesystems from XIV provided storage.

9.5.1 Creating partitions and filesystems without LVM

The mutlipathed devices can be used for creating partitions and filesystems in traditional non-LVM form, as illustrated in Example 9-14.

Example 9-14 Creating partition on multipath device, forcing read partition into running kernel

```
# fdisk /dev/mapper/mpath1

The number of cylinders for this disk is set to 2088.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
```

```
First cylinder (1-2088, default 1):
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-2088, default 2088):
Using default value 2088

Command (m for help): p
Disk /dev/mapper/mpath1: 17.1 GB, 17179869184 bytes
255 heads, 63 sectors/track, 2088 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

              Device Boot      Start         End      Blocks   Id  System
/dev/mapper/mpath1p1               1        2088    16771828+  83  Linux

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
WARNING: Re-reading the partition table failed with error 22: Invalid argument.
The kernel still uses the old table.
The new table will be used at the next reboot.
Syncing disks.

[/]#  partprobe -s /dev/mapper/mpath1
/dev/mapper/mpath1: msdos partitions 1
[/]# fdisk -l  /dev/mapper/mpath1

Disk /dev/mapper/mpath1: 17.1 GB, 17179869184 bytes
255 heads, 63 sectors/track, 2088 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

              Device Boot      Start         End      Blocks   Id  System
/dev/mapper/mpath1p1               1        2088    16771828+  83  Linux
```

Note that the **partprobe** command, the program that informs the operating system kernel of partition table changes, needs to be run to overcome the condition, "*The kernel still uses the old table.*" and to make the running system aware of the newly created partition.

Example 9-15 shows the creation and mounting of a filesystem.

Example 9-15 Creating and mounting filesystem

```
# mkfs -t ext3 /dev/mapper/mpath1p1
mke2fs 1.39 (29-May-2006)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
2097152 inodes, 4192957 blocks
209647 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=0
128 block groups
32768 blocks per group, 32768 fragments per group
16384 inodes per group
Superblock backups stored on blocks:
  32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
  4096000
```

```
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 20 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.

# fsck /dev/mapper/mpath1p1
fsck 1.39 (29-May-2006)
e2fsck 1.39 (29-May-2006)
/dev/mapper/mpath1p1: clean, 11/2097152 files, 109875/4192957 blocks
# mkdir /tempmount
# mount /dev/mapper/mpath1p1 /tempmount
# df -m /tempmount
Filesystem           1M-blocks      Used Available Use% Mounted on
/dev/mapper/mpath1p1     16122       173     15131   2% /tempmount
```

The new filesystem is mounted to a temporary location `/tempmount`. To make this mount point persistent across reboots, you have to create an entry in `/etc/fstab` to mount the new filesystem at boot time.

9.5.2 Creating LVM-managed partitions and filesystems

Logical Volume Manager (LVM) is widely used tool for managing storage in Linux environment. LVM allows combining multiple physical devices into a form of logical partition and provides capability to resize logical partitions. Certain filesystem types also allow on-line resizing. The combination of LUN discovery (provided by XIV Host Attachment Kit), LVM and filesystem capable of on-line resizing provides truly enterprise storage management solution for Linux operating environment. The following example demonstrates the capability of adding storage and increasing filesystem size on demand.

By default, LVM will not recognize device names starting with “/dev/mapper/mpath”. The LVM configuration file `/etc/lvm/lvm.conf` needs to be modified to allow LVM to use multipathed devices.

To verify what physical devices are already LVM-enabled and configured, use the **pvscan** command as shown in Example 9-16.

Example 9-16 Listing configured LVM physical devices

```
# pvscan
  PV /dev/hda2   VG VolGroup00   lvm2 [74.41 GB / 0    free]
  Total: 1 [74.41 GB] / in use: 1 [74.41 GB] / in no VG: 0 [0   ]
```

The new filtering rule for LVM should now exclude the existing device `/dev/hda2`. The default filter for devices that LVM would recognize is `filter = [ "a/.*/" ]`. This rule accepts all block devices but not XIV-specific and multipathed devices. To support the existing device and the new XIV devices, we need to change the LVM filter setting as follows:

```
filter = [ "a|^/dev/mapper/mpath.*$|", "a|1XIV.*|", "a|^/dev/hd.*$|", "r|.*|" ]
```

The new rule would make LVM to recognize multipathed devices (`^/dev/mapper/mpath.*$`), XIV specific devices (`1XIV.*`) as well as the existing device `/dev/hda2` (`^/dev/hd.*$`). All other device types will be rejected (`r|.*|`).

The filter setting in your environment can be different to allow LVM management of other device types. For instance, if your system boots from internal SCSI disk and some of its partitions are LVM managed, you will need an extra (^/dev/sd.*$) rule in you filter to make LVM recognize standard non-multipathed SCSI devices.

The following line needs to be added to /etc/lvm/lvm.conf

```
types = [ "device-mapper", 253 ]
```

This rule lists a pair of additional acceptable block device types found in /proc/devices

Create a backup copy of your /etc/lvm/lvm.conf file before making changes, then change the file as just described. To verify the changes, run the command shown in Example 9-17.

Example 9-17 Verifying LVM filter and types settings

```
# lvm dumpconfig | egrep 'filter|types'
      filter=["a|^/dev/mapper/mpath.*$|", "a|1XIV.*|", "a|^/dev/hd.*$|", "r|.*|"]
      types=["device-mapper", 253]
```

To verify that multipathed devices are now being recognized by LVM, use the **vgscan** command as shown in Example 9-18.

Example 9-18 Multipath devices visible by LVM

```
# vgscan -vv
      Setting global/locking_type to 1
      File-based locking selected.
      Setting global/locking_dir to /var/lock/lvm
      Locking /var/lock/lvm/P_global WB
    Wiping cache of LVM-capable devices
    Wiping internal VG cache
    Finding all volume groups
      /dev/hda: size is 156312576 sectors
      /dev/hda1: size is 208782 sectors
      /dev/hda1: size is 208782 sectors
      /dev/hda1: No label detected
      /dev/hda2: size is 156103605 sectors
      /dev/hda2: size is 156103605 sectors
      /dev/hda2: lvm2 label detected
      /dev/mapper/mpath0: size is 33554432 sectors
      /dev/mapper/mpath0: size is 33554432 sectors
      /dev/mapper/mpath0: No label detected
      ...
```

For a more detailed output of the **vgscan** command, use the '-vvv' option.

The next step is to create LVM-type partitions on the multipathed XIV devices. All devices will have to be configured as illustrated in Example 9-19.

Example 9-19 Preparing multipathed device to be used with LVM

```
[/]# fdisk  /dev/mapper/mpath4
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel
Building a new DOS disklabel. Changes will remain in memory only,
until you decide to write them. After that, of course, the previous
content won't be recoverable.
```

```
The number of cylinders for this disk is set to 2088.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)
Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-2088, default 1):
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-2088, default 2088):
Using default value 2088

Command (m for help): t
Selected partition 1
Hex code (type L to list codes): 8e
Changed system type of partition 1 to 8e (Linux LVM)

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
WARNING: Re-reading the partition table failed with error 22: Invalid argument.
The kernel still uses the old table.
The new table will be used at the next reboot.
Syncing disks.
[/]# partprobe -s /dev/mapper/mpath4
/dev/mapper/mpath4: msdos partitions 1
[/]# fdisk -l /dev/mapper/mpath4
Disk /dev/mapper/mpath4: 17.1 GB, 17179869184 bytes
255 heads, 63 sectors/track, 2088 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
              Device Boot      Start         End      Blocks   Id  System
/dev/mapper/mpath4p1               1        2088    16771828+  8e  Linux LVM
[/]# pvcreate /dev/mapper/mpath4p1
  Physical volume "/dev/mapper/mpath4p1" successfully created
```

Before a filesystem can be created on LVM-managed physical volumes (PV), a volumegroup (VG) and logical volume (LV) need to be created using space available on the physical volumes. Use the commands shown in Example 9-20.

Example 9-20 Creating VG data_vg and LV data_lv

```
[/]# pvscan
  PV /dev/hda2              VG VolGroup00      lvm2 [74.41 GB / 0    free]
  PV /dev/mapper/mpath3p1                      lvm2 [15.99 GB]
  PV /dev/mapper/mpath2p1                      lvm2 [15.99 GB]
  PV /dev/mapper/mpath4p1                      lvm2 [15.99 GB]
  PV /dev/mapper/mpath5p1                      lvm2 [15.99 GB]
  Total: 5 [138.39 GB] / in use: 1 [74.41 GB] / in no VG: 4 [63.98 GB]
```

```
[/]# vgcreate data_vg /dev/mapper/mpath2p1 /dev/mapper/mpath3p1 \
/dev/mapper/mpath4p1 /dev/mapper/mpath5p1
  Volume group "data_vg" successfully created
[root@orcakpvhd97 lvm]# vgdisplay data_vg
  --- Volume group ---
  VG Name               data_vg
  System ID
  Format                lvm2
  Metadata Areas        4
  Metadata Sequence No  1
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                0
  Open LV               0
  Max PV                0
  Cur PV                4
  Act PV                4
  VG Size               63.97 GB
  PE Size               4.00 MB
  Total PE              16376
  Alloc PE / Size       0 / 0
  Free  PE / Size       16376 / 63.97 GB
  VG UUID               Ms7Mm6-XryL-9upe-73O1-iBkt-eMyZ-6T8gqO

[/]# lvcreate -L 63G data_vg -n data_lv
[/]# lvscan
  ACTIVE            '/dev/data_vg/data_lv' [63.00 GB] inherit
  ACTIVE            '/dev/VolGroupOO/LogVolOO' [72.47 GB] inherit
  ACTIVE            '/dev/VolGroupOO/LogVolO1' [1.94 GB] inherit
[/]# lvdisplay /dev/data_vg/data_lv
  --- Logical volume ---
  LV Name                /dev/data_vg/data_lv
  VG Name                data_vg
  LV UUID                i3Kakx-6GUT-dr5G-1AdP-cqYv-vz52-sd9Gzr
  LV Write Access        read/write
  LV Status              available
  # open                 0
  LV Size                63.00 GB
  Current LE             16128
  Segments               4
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     256
  Block device           253:13
```

The filesystem can now be created using the newly created logical volume `data_lv`. The **lvdisplay** command output shows that 63 GB of space is available on this logical volume. In our example we will be using the `ext3` filesystem type because this type of a filesystem allows on-line resizing. See the illustration shown in Example 9-21.

Example 9-21 Creating and mounting ext3 filesystem

```
[/]# mkfs -t ext3 /dev/data_vg/data_lv
mke2fs 1.39 (29-May-2006)
```

```
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
8257536 inodes, 16515072 blocks
825753 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=0
504 block groups
32768 blocks per group, 32768 fragments per group
16384 inodes per group
Superblock backups stored on blocks:
  32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
  4096000, 7962624, 11239424

Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 39 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
[/]# fsck /dev/data_vg/data_lv
fsck 1.39 (29-May-2006)
e2fsck 1.39 (29-May-2006)
/dev/data_vg/data_lv: clean, 11/8257536 files, 305189/16515072 blocks
[/]# mkdir /xivfs
[/]# mount /dev/data_vg/data_lv /xivfs
[/]# df -m /xivfs
Filesystem           1M-blocks      Used Available Use% Mounted on
/dev/mapper/data_vg-data_lv
                         63500       180     60095   1% /xivfs
```

The newly created ext3 filesystem is mounted on /xivfs and has approximately 60GB of available space. The filesystem is ready to accept client data.

In our example we write some arbitrary data onto the filesystem to use all the space available and create an artificial “out of space” condition. After the filesystem utilization reaches 100%, an additional XIV volume is mapped to the server and initialized. VG and LV are then expanded to use the available space on that volume. The last step of the process is the on-line filesystem resizing to eliminate the “out of space” condition. Refer to Example 9-22.

Example 9-22 Filesystem “out of space” condition

```
# df
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/mapper/VolGroup00-LogVol00
                      73608360   3397416  66411496   5% /
/dev/hda1               101086     18785     77082  20% /boot
tmpfs                  1037708         0   1037708   0% /dev/shm
/dev/mapper/mpath1p1  16508572    176244  15493740   2% /tempmount
/dev/mapper/data_vg-data_lv
                      65023804  63960868         0 100% /xivfs
```

Because the filesystem has no blocks available, any write attempts to this filesystem will fail. An additional XIV volume is mapped and discovered by the system as described in Example 9-23.

The discovery of the new XIV LUN is done using Host Attachment Kit command:

```
/opt/xiv/host_attach/bin/host_attach_fc
```

Example 9-23 Discovering the new LUN

```
[/]# /opt/xiv/host_attach/bin/xiv_devlist
XIV devices
===========
Device               Vol Name     XIV Host     Size    Paths XIV ID   Vol ID
-------------------------------------------------------------------------------
mpath2               orcah...l_10 orcakpvhd97  17.2GB  4/4   MN00021  48
mpath1               orcah...l_09 orcakpvhd97  17.2GB  4/4   MN00021  47
mpath4               orcah...l_12 orcakpvhd97  17.2GB  4/4   MN00021  50
mpath0               orcah...l_08 orcakpvhd97  17.2GB  4/4   MN00021  46
mpath3               orcah...l_11 orcakpvhd97  17.2GB  4/4   MN00021  49
mpath5               orcah...l_13 orcakpvhd97  17.2GB  4/4   MN00021  51

Non-XIV devices
===============
Device               Size    Paths
-----------------------------------

(VIX volume mapping is performed)

[/]# /opt/xiv/host_attach/bin/xiv_fc_admin --rescan

[/]# /opt/xiv/host_attach/bin/xiv_devlist
XIV devices
===========
Device               Vol Name     XIV Host     Size    Paths XIV ID   Vol ID
-------------------------------------------------------------------------------
mpath2               orcah...l_10 orcakpvhd97  17.2GB  4/4   MN00021  48
mpath10              orcah...l_02 orcakpvhd97  17.2GB  4/4   MN00021  40
mpath1               orcah...l_09 orcakpvhd97  17.2GB  4/4   MN00021  47
mpath4               orcah...l_12 orcakpvhd97  17.2GB  4/4   MN00021  50
mpath0               orcah...l_08 orcakpvhd97  17.2GB  4/4   MN00021  46
mpath3               orcah...l_11 orcakpvhd97  17.2GB  4/4   MN00021  49
mpath5               orcah...l_13 orcakpvhd97  17.2GB  4/4   MN00021  51

Non-XIV devices
===============
Device               Size    Paths
-----------------------------------
```

The new XIV LUN is discovered by the system. `/dev/mapper/mpath10` is the device name assigned to the newly discovered LUN. Initialization of the new volume (creating partition, assigning partition type is done as described in Example 9-24. After the new LVM physical volume is created, VG is expanded to use the space available on that volume (an additional 16.9GB).

Example 9-24 Using the new LUN

```
[/]# pvcreate /dev/mapper/mpath10p1
  Physical volume "/dev/mapper/mpath10p1" successfully created

[/]# pvscan
  PV /dev/mapper/mpath2p1    VG data_vg         lvm2 [15.99 GB / 0    free]
  PV /dev/mapper/mpath3p1    VG data_vg         lvm2 [15.99 GB / 0    free]
  PV /dev/mapper/mpath4p1    VG data_vg         lvm2 [15.99 GB / 0    free]
  PV /dev/mapper/mpath5p1    VG data_vg         lvm2 [15.99 GB / 992.00 MB free]
  PV /dev/hda2               VG VolGroup00      lvm2 [74.41 GB / 0    free]
  PV /dev/mapper/mpath10p1                      lvm2 [15.99 GB]
  Total: 6 [154.37 GB] / in use: 5 [138.38 GB] / in no VG: 1 [15.99 GB]

[/]# vgextend data_vg /dev/mapper/mpath10p1
  Volume group "data_vg" successfully extended
# lvscan
  ACTIVE            '/dev/data_vg/data_lv' [63.00 GB] inherit
  ACTIVE            '/dev/VolGroup00/LogVol00' [72.47 GB] inherit
  ACTIVE            '/dev/VolGroup00/LogVol01' [1.94 GB] inherit
[/]# lvdisplay /dev/data_vg/data_lv
  --- Logical volume ---
  LV Name                /dev/data_vg/data_lv
  VG Name                data_vg
  LV UUID                i3Kakx-6GUT-dr5G-1AdP-cqYv-vz52-sd9Gzr
  LV Write Access        read/write
  LV Status              available
  # open                 1
  LV Size                63.00 GB
  Current LE             16128
  Segments               4
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     256
  Block device           253:13

# vgdisplay data_vg
  --- Volume group ---
  VG Name               data_vg
  System ID
  Format                lvm2
  Metadata Areas        5
  Metadata Sequence No  3
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                1
  Open LV               1
  Max PV                0
  Cur PV                5
  Act PV                5
  VG Size               79.96 GB
  PE Size               4.00 MB
  Total PE              20470
  Alloc PE / Size       16128 / 63.00 GB
  Free  PE / Size       4342 / 16.96 GB
```

```
  VG UUID               Ms7Mm6-XryL-9upe-7301-iBkt-eMyZ-6T8gqO

[/]# lvextend  -L +16G /dev/data_vg/data_lv
  Extending logical volume data_lv to 79.00 GB
  Logical volume data_lv successfully resized

[/]# resize2fs /dev/mapper/data_vg-data_lv
resize2fs 1.39 (29-May-2006)
Filesystem at /dev/mapper/data_vg-data_lv is mounted on /xivfs; on-line resizing
required
Performing an on-line resize of /dev/mapper/data_vg-data_lv to 20709376 (4k)
blocks.
The filesystem on /dev/mapper/data_vg-data_lv is now 20709376 blocks long.

[/]# df
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/mapper/VolGroup00-LogVol00
                      73608360   3397416  66411496   5% /
/dev/hda1               101086     18785     77082  20% /boot
tmpfs                  1037708         0   1037708   0% /dev/shm
/dev/mapper/mpath1p1  16508572    176244  15493740   2% /tempmount
/dev/mapper/data_vg-data_lv
                      81537776  63964892  13431216  83% /xivfs
```

After the `/xivfs` filesystem was resized with the `resize2fs` command, its utilization dropped from 100% to 83%. Note that the filesystem remained mounted throughout the process of discovering the new LUN, expanding VG and LV, and on-line resizing of the filesystem itself.

10

VMware ESX host connectivity

This chapter explains OS-specific considerations for host connectivity and describes the host attachment related tasks for ESX 3.5.

10.1 Attaching an ESX 3.5 host to XIV

This section describes the attachment of ESX 3.5 based hosts to the XIV Storage System. It provides specific instructions for Fibre Channel (FC) and Internet Small Computer System Interface (iSCSI) connections. All the information in this section relates to ESX 3.5 (and not other versions of ESX) unless otherwise specified.

ESX is part of VMware Virtual Infrastructure 3 (VI3) which comprises a number of products. ESX is the core product; the other companion products enhance or extent ESX. We only discuss ESX in this chapter.

The procedures and instructions given here are based on code that was available at the time of writing this book. For the latest information about XIV OS support, refer to the System Storage Interoperability Center (SSIC) at:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

Also, refer to the XIV Storage System *Host System Attachment Guide for VMware-Installation Guide*, which is available at:

http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp

10.2 Prerequisites

To successfully attach an ESX host to XIV and assign storage, a number of prerequisites need to be met. Here is a generic list, however, your environment might have additional requirements:

- Complete the cabling.
- Configure the zoning.
- Install any service packs and/or updates if required.
- Create volumes to be assigned to the host.

Supported versions of ESX

At the time of writing the following versions of ESX are supported:

- ESX 4.0
- ESX 3.5
- ESX 3.0

Supported FC HBAs

Supported FC HBAs are available from IBM, Emulex, and QLogic. Further details on driver versions are available from SSIC at the following Web site:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

Unless otherwise noted in SSIC, use any supported driver and firmware by the HBA vendors (the latest versions are always preferred). For HBAs in Sun systems, use Sun branded HBAs and Sun ready HBAs only.

SAN boot

The following versions of ESX support SAN boot:

- ESX 4.0
- ESX 3.5

Multi-path support

VMware provides its own multipathing I/O driver for ESX. No additional drivers or software are required. As such, the Host Attachment Kit only provides documentation, and no software installation is required.

10.3 ESX host FC configuration

This section describes attaching ESX hosts through FC and provides detailed descriptions and installation instructions for the various software components required.

Installing HBA drivers

ESX includes drivers for all the HBAs that it supports. VMware strictly controls driver policy, and only drivers provided by VMware should be used. Any driver updates are normally included in service/update packs.

Scanning for new LUNs

Before you can scan for new LUNs your host needs to be added and configured on the XIV (see Chapter 6., “Host connectivity” on page 183 for information on how to do this).

ESX hosts that access the same shared LUNs should be grouped in a cluster (XIV cluster) and the LUNs assigned to the cluster. Refer to Figure 10-1 and Figure 10-2 for how this might typically be set up.

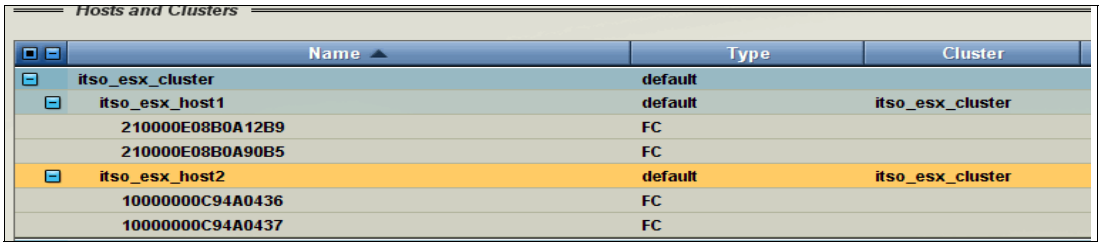


Figure 10-1 ESX host cluster setup in XIV GUI

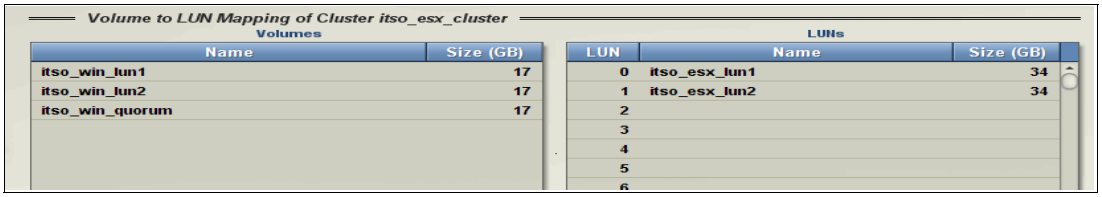


Figure 10-2 ESX LUN mapping to the cluster

To scan and configure new LUNs follow these instructions:

1. After the host definition and LUN mappings have been completed in the XIV Storage System, go to the **Configuration** tab for your host, and select **Storage Adapters** as shown in Figure 10-3.

   Here you can see vmhba2 highlighted but a rescan will scan across all adapters. The adapter numbers might be enumerated differently on the different hosts; this is not an issue.

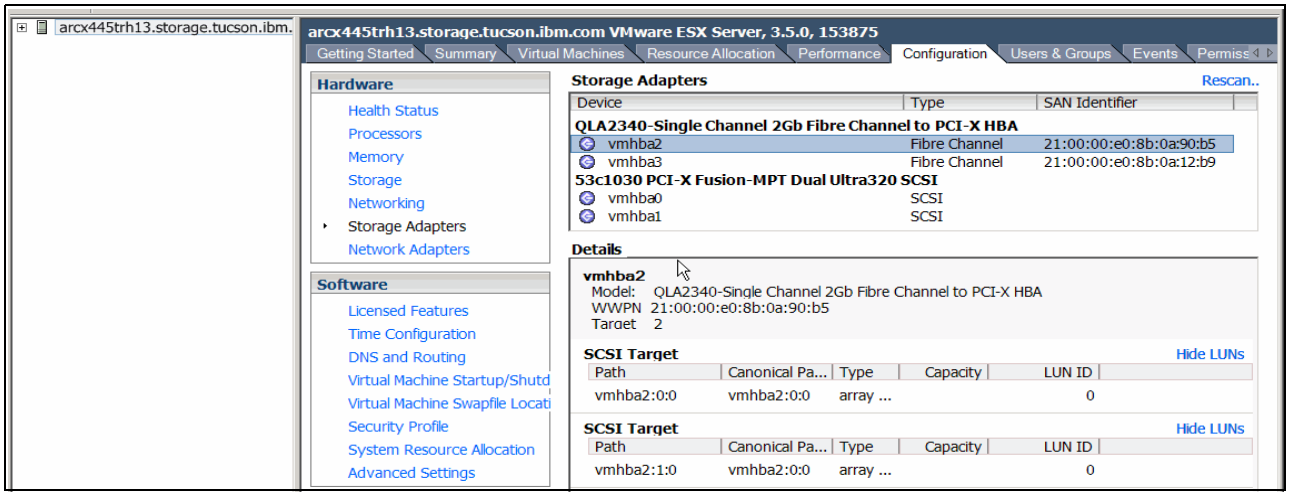


Figure 10-3 Select Storage Adapters

2. Select **Rescan** and then **OK** to scan for new storage devices as shown in Figure 10-4.

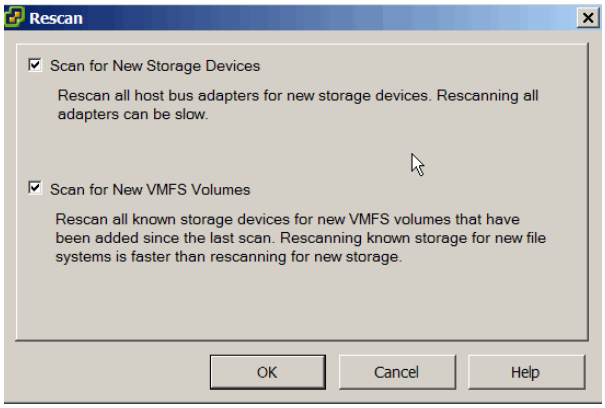


Figure 10-4 Rescan for New Storage Devices

3. The new LUNs assigned will appear in the *Details* pane as depicted in Figure 10-5.

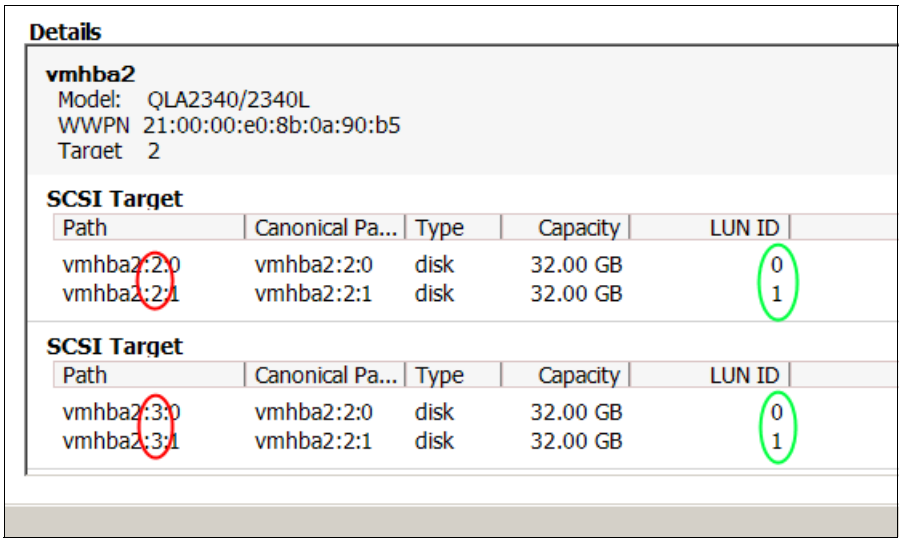


Figure 10-5 FC discovered LUNs on vmhba2

Here, you observe that controller vmhba2 can see two LUNs (LUN 0 and LUN 1) circled in green and they are visible on two targets (2 and 3) circled in red. The other controllers in the host will show the same path and LUN information.

For detailed information about how to now use these LUNs with virtual machines, refer to the VMware guides, available at the following Web sites:

http://www.vmware.com/pdf/vi3_35/esx_3/r35u2/vi3_35_25_u2_admin_guide.pdf
http://www.vmware.com/pdf/vi3_35/esx_3/r35u2/vi3_35_25_u2_3_server_config.pdf

Assigning paths

The XIV is an active/active storage system and therefore it can serve I/Os to all LUNs using every available path. However, the driver with ESX 3.5 cannot perform the same function and by default cannot fully load balance. It is possible to partially overcome this limitation by setting the correct pathing policy and distributing the IO load over the available HBAs and XIV ports. This could be referred to as 'manually' load balancing. To achieve this, follow the instructions below.

1. The pathing policy in ESX 3.5 can be set to either **Most Recently Used** (MRU) or **Fixed**. When accessing storage on the XIV the correct policy is **Fixed**. In the VMware Infrastructure Client select the server then **Configuration** tab → **Storage**. Refer to Figure 10-6.

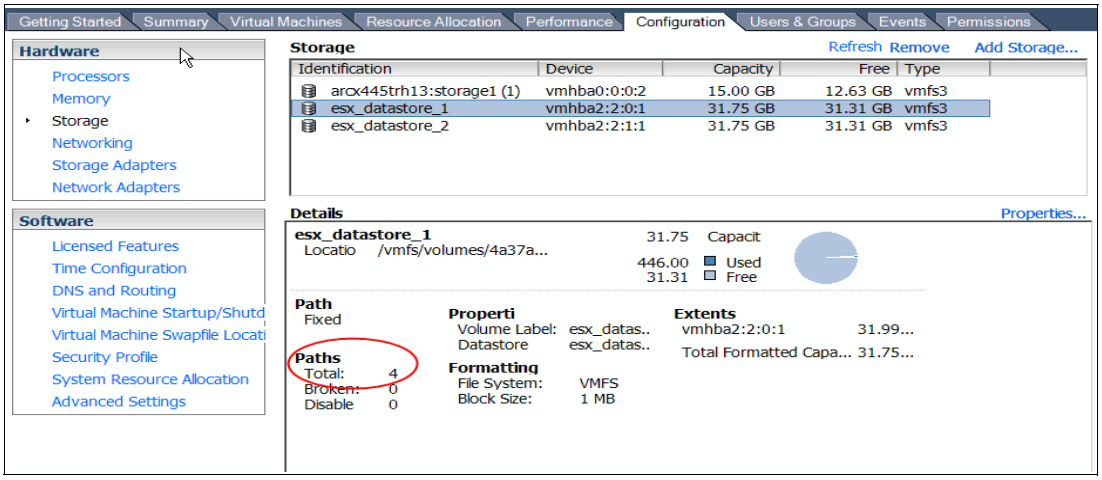


Figure 10-6 Storage paths

You can see the LUN highlighted (esx_datastore_1) and the number of paths is 4 (circled in red).

2. Select **Properties** to bring up further details about the paths, as shown in Figure 10-7.

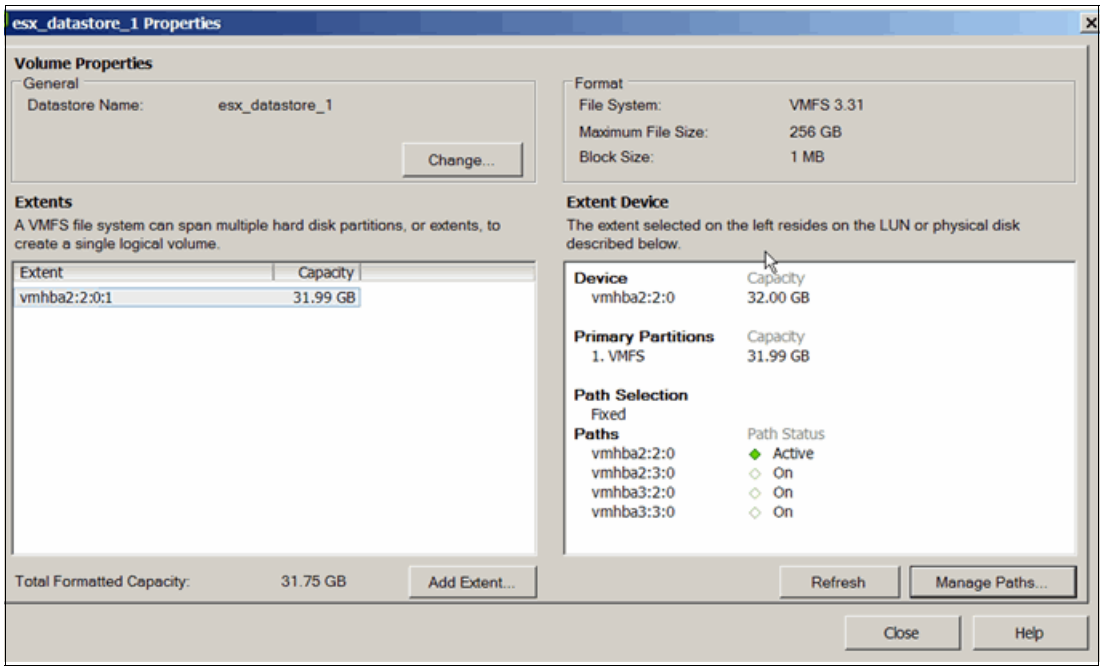


Figure 10-7 Storage path details

You can see here that the active path is vmhba2:2:0.

3. To change the current path, select **Manage Paths** (refer to Figure 10-8). The pathing policy should be **Fixed;** if it is not, then select **Change** in the *Policy* pane and change it to **Fixed**.

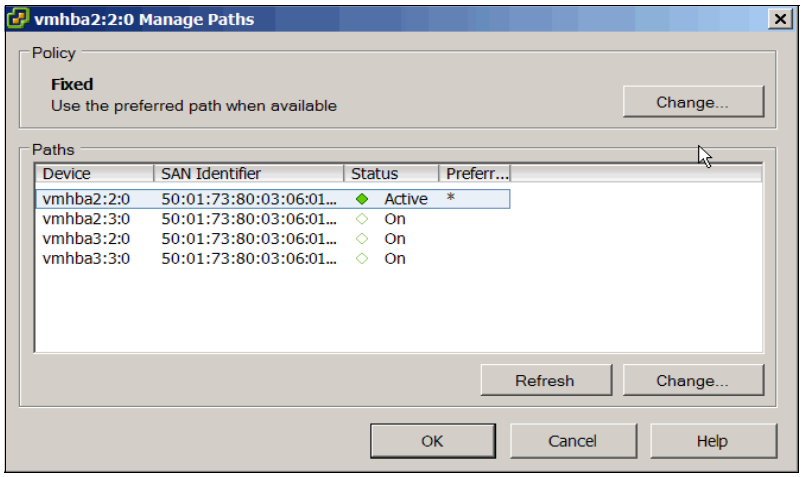


Figure 10-8 Change paths

4. To manually load balance, highlight the preferred path and select **Change** in the **Paths** pane. Then, assign an HBA and target port to the LUN. Refer to Figure 10-9, Figure 10-10, and Figure 10-11.

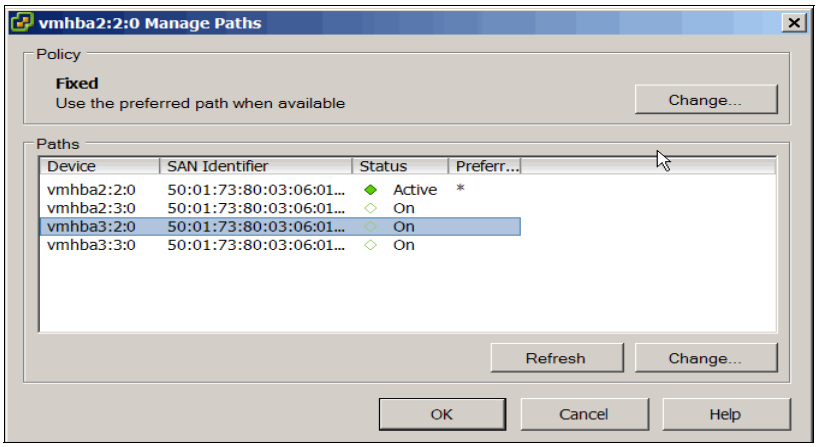


Figure 10-9 Change to new path

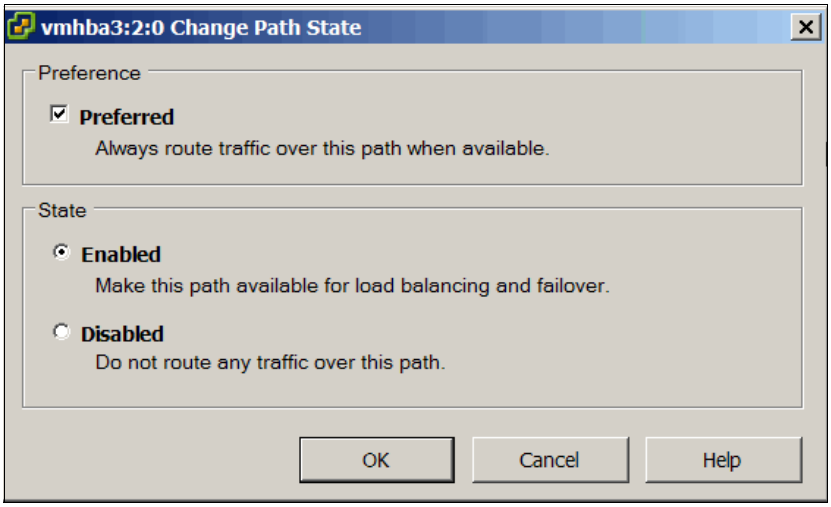


Figure 10-10 Set preferred

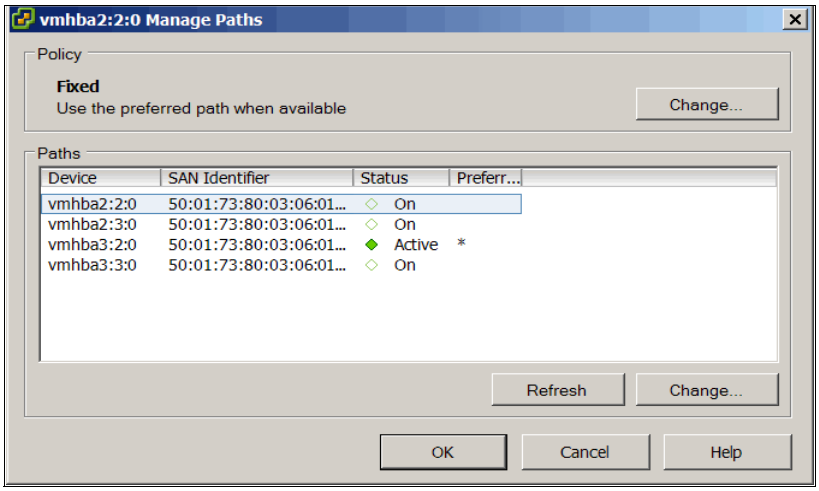


Figure 10-11 New preferred path set

5. Repeat steps 1-4 to manually balance IO across the HBAs and XIV target ports. Due to the manual nature of this configuration, it will need to be reviewed over time.

Important: When setting paths, all hosts within the same datacenter (also known as a farm) should access each individual LUN via the same XIV ports. For more information regarding setting up paths for ESX, refer to the VMware ESX documentation.

Example 10-1 and Example 10-2 show the results of manually configuring two LUNs on separate preferred paths on two ESX hosts. Only two LUNs are shown for clarity, but this can be applied to all LUNs assigned to the hosts in the ESX datacenter.

Example 10-1 ESX Host 1 preferred path

```
[root@arcx445trh13 root]# esxcfg-mpath -l
Disk vmhba0:0:0 /dev/sda (34715MB) has 1 paths and policy of Fixed
 Local 1:3.0 vmhba0:0:0 On active preferred

Disk vmhba2:2:0 /dev/sdb (32768MB) has 4 paths and policy of Fixed
 FC 5:4.0 210000e08b0a90b5<->5001738003060140 vmhba2:2:0 On active preferred
 FC 5:4.0 210000e08b0a90b5<->5001738003060150 vmhba2:3:0 On
 FC 7:3.0 210000e08b0a12b9<->5001738003060140 vmhba3:2:0 On
 FC 7:3.0 210000e08b0a12b9<->5001738003060150 vmhba3:3:0 On

Disk vmhba2:2:1 /dev/sdc (32768MB) has 4 paths and policy of Fixed
 FC 5:4.0 210000e08b0a90b5<->5001738003060140 vmhba2:2:1 On
 FC 5:4.0 210000e08b0a90b5<->5001738003060150 vmhba2:3:1 On
 FC 7:3.0 210000e08b0a12b9<->5001738003060140 vmhba3:2:1 On
 FC 7:3.0 210000e08b0a12b9<->5001738003060150 vmhba3:3:1 On active preferred
```

Example 10-2 ESX host 2 preferred path

```
[root@arcx445bvkf5 root]# esxcfg-mpath -l
Disk vmhba0:0:0 /dev/sda (34715MB) has 1 paths and policy of Fixed
 Local 1:3.0 vmhba0:0:0 On active preferred

Disk vmhba4:0:0 /dev/sdb (32768MB) has 4 paths and policy of Fixed
 FC 7:3.0 10000000c94a0436<->5001738003060140 vmhba4:0:0 On active preferred
 FC 7:3.0 10000000c94a0436<->5001738003060150 vmhba4:1:0 On
 FC 7:3.1 10000000c94a0437<->5001738003060140 vmhba5:0:0 On
 FC 7:3.1 10000000c94a0437<->5001738003060150 vmhba5:1:0 On

Disk vmhba4:0:1 /dev/sdc (32768MB) has 4 paths and policy of Fixed
 FC 7:3.0 10000000c94a0436<->5001738003060140 vmhba4:0:1 On
 FC 7:3.0 10000000c94a0436<->5001738003060150 vmhba4:1:1 On
 FC 7:3.1 10000000c94a0437<->5001738003060140 vmhba5:0:1 On
 FC 7:3.1 10000000c94a0437<->5001738003060150 vmhba5:1:1 On active preferred
```

11

VIOS clients connectivity

This chapter explains connectivity to XIV for Virtual I/O Server (VIOS) clients, including AIX, Linux on Power and in particular, IBM i. VIOS is a component of Power VM that provides the ability for LPARs (VIOS clients) to share resources.

11.1 IBM Power VM overview

IBM Power VM is a special software appliance tied to IBM POWER Systems, that is, the converged IBM i and IBM p server platforms. It is licensed on a POWER system processor basis.

IBM PowerVM™ is a virtualization technology for AIX, IBM i, and Linux environments on IBM POWER processor-based systems. IBM Power Systems™ servers coupled with PowerVM technology are designed to help clients build a dynamic infrastructure, reducing costs, managing risk, and improving service levels.

PowerVM offers a secure virtualization environment that offers the following major features and benefits:

- Consolidates diverse sets of applications built for multiple operating systems on a single server: AIX, IBM i, and Linux
- Virtualizes processor, memory, and I/O resources to increase asset utilization and reduce infrastructure costs
- Dynamically adjusts server capability to meet changing workload demands
- Moves running workloads between servers to maximize availability and avoid planned downtime

11.1.1 Virtual I/O Server (VIOS)

Virtual I/O Server (VIOS) is virtualization software that runs in a separate partition of your POWER system. Its purpose is to provide virtual storage and networking resources to one or more client partitions.

The Virtual I/O Server owns the physical I/O resources like Ethernet and SCSI/FC adapters. It virtualizes those resources for its client LPARs to share them remotely using the built-in hypervisor services. These client LPARs can be quickly created, typically owning only real memory and shares of CPUs without any physical disks or physical Ethernet adapters.

Virtual SCSI support allows VIOS client partitions to share disk storage that is physically assigned to the Virtual I/O Server logical partition. This virtual SCSI support of VIOS is used to make storage devices such as XIV that do not support the IBM i proprietary 520-byte/sectors format available to IBM i clients of VIOS.

VIOS owns the physical adapters such as the Fiber Channel storage adapters connected to the IBM XIV Storage System. The LUNs of the physical storage devices seen by VIOS are mapped to VIOS virtual SCSI (VSCSI) server adapters created as part of its partition profile.

The client partition with its corresponding virtual SCSI client adapters defined in its partition profile connects to the VIOS virtual SCSI server adapters via the hypervisor with VIOS performing SCSI emulation and acting as the SCSI target for IBM i. Figure 11-1 shows an example of the Virtual I/O Server owning the physical disk devices and its virtual SCSI connections to two client partitions.

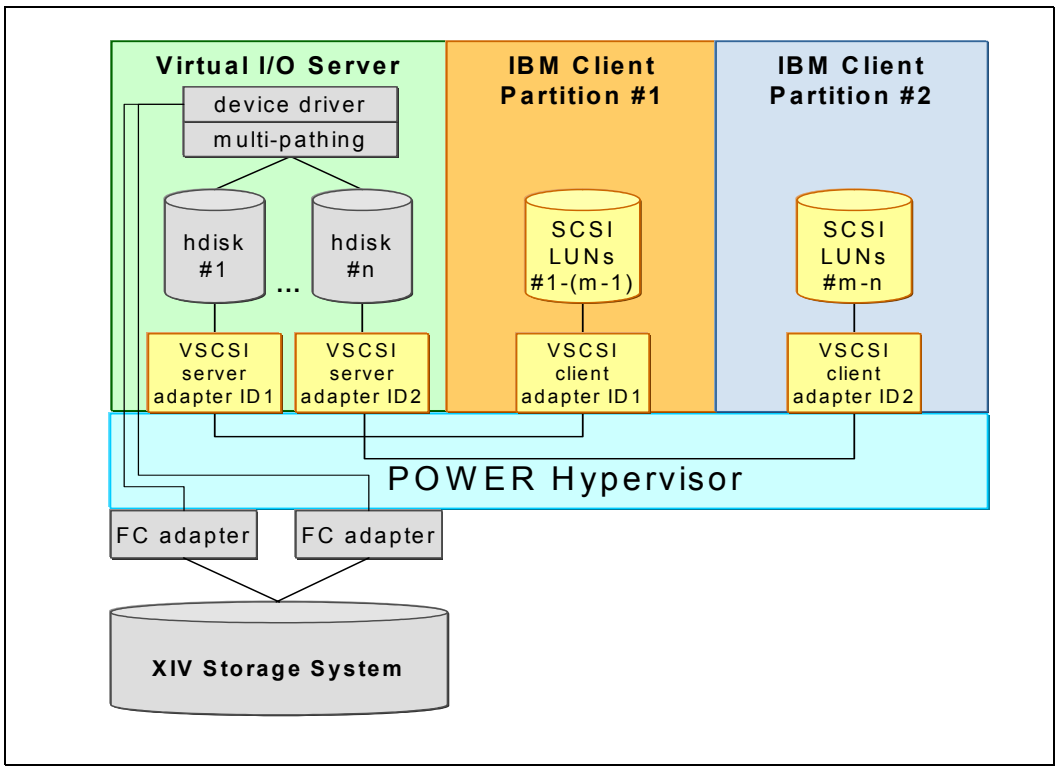


Figure 11-1 VIOS Virtual SCSI Support

11.1.2 Node Port ID Virtualization (NPIV)

The VIOS technology has been enhanced to boost the flexibility of Power Systems servers with support for Node Port ID Virtualization (NPIV). NPIV simplifies the management and improves performance of Fibre Channel SAN environments by standardizing a method for Fibre Channel ports to virtualize a physical node port ID into multiple virtual node port IDs.

VIOS takes advantage of this feature and can export the virtual node port IDs to multiple VIOS clients. The VIOS clients see this node port ID and can discover devices just as if the physical port was attached to the VIOS client. VIOS does not do any device discovery on ports using NPIV. Thus there are no devices shown on VIOS connected to NPIV adapters. The discovery is left for the VIOS client and all the devices found during discovery are seen only by the client. This allows the VIOS client to use FC SAN storage specific multipathing software on the client to discover and manage devices.

Figure 11-2 shows a managed system configured to use NPIV, running two VIOS partitions each with one physical Fibre Channel card. Each VIOS partition provides virtual Fibre Channel adapters to the VIOS clients. For increased serviceability, you can use MPIO in the AIX client.

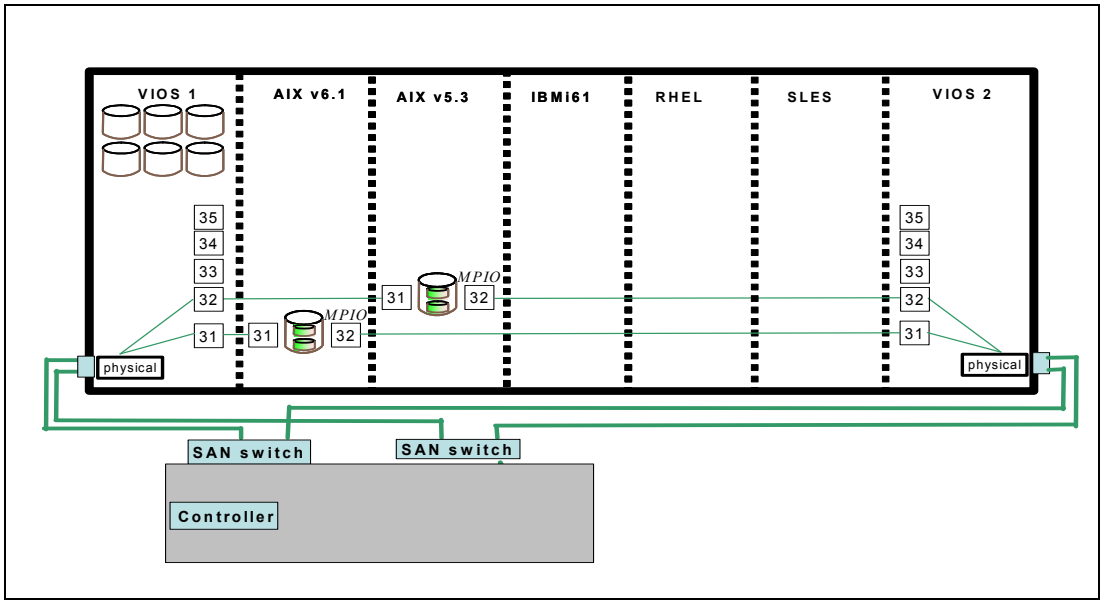


Figure 11-2 Virtual I/O Server Partitions with NPIV

Further information regarding Power VM virtualization management can be found in the IBM Redbooks publication, *IBM PowerVM Virtualization Managing and Monitoring*, SG24-7590.

11.2 Power VM client connectivity to XIV

This section discusses the configuration for VIOS clients (Virtual SCSI or NPIV) on VIOS server that is attached to the XIV storage.

11.2.1 Planning for VIOS

PowerVM comes shipped with a VIOS installation DVD and an authorization code that needs to be entered on the HMC before a VIOS partition can be created.

Note: While PowerVM and VIOS themselves are supported on both POWER5™ and POWER6® systems, IBM i, being a client of VIOS, is supported only on POWER6 systems.

Minimum system hardware and software requirements

The minimum system hardware and software requirements are as follows:

- **For Virtual SCSI clients and IBM Power VM version 2.1.1**:
  - XIV version 10.0.1.c or later with Host Attachment Kit = 1.1.0.1 or later
  - The use of SAN switches (FC direct attach is not supported)
  - VIOS Client support includes:
    - AIX clients v5.3 TL10 or later and v6.1 TL3 or later
    - Linux on Power (SUSE Linux Enterprise Server 9, 10 & 11), (Red Hat Enterprise Linux version 5.3)
    - System i® V6R1 (requires Power6 hardware)
- **For NPIV with IBM Power VM version 2.1.1**:
  - XIV version 10.0.1.c or later
  - The use of SAN switches that support NPIV emulation (FC direct attach is not supported)
  - NPIV support requires Power6 hardware
  - 8 Gigabit Dual Port Fibre Channel adapter, (feature code 5735)
  - NPIV client support includes:
    - AIX v5.3 TL10 or later and v6.1 TL3 or later
    - Linux on Power (SuSE Linux Enterprise Server 11)
  - XIV Host Attachment Kit = 1.1.0.1 or later
- **For IBM i client with IBM Power VM version 2.1.1**:
  - IBM POWER Systems POWER6 server model (AIX = 5.3.10 and 6.1.3 (Blades))

    IBM Power 520 Express (8203-E4A) and IBM Power 550 Express (8204-E8A) are not supported.
  - System firmware 320_040_031 or later
  - HMC V7 R3.2.0 or later
  - IBM i V6R1 or later
  - XIV = 10.0.1.c or later with Host Attachment Kit = 1.1.0.1 or later

For up to date information regarding supported levels, refer to the XIV interoperability matrix or the System Storage Interoperability Center (SSIC) at:

http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

For a complete list of supported devices, refer to the VIOS datasheet at:

http://www.software.ibm.com/webapp/set2/sas/f/vios/documentation/datasheet.html

11.2.2 Switches and zoning

As best practice, we recommend connecting and zoning the switches as follows:

1. Spread VIOS host adapters or ports equally between all the XIV interface modules.
2. Spread VIOS host adapters or ports equally between the switches.

   An example of zoning is shown in Figure 11-3.

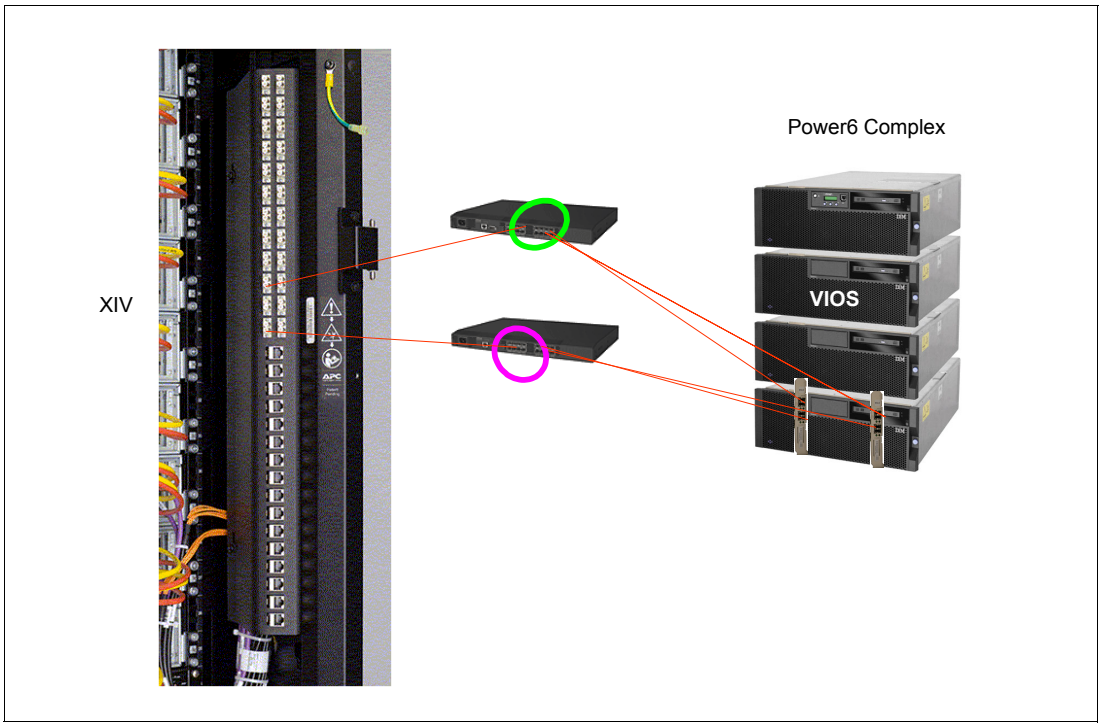


Figure 11-3 Zoning switches for IBM XIV connection

11.2.3 XIV-specific packages for VIOS

For VIOS and NPIV clients to recognize the disks mapped from the XIV Storage System, a specific fileset is required on the specific system. This fileset allows the specific device function and attributes for XIV. The fileset can be downloaded from:

`ftp://ftp.software.ibm.com/storage/XIV/`

XIV-specific packages are supported for both Virtual SCSI support for IBM XIV storage connections to AIX and Linux for Power clients and NPIV AIX clients connected to IBM XIV storage via VIO Servers:

- IBM XIV Host Attachment Kit version 1.1.0.1 or later for AIX
- No IBM XIV Host Attachment Kit is available for SLES 11 (at the time this publication was written, SLES 11 was the only pLinux OS supported with NPIV)

Installing the XIV-specific package for VIOS

To install the fileset, follow these steps:

1. Download or copy the downloaded fileset to your VIOS system.
2. From the VIOS prompt, execute the **`oem_setup_env`** command, which will place the padmin user into a non-restricted UNIX(R) root shell.
3. From the root prompt, change to the directory where your XIV package is located and execute the **`inutoc .`** command to create the table of contents file.
4. Use the AIX **`installp`** command or SMITTY (**smitty** → **Software Installation and Maintenance** → **Install and Update Software** → **Install Software**) to install the XIV disk package. Complete the parameters as shown in Example 11-1 and Figure 11-4.

Example 11-1 Manual installation

```
installp -aXY -d . disk.fcp.2810.1.1.0.1.bff
```

```
                                Install Software

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                                   [Entry Fields]
* INPUT device / directory for software                 .
* SOFTWARE to install                                   [+ 1.1.0.1  IBM 2810XI> +
  PREVIEW only? (install operation will NOT occur)      no                      +
  COMMIT software updates?                              yes                     +
  SAVE replaced files?                                  no                      +
  AUTOMATICALLY install requisite software?             yes                     +
  EXTEND file systems if space needed?                  yes                     +
  OVERWRITE same or newer versions?                     no                      +
  VERIFY install and check file sizes?                  no                      +
  Include corresponding LANGUAGE filesets?              yes                     +
  DETAILED output?                                      no                      +
  Process multiple volumes?                             yes                     +
  ACCEPT new license agreements?                        no                      +
[MORE...8]

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 11-4 smitty installation

Run the FC discovery command (**cfgmgr or cfgdev**) after the XIV specific package installs successfully. Running the **cfgmgr** procedure is illustrated in Example 11-2.

Example 11-2 FC discovery

```
The following command is run from the non-restricted UNIX(R) root shell
# cfgmgr -v

The following command is run from the VIOS padmin shell:
$ cfgdev
```

Now, when we list the disks, we see the correct number of disks assigned from the storage for all corresponding Virtual SCSI clients, and the disks are displayed as XIV disks, as shown in Example 11-3.

Example 11-3 XIV labeled FC disks

```
$ lsdev -type disk
hdisk0   Available SAS Disk Drive
hdisk1   Available SAS Disk Drive
hdisk5   Available IBM 2810XIV Fibre Channel Disk
hdisk6   Available IBM 2810XIV Fibre Channel Disk
hdisk7   Available IBM 2810XIV Fibre Channel Disk
```

Note: XIV Volumes should have already been created and mapped to VIO Server host connections.

Multipathing

Native multipath drivers are supported for both VIOS for IBM XIV storage connection to AIX and Linux for Power clients and NPIV clients connected to IBM XIV storage via VIOS.

However, at the time this publication was written, no multipath drivers were supported in VIOS for IBM XIV storage connectivity to an IBM i client.

With IBM XIV Storage connected to IBM i client via VIOS, it is possible to implement multipath so that a logical volume connects to VIOS via multiple *physical* host adapters (or ports) in VIOS. However, *virtual SCSI adapters* are used in single path. Refer to the IBM Redbooks Publication, *IBM i and Midrange External Storage*, SG24-7668, for more information about how to ensure redundancy of VIOS servers and consequently virtual SCSI adapters.

Note: Using IBM i multipathing to VIOS is currently not supported.

Best practice recommendations

The default algorithm is round_robin with a queue_depth=32.

The reservation policy on the hdisks should be set to no_reserve if Multipath I/O is being used. If dual VIO Servers (see 11.3, “Dual VIOS servers” on page 289)are being used the same reservation policy check needs to be done on the second VIO server.

Run the **`lsdev -dev hdisk# -attr`** command to list the attributes of the disk you choose for MPIO. See Figure 11-5 for output details.

```
$ lsdev -dev hdisk2 -attr
attribute       value                                    description                  user_settable

PCM             PCM/friend/2810xivpcm                    Path Control Module          False
PR_key_value                                             Persistent Reserve Key Value True
algorithm       round_robin                              Algorithm                    True
hcheck_cmd      inquiry                                  Health Check Command         True
hcheck_interval 60                                       Health Check Interval        True
hcheck_mode     nonactive                                Health Check Mode            True
lun_id          0x1000000000000                          Logical Unit Number ID       False
lun_reset_spt   yes                                      Support SCSI LUN reset       True
max_transfer    0x40000                                  Maximum TRANSFER Size        True
node_name       0x5001738000ca0000                       FC Node Name                 False
pvid            none                                     Physical volume identifier   False
q_type          simple                                   Queuing TYPE                 True
queue_depth     32                                       Queue DEPTH                  True
reserve_policy  no_reserve                               Reserve Policy               True
rw_timeout      30                                       READ/WRITE time out value    True
scsi_id         0x2d004e                                 SCSI ID                      False
unique_id       261120017380000CA0039072810XIV03IBMfcp   Unique device identifier     False
ww_name         0x5001738000ca0180                       FC World Wide Name           False
$
```

Figure 11-5 List disk attributes

Run the **`chdev`** command to change the reservation policy on the hdisk to no_reserve. See Example 11-4 for a complete command and parameters.

Example 11-4 chdev command

```
$ chdev -dev hdisk2 -attr reserve_policy=no_reserve
hdisk2 changed
```

11.3 Dual VIOS servers

For higher availability, it is recommended to deploy dual VIOS servers on your POWER6 hardware complex. For redundancy, the VIO servers should have their own dedicated physical resources. In addition, this setup depicted in Figure 11-6, allows for all clients, excluding IBM i, to have dual pathing serviced by multiple VIO Servers to the IBM XIV storage:

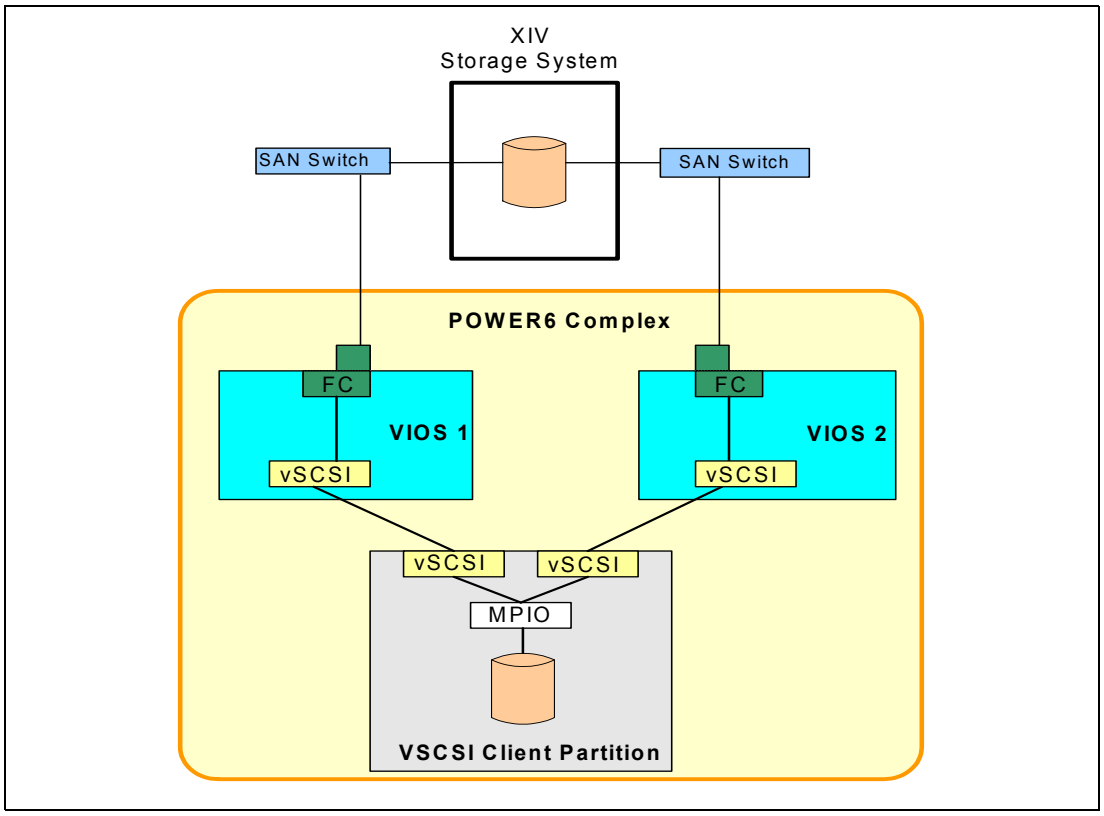


Figure 11-6 Dual VIO servers

Note: Multiple paths can be obtained from a single VIO Server, but dual VIO Servers provide for additional redundancy in case one server encounters a disaster or requires downtime for maintenance.

In addition, this setup allows for IBM i clients to utilize dual VIOS servers as a load balancing mechanism by allowing each VIO server to allocate access to a subset of the total volumes seen by the IBM i systems. These concepts are illustrated in Figure 11-7.

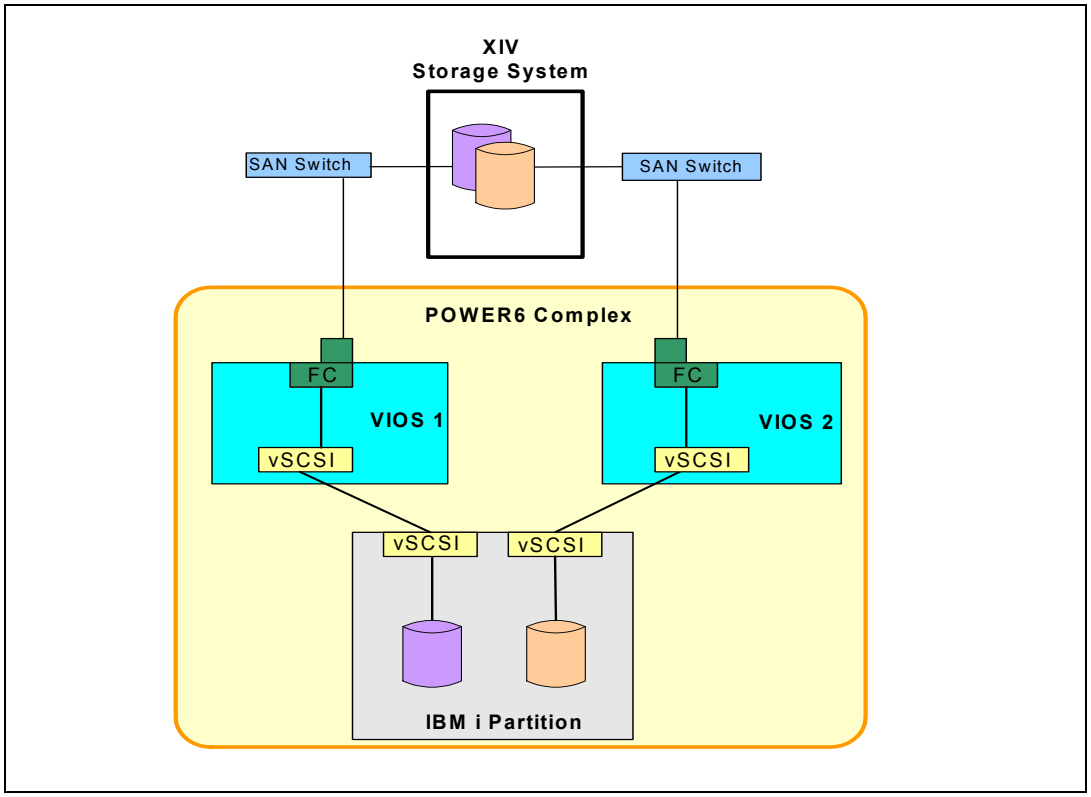


Figure 11-7 IBM i client load balancing between 2 VIO servers

An IBM i client partition in this environment has a dependency on VIOS: If the VIOS partition fails, IBM i on the client will lose contact with the virtualized XIV LUNs. The LUNs would also become unavailable if VIOS is brought down for scheduled maintenance or a release upgrade. To remove this dependency, two VIOS partitions can be used to simultaneously provide virtual storage to one or more IBM i client partitions.

The configuration for two VIOS partitions for the same client partition uses the same concepts as that for a single VIOS. In addition, a second virtual SCSI client adapter exists in the client LPAR, connected to a virtual SCSI server adapter in the second VIOS on the same Power server. A second set of LUNs of the same number and size is created on the same or a different XIV, and connected to the second VIOS. The host-side configuration of the second VIOS mimics that of the first host, with the same number of LUNs (hdisks), vtscsiX, and vhostX devices.

As a result, the client partition recognizes a second set of virtual disks of the same number and size. To achieve redundancy, adapter-level mirroring is used between the two sets of virtualized LUNs from the two hosts. Thus, if a VIOS partition fails or is taken down for maintenance, mirroring will be suspended, but the IBM i client will continue to operate. When the inactive VIOS is either recovered or restarted, mirroring can be resumed in IBM i.

Note that the dual-VIOS solution just described provides a level of redundancy by attaching two separate sets of XIV LUNs to the same IBM i client through separate VIOS partitions. It is not an MPIO solution that provides redundant paths to the same set of LUNs.

11.4 Additional considerations for IBM i as a VIOS client

This section contains additional information for IBM i.

11.4.1 Assigning XIV Storage to IBM i

In 11.2, “Power VM client connectivity to XIV” on page 284 we have explained how to configure VIOS to recognize LUNs defined in the XIV system

For VIOS to virtualize LUNs created on XIV to an IBM i client partition, both HMC and VIOS objects must be created. In the HMC, the minimum required configuration is:

- One virtual SCSI server adapter in the host partition
- One virtual SCSI client adapter in the client partition

This virtual SCSI adapter pair allows the client partition to send read and write I/O operations to the host partition. More than one virtual SCSI pair can exist for the same client partition in this environment. To minimize performance overhead in VIOS, the virtual SCSI connection is used to send I/O requests, but not for the actual transfer of data. Using the capability of the Power Hypervisor for Logical Remote Direct Memory Access (LRDMA), data are transferred directly from the Fibre Channel adapter in VIOS to a buffer in memory of the IBM i client partition.

In an IBM i client partition, a virtual SCSI client adapter is recognized as a type 290A DCxx storage controller device.

In VIOS, a virtual SCSI server adapter is recognized as a vhostX device. A new object must be created for each XIV LUN that will be virtualized to IBM i: a virtual target SCSI device, or vtscsiX. A vtscsiX device makes a storage object in VIOS available to IBM i as a standard DDxxx disk unit.

There are three types of VIOS storage objects that can be virtualized to IBM i:

- Physical disk units or volumes (hdiskX), which are XIV LUNs in this case
- Logical volumes (hdX and other)
- Files in a directory

For both simplicity and performance reasons, we recommend that you virtualize XIV LUNs to IBM i directly as physical devices (hdiskX), and not through the use of logical volumes or files.

A vtscsiX device links a LUN available in VIOS (hdiskX) to a specific virtual SCSI adapter (vhostX). In turn, the virtual SCSI adapter in VIOS is already connected to a client SCSI adapter in the IBM i client partition. Thus, the hdiskX LUN is made available to IBM i through a vtscsiX device.

What IBM i storage management recognizes as a DDxxx disk unit is not the XIV LUN itself, but the corresponding vtscsiX device. The vtscsiX device correctly reports the parameters of the LUN, such as size, to the virtual storage code in IBM i, which in turn passes them on to storage management.

Multiple vtscsiX devices, corresponding to multiple XIV LUNs, can be linked to a single vhostX virtual SCSI server adapter and made available to IBM i. Up to 16 LUNs can be virtualized to IBM i through a single virtual SCSI connection. If more than 16 LUNs are required in an IBM i client partition, an additional pair of virtual SCSI server (VIOS) and client (IBM i) adapters must be created in the HMC. Additional LUNs available in VIOS can then be linked to the new vhostX device through vtscsiX devices, making them available to IBM i.

11.4.2 Identify VIOS devices assigned to the IBM i client

Use the following method to identify the virtual devices assigned to an IBM i client.

The vhost# that was used to map devices on the VIO Server is required for identifying devices. See Example 11-5 for specific commands to identify virtual devices, making note of the Virtual Target Disk (VTD) and Backing Device information.

Example 11-5 lsmap command to identify virtual devices

```
$ lsmap -vadapter vhost5
SVSA            Physloc                                      Client Partition ID
--------------- -------------------------------------------- ------------------
vhost5          U9117.MMA.10OD394-V5-C16                     0x00000007

VTD                   vhdisk280
Status                Available
LUN                   0x8100000000000000
Backing device        hdisk280
Physloc               U5802.001.00H0104-P1-C3-T2-W5001738000230150-L1000000000000
```

Examine the details for a single virtual target disk (VTD). See Example 11-8 for specific command to list details for a single VTD.

Example 11-6 Display details for a single virtual disk

```
$ lsdev -dev vhdisk280 -vpd
  vhdisk280        U9117.MMA.10OD394-V5-C16-L1  Virtual Target Device - Disk
```

Note: The LUN number = L1.

Display the disk units details in IBM i to determine mapping between VIOS and IBM i. The ‘Ctl’ ID correlates to the ‘LUN’ ID. See Figure 11-8 for a detailed view.

```
                          Display Disk Unit Details

Type option, press Enter.
  5=Display hardware resource information details

                Serial          Sys  Sys    I/O   I/O
OPT ASP Unit    Number          Bus  Card Adapter Bus    Ctl  Dev  Compressed
 =    1    1   YAQXE2CU9P9E    255    2             0      1    0   No
 _    2    2   YYV2A8P9HF2U    255    3             0      7    0   No
 _    2    3   YLTX8ZTVW6HJ    255    3             0     11    0   No
 _    2    4   YMUFLQAVZVSX    255    3             0      3    0   No
 _    2    5   Y5RP37DE29XD    255    3             0      5    0   No
 _    2    6   YJWGG7S4DVA4    255    3             0      9    0   No
 _    2    7   Y99H5PUZYWZQ    255    3             0      1    0   No
 _    2    8   YQFCQ2BA7D9F    255    3             0      6    0   No
 _    2    9   Y7ESBQ9CGZAU    255    3             0     10    0   No
 _    2   10   YEJSNF23VJU5    255    3             0      2    0   No
 _    2   11   Y4CXMN7SQ52X    255    3             0     12    0   No
 _    2   12   Y9KFWDCGV8PT    255    3             0      4    0   No
```

Figure 11-8 Display disk units details in IBM i

Note: Unit 8 (serial number YQFCQ2BA7D0F & Ctl 1) on IBM i maps back to vhdisk280 on VIO Server.

12

SVC specific considerations

This chapter discusses specific considerations for attaching the XIV Storage System to a SAN Volume Controller (SVC).

12.1 Attaching SVC to XIV

When attaching the SAN Volume Controller (SVC) to XIV, in conjunction with connectivity guidelines already presented in Chapter 6, “Host connectivity” on page 183, the following considerations apply:

- Supported versions of SVC
- Cabling considerations
- Zoning considerations
- XIV Host creation
- XIV LUN creation
- SVC LUN allocation
- SVC LUN mapping
- SVC LUN management

12.2 Supported versions of SVC

At the time of writing, currently, SVC code v4.3.0.1 and forward are supported when connecting to the XIV Storage System. For up-to-date information, refer to:

http://www.ibm.com/support/docview.wss?rs=591&uid=ssg1S1003277#XIV

For specific information regarding SVC code, refer to the SVC support page located at:

http://www.ibm.com/systems/support/storage/software/sanvc

The SVC supported hardware list, device driver, and firmware levels for the SAN Volume Controller can be viewed at:

http://www.ibm.com/support/docview.wss?rs=591&uid=ssg1S1003277

Information about the SVC 4.3.x Recommended Software Levels can be found at:

http://www.ibm.com/support/docview.wss?rs=591&uid=ssg1S1003278

While SVC supports the IBM XIV System with a minimum SVC Software level of 4.3.0.1, we recommend that SVC software be a minimum of v4.3.1.4 or higher.

Cabling considerations

In order to take advantage of the combined capabilities of SVC and XIV, you should connect ports 1 and 3 from every interface module into the fabric for SVC use.

Figure 12-1 shows a two node cluster connected using redundant fabrics.

In this configuration:

- Each SVC node is equipped with four FC ports. Each port is connected to one of two FC switches.
- Each of the FC switches has a connection to a separate FC port of each of the six Interface Modules.

This configuration has no single point of failure:

- If a module fails, each SVC host remains connected to 5 other modules.
- If an FC switch fails, each node remains connected to all modules.
- If an SVC HBA fails, each host remains connected to all modules.
- If an SVC cable fails, each host remains connected to all modules.

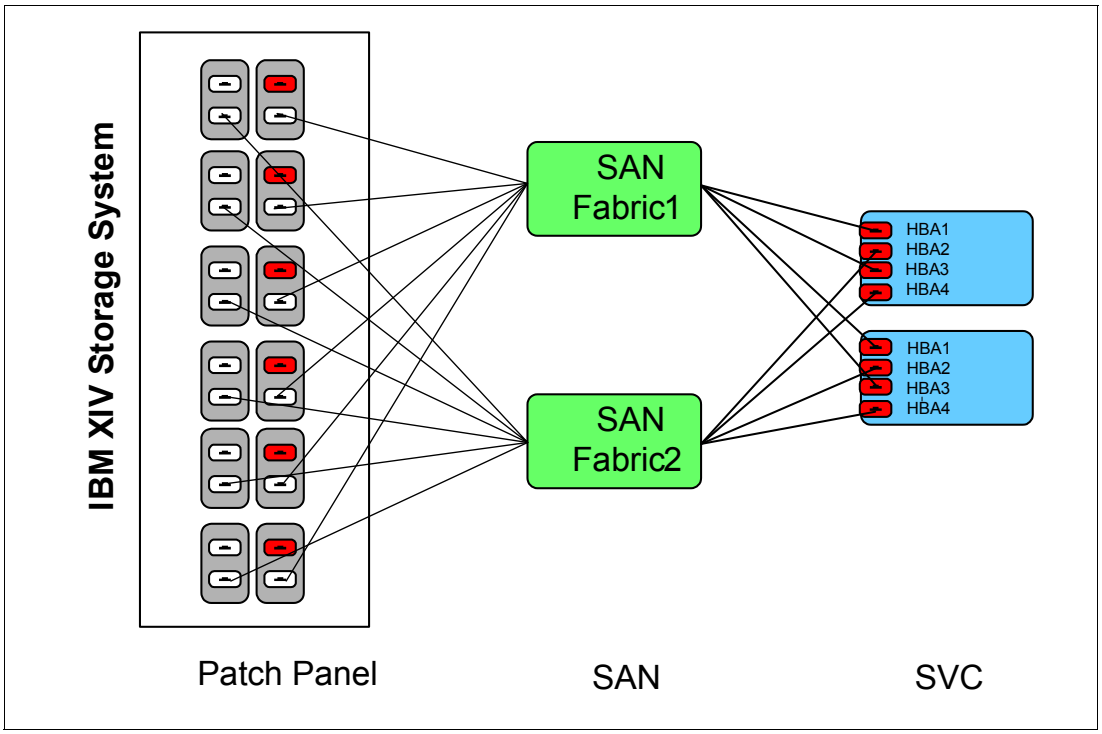


Figure 12-1 2 node SVC configuration with XIV

SVC and IBM XIV system port naming conventions

The port naming convention for the IBM XIV System ports are:

- WWPN: 5001738NNNNNRRMP
  - 001738 = Registered identifier for XIV
  - NNNNN = Serial number in hex
  - RR = Rack ID (01)
  - M = Module ID (4-9)
  - P = Port ID (0-3)

The port naming convention for the SVC ports are:

- WWPN: 5005076801X0YYZZ
  - 076801 = SVC
  - X0 = first digit is the port number on the node (1-4)
  - YY/ZZ = node number (hex value)

Zoning considerations

As a best practice, a single zone containing all 12 XIV Storage System FC ports along with all SVC node ports (a minimum of eight) should be enacted when connecting the SVC into the SAN with the XIV Storage System. This any-to-any connectivity allows the SVC to strategically multi-path its I/O operations according to the logic aboard the controller, again making the solution as a whole more effective:

- SVC nodes should connect to all Interface Modules using port 1 and port 3 on every module.
- Zones for SVC nodes should include all the SVC HBAs and all the storage HBAs (per fabric). Further details on zoning with SVC can be found in the IBM Redbooks publication, *Implementing the IBM System Storage SAN Volume Controller V4.3*, SG24-6423.

The zoning capabilities of the SAN switch are used to create distinct zones. The SVC in release 4 supports 1 Gbps, 2 Gbps, or 4 Gbps Fibre Channel fabrics. This depends on the hardware platform and on the switch where the SVC is connected.

We recommend connecting the SVC and the disk subsystem to the switch operating at the highest speed, in an environment where you have a fabric with multiple speed switches. All SVC nodes in the SVC cluster are connected to the same SAN, and present virtual disks to the hosts.

There are two distinct zones in the fabric:

- Host zones: These zones allow host ports to see and address the SVC nodes. There can be multiple host zones.
- Disk zone: There is one disk zone in which the SVC nodes can see and address the LUNs presented by XIV.

Creating a host object for SVC

Although a single host instance can be created for use in defining and then implementing the SVC, the ideal host definition for use with SVC is to consider each node of the SVC (a minimum of two) an instance of a cluster.

When creating the SVC host definition, first select **Add Cluster** and give the SVC host definition a name. Next, select **Add Host** and give the first node instance a **Name** making sure to select the **Cluster** drop-down list box and choose the SVC cluster just created.

After these have been added, repeat the steps for each instance of a node in the cluster.

From there, right-click a node instance and select **Add Port**. In Figure 12-2, note that four ports per node can be added by referencing almost identical World Wide Port Names (WWPN) to ensure the host definition is accurate.

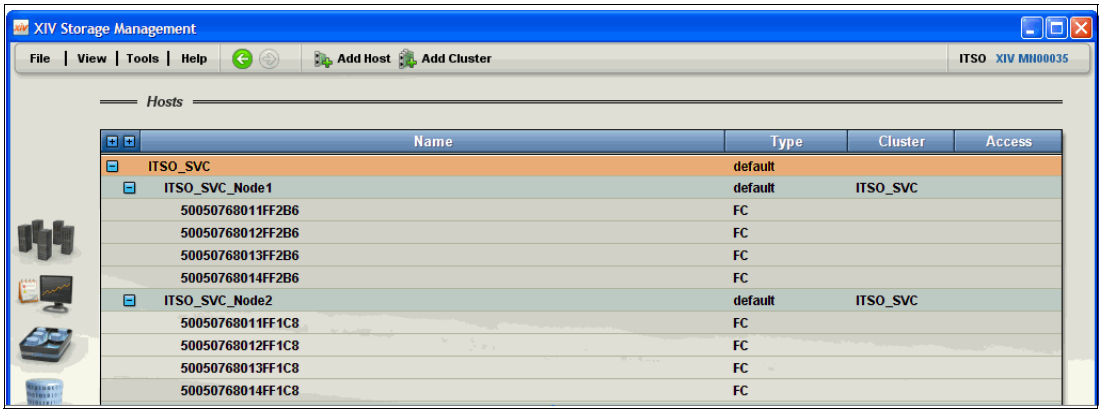


Figure 12-2 SVC host definition on XIV Storage System

By implementing the SVC as listed above, host management will ultimately be simplified and statistical metrics will be more effective because performance can be determined at the node level instead of the SVC cluster level.

For instance, after the SVC is successfully configured with the XIV Storage System, if an evaluation of the VDisk management at the I/O Group level is needed to ensure efficient utilization among the nodes, a comparison of the nodes can achieved using the XIV Storage System statistics as documented in 13.3.1, “Using the GUI” on page 305.

See Figure 12-3 for a sample display of node performance statistics.

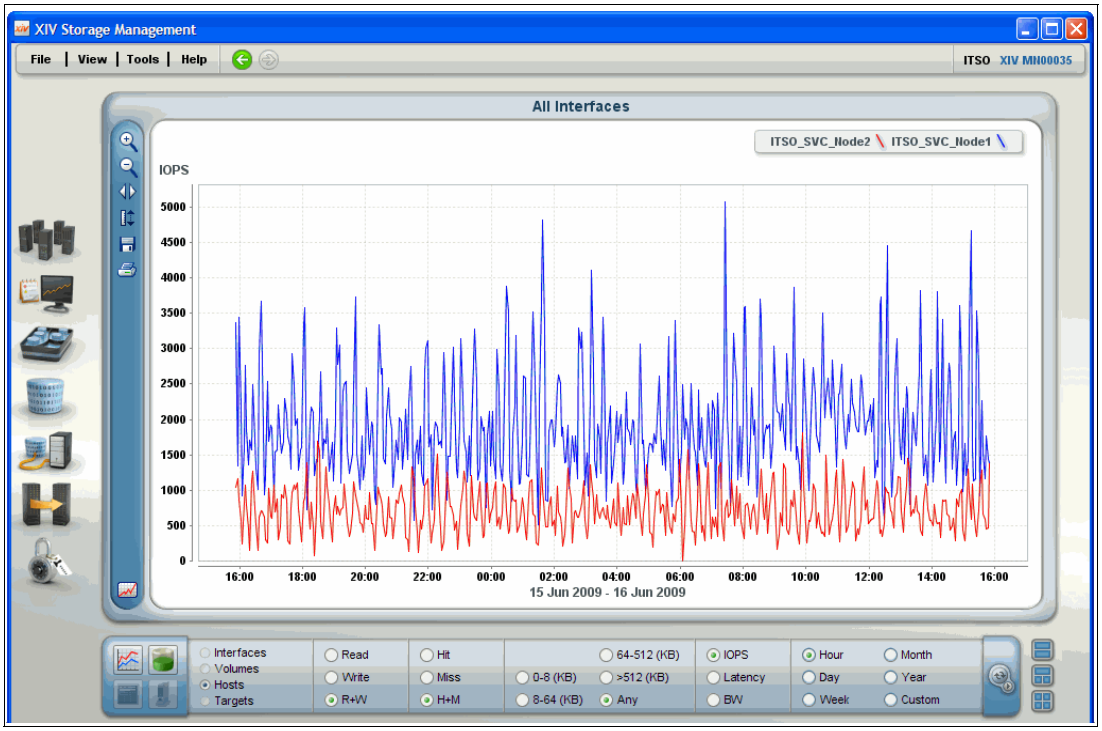


Figure 12-3 SVC node performance statistics on XIV Storage System

Volume creation for use with SVC

The IBM XIV System currently supports from 27 TB to 79 TB of usable capacity. The minimum volume size is 17 GB. While smaller LUNs can be created, we recommend that LUNs should be defined on 17 GB boundaries to maximize the physical space available.

SVC has a maximum LUN size of 2 TB that can be presented to it as a Managed Disk (MDisk). It has a maximum of 511 LUNs that can be presented from the IBM XIV System and does not currently support dynamically expanding the size of the MDisk.

Note: At the time of this writing, a maximum of 511 LUNs from the XIV Storage System can be mapped to an SVC cluster.

For a fully populated rack, with 12 ports, you should create 48 volumes of 1632 GB each. This takes into account that the largest LUN that SVC can use is 2 TB.

Because the IBM XIV System configuration grows from 6 to 15 modules, use the SVC rebalancing script to restripe VDisk extents to include new MDisks. The script is located at:

`http://www.ibm.com/alphaworks`

From there, go to the “all downloads” section and search on “svctools.”

Tip: Always use the largest volumes possible, without exceeding the 2 TB limit of SVC.

Figure 12-4 shows the number of 1632 GB LUNs created, depending on the XIV capacity:

Number of LUNs (MDisks) at 1632GB each	IBM XIV System TB used	IBM XIV System TB Capacity Available
16	26.1	27
26	42.4	43
30	48.9	50
33	53.9	54
37	60.4	61
40	65.3	66
44	71.8	73
48	78.3	79

Figure 12-4 1 Recommended values using 1632 GB LUNs

Restriction: The use of any XIV Storage System copy services functionality on LUNs presented to the SVC is not supported. Snapshots, thin provisioning, and replication is not allowed on XIV Volumes managed by SVC (MDisks).

LUN allocation using the SVC

The best use of the SVC virtualization solution with the XIV Storage System can be achieved by executing LUN allocation using some basic parameters:

- Allocate all LUNs, known to the SVC as MDisks, to one Managed Disk Group (MDG). If multiple IBM XIV Storage Systems are being managed by SVC, there should be a separate MDG for each physical IBM XIV System. We recommend that you do not include multiple disk subsystems in the same MDG, because the failure of one disk subsystem will make the MDG go offline, and thereby all VDisks belonging to the MDG will go offline.

  SVC supports up to 128 MDGs.
- In creating one MDG per XIV Storage System, use 1 GB or larger extent sizes because this large extent size ensures that data is striped across all XIV Storage System drives.

Figure 12-5 illustrates those two parameters, number of managed disks, and extent size, used in creating the MDG.

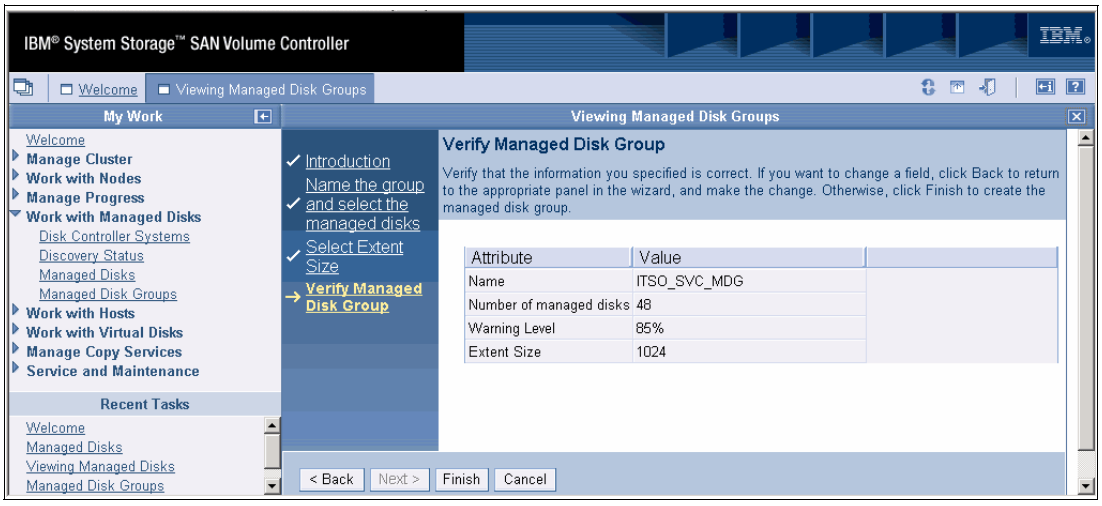


Figure 12-5 SVC Managed Disk Group creation

Doing so will drive I/O to the 4 MDisks/LUN per each of the 12 XIV Storage System Fibre Channel ports, resulting in an optimal queue depth on the SVC to adequately use the XIV Storage System.

Finalize the LUN allocation by creating striped VDisks for use by employing all 48 Mdisks in the newly created MDG.

Queue depth

SVC submits I/O to the back-end storage (MDisk) in the same fashion as any direct-attached host. For direct-attached storage, the queue depth is tunable at the host and is often optimized based on specific storage type as well as various other parameters, such as the number of initiators. For SVC, the queue depth is also tuned. The optimal value used is calculated internally. The current algorithm used with SVC4.3 to calculate queue depth follows.

There are two parts to the algorithm: a per MDisk limit and a per controller port limit.

```
Q = ((P x C) / N) / M
```

Where:

Q = The queue depth for any MDisk in a specific controller.
P = Number of WWPNs visible to SVC in a specific controller.
N = Number of nodes in the cluster.
M = Number of MDisks provided by the specific controller.
C = A constant. C varies by controller type:

- DS4100, and EMC Clarion = 200
- DS4700, DS4800, DS6K, DS8K and XIV = 1000
- Any other controller = 500

- If a 4 node SVC cluster is being used, 16 ports on the IBM XIV System and 64 MDisks, this will yield a queue depth that would be:

  ```
  Q = ((16 ports*1000)/4 nodes)/64 MDisks = 62.
  ```

  The maximum Queue depth allowed by SVC is 60 per MDisk.

- If a 4 node SVC cluster is being used, 12 ports on the IBM XIV System and 48 MDisks, this will yield a queue depth that would be:

  ```
  Q = ((12 ports*1000)/4 nodes)/48 MDisks = 62.
  ```

  The maximum Queue depth allowed by SVC is 60 per MDisk.

SVC4.3.1 has introduced dynamic sharing of queue resources based on workload. MDisks with high workload can now borrow some unused queue allocation from less busy MDisks on the same storage system. While the values are calculated internally and this enhancement provides for better sharing, it is important to consider queue depth in deciding how many MDisks to create. In these examples, when SVC is at the maximum queue depth of 60 per MDisk, dynamic sharing does not provide additional benefit.

Striped, Sequential or Image Mode VDisk guidelines

When creating a VDisk for host access, it can be created as Striped, Sequential, or Image Mode.

Striped VDisks provide for the most straightforward management. With Striped VDisks, they will be mapped to the number of MDisks in a MDG. All extents are automatically spread across all ports on the IBM XIV System. Even though the IBM XIV System already stripes the data across the entire back-end disk, we recommend that you configure striped VDisks.

We would not recommend the use of Image Mode Disks unless it is for temporary purposes. Utilizing Image Mode disks creates additional management complexity with the one-to-one VDisk to MDisk mapping.

Each node presents a VDisk to the SAN through four ports. Each VDisk is accessible from the two nodes in an I/O group. Each HBA port can recognize up to eight paths to each LUN that is presented by the cluster. The hosts must run a multipathing device driver before the multiple paths can resolve to a single device. You can use fabric zoning to reduce the number of paths to a VDisk that are visible by the host. The number of paths through the network from an I/O group to a host must not exceed eight; configurations that exceed eight paths are not supported. Each node has four ports and each I/O group has two nodes. We recommend that a VDisk be seen in the SAN by four paths.

Guidelines for SVC extent size

SVC divides the managed disks (MDisks) that are presented by the IBM XIV System into smaller chunks that are known as extents. These extents are then concatenated to make virtual disks (VDisks). All extents that are used in the creation of a particular VDisk must all come from the same Managed Disk Group (MDG).

SVC supports extent sizes of 16, 32, 64, 128, 256, 512, 1024, and 2048 MB. The extent size is a property of the Managed Disk Group (MDG) that is set when the MDG is created. All managed disks, which are contained in the MDG, have the same extent size, so all virtual disks associated with the MDG must also have the same extent size.

Figure 12-6 depicts the relationship of an MDisk to MDG to a VDisk.

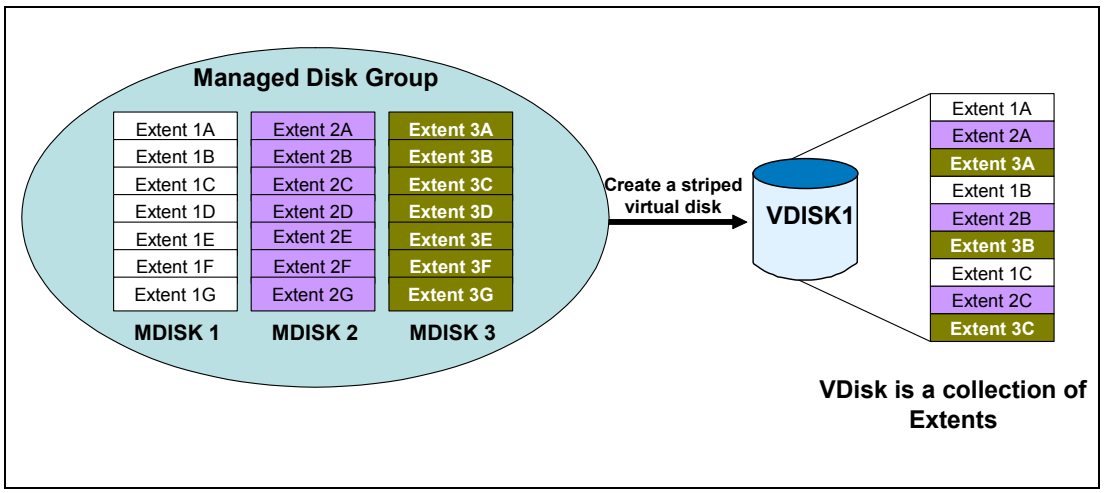


Figure 12-6 MDisk to VDisk mapping

The recommended extent size is 1 GB. While smaller extent sizes can be used, this will limit the amount of capacity that can be managed by the SVC Cluster.

13

Performance characteristics

In Chapter 2, “XIV logical architecture and concepts” on page 9, we have described the XIV Storage System’s parallel architecture, disk utilization, and unique caching algorithms. These characteristics, inherent to the system design, deliver optimized and consistent performance regardless of the workload the XIV Storage System endures.

The current chapter further explores the concepts behind this high performance, provides the best practice recommendations when connecting to an XIV Storage System, and explains how to extract statistics provided by the XIV Storage System Graphical User Interface (GUI) and Extended Command Line Interface (XCLI).

13.1 Performance concepts

The XIV Storage System maintains a high level of performance by leveraging all the disk, memory, and I/O resources in the system at all times. The system offers advanced caching features, employing effective data mirroring, and integrating power copy services functionality such as snapshots and remote mirroring. These characteristics are the basis for many unique features that distinguish XIV from its competition.

In this chapter we cover these functions as they pertain to the XIV Storage System:

- Full disk resource utilization
- Caching mechanisms
- Data mirroring
- Snapshots

13.1.1 Full disk resource utilization

Utilization of all disk resources improves the performance by minimizing the bottlenecks within the system. The XIV Storage System stripes and mirror data into 1 MB partitions across all the disks in the system; it then disperses the 1 MB partitions in a pseudo-random distribution. This pseudo-random distribution results in a lower access density, which is measured by throughput divided by the total disk capacity. Refer to Chapter 2, “XIV logical architecture and concepts” on page 9 for further details about the architecture of the system.

Several benefits result from fully utilizing all of the disk resources. Each disk drive performs an equal workload as the data is balanced across the entire system. The pseudo-random distribution ensures load balancing at all times and eliminates *hot-spots* in the system.

13.1.2 Caching mechanisms

The XIV Storage System caching management is unique, by dispersing the cache into each module as opposed to a central memory cache. The distributed cache enables each module to concurrently service host I/Os and cache to disk access, as opposed to the central memory caching algorithm, which implements memory locking algorithms that generate access contention.

To improve memory management, each Data Module uses a PCI Express (PCI-e) bus between the cache and the disk modules, which provides a sizable interconnect between the disk and the cache. This design aspect allows large amounts of data to be quickly transferred between the disks and the cache via the bus.

Having a large bus “pipe” permits the XIV Storage System to have small cache pages. More so, a large bus “pipe” between the disk and the cache allows the system to perform many small requests in parallel, again improving the performance.

A Least Recently Used (LRU) algorithm is the basis for the cache management algorithm. This feature allows the system to generate a high hit ratio for frequently utilized data. In other words, the efficiency of the cache usage for small transfers is very high, when the host is accessing the same data set.

The cache algorithm starts with a single 4 KB page and gradually increase the number of pages prefetched until an entire partition, 1 MB, is read into cache. If the access results in a cache hit, the algorithm doubles the amount of data prefetched into the system.

The prefetching algorithm continues to double the prefetch size until a cache miss occurs, or the prefetch size maximum of 1 MB is obtained. Because the modules are managed independently if a prefetch crosses a module boundary, then the logically adjacent module (for that volume) is notified in order to begin pre-staging the data into its local cache.

13.1.3 Data mirroring

The XIV Storage System maintains two copies of each 1 MB data partition, referred to as the *primary partition* and *secondary* partition. The primary partition and secondary partition for the same data are also kept on separate disks in separate modules. We call this *data mirroring*.

By implementing data mirroring, the XIV Storage System performs a single disk access on reads and two disk accesses on writes; one access for the primary copy and one access for the mirrored secondary copy. Other storage systems that use RAID architecture might translate the I/O into two disk writes and two disk reads for RAID 5 and three disk writes and three disk reads for RAID 6. This allows the XIV Storage System data mirroring algorithm reduce the disk access times and provide quicker responses to requests.

A 1 MB partition is the amount of data stored on a disk with a contiguous address range. Because the cache operates on 4 KB pages, the smallest chunk of data that can be staged into cache is a single cache page, or 4 KB. The data mirroring only mirrors the data that has been modified. By only storing modified data, the system performs at maximum efficiency.

It is also important to note that the data mirroring scheme is not the same as a data stripe in a RAID architecture. Specifically, a *partition* refers to a contiguous region on a disk. The partition is how the XIV Storage System tracks and manages data. This large 1 MB partition is not the smallest workable unit. Because the cache uses 4 KB pages, the system can stage and modify smaller portions of data within the partition, which means that the large partition assists the sequential workloads and the small cache page improves performance for the random workloads.

Disk rebuilds in which data is populated to a new disk after replacement of a phased out or failed disk gains an advantage due to the mirroring implementation. When a disk fails in the RAID 5 or RAID 6 system, a spare disk is added to the array and the data is reconstructed on that disk using the parity data. The process can take several hours based on the size of the disk drive. With the XIV Storage System, the rebuild is not focused on one disk. After the disk is replaced, the system enters a redistribution phase. In the background, the work is spread across all the disks in the system as it slowly moves data back onto to the new disk. By spreading the work across all the disks, each disk is performing a small percentage of work, therefore the impact to the host is minimal.

13.1.4 Snapshots

Snapshots complete nearly instantly within the XIV Storage System. When a snapshot is issued, no data is copied but rather the snapshot creates system pointers to the original data. As the host writes modified data in the master volume, the XIV Storage System redirects the write data to a new partition. Only the data that was modified by the host is copied into the new partition, which prevents moving the data multiple times and simplifies the internal management of the data. Refer to the *Theory of Operations*, GA32-0639-03 for more details about how the snapshot function is implemented.

13.2 Best practices

Tuning of the XIV Storage System is not required by design. Because the data is balanced across all the disks, the performance is at maximum efficiency. This section is dedicated to external considerations that enable maximum performance. The recommendations in this section are host-agnostic and are general rules when operating the XIV Storage System.

13.2.1 Distribution of connectivity

The main goal for the host connectivity is to create a balance of the resources in the XIV Storage System. Balance is achieved by distributing the physical connections across the Interface Modules. A host usually manages multiple physical connections to the storage device for redundancy purposes via a SAN connected switch. It is ideal to distribute these connections across each of the Interface Modules. This way the host utilizes the full resources of each module that is connected to and can obtain maximum performance. It is important to note that it is not necessary for each host instance to connect to each Interface Module. However, when the host has more than one physical connection, it is beneficial to have the cables divided across the modules.

Similarly, if multiple hosts and have multiple connections, make sure to spread the connections evenly across the Interface Modules. Refer to 3.2.4, “Patch panel” on page 58.

13.2.2 Host configuration considerations

There are several key points when configuring the host for optimal performance. Because the XIV Storage System is distributing the data across all the disks an additional layer of volume management at the host, such as Logical Volume Manager (LVM), might hinder performance for workloads. Multiple levels of striping can create an imbalance across a specific resource. Therefore, the recommendation is to disable host striping of data for XIV Storage System volumes and allow the XIV Storage System to manage the data.

Based on your host workload, you might need to modify the maximum transfer size that the host generates to the disk to obtain the peak performance. For applications with large transfer sizes, if a smaller maximum host transfer size is selected, the transfers are broken up, causing multiple round-trips between the host and the XIV Storage System. By making the host transfer size as large or larger than the application transfer size, fewer round-trips occur, and the system experiences improved performance. If the transfer is smaller than the maximum host transfer size, the host only transfers the amount of data that it has to send.

Refer to the vendor hardware manuals for queue depth recommendations.

Due to the distributed data features of the XIV Storage System, high performance is achieved by parallelism. Specifically, the system maintains a high level of performance as the number of parallel transactions occur to the volumes. Ideally, the host workload can be tailored to use multiple threads or spread the work across multiple volumes.

13.2.3 XIV sizing validation

Currently, your IBM Representative provides sizing recommendations based on the workload.

13.3 Performance statistics gathering

During normal operation, the XIV Storage System constantly gathers statistical information. The data can then be processed using the GUI or Extended Command Line Interface (XCLI). This section introduces the techniques for processing the statistics data.

13.3.1 Using the GUI

The GUI provides a mechanism to gather statistics data. For a description of setting up and using the GUI, refer to Chapter 4, “Configuration” on page 79. When working with the statistical information, the XIV Storage System collects and maintains the information internally. As the data ages, it is consolidated to save space. By selecting specific filters, the requested data is mined and displayed. This section discusses the functionality of the GUI and how to retrieve the required data.

The first item to note is the current IOPS for the system is always displayed in the bottom center of the window. This feature provides simple access to the current stress of the system. Figure 13-1 illustrates the GUI and the IOPS display; it also shows how to start the statistics monitor.

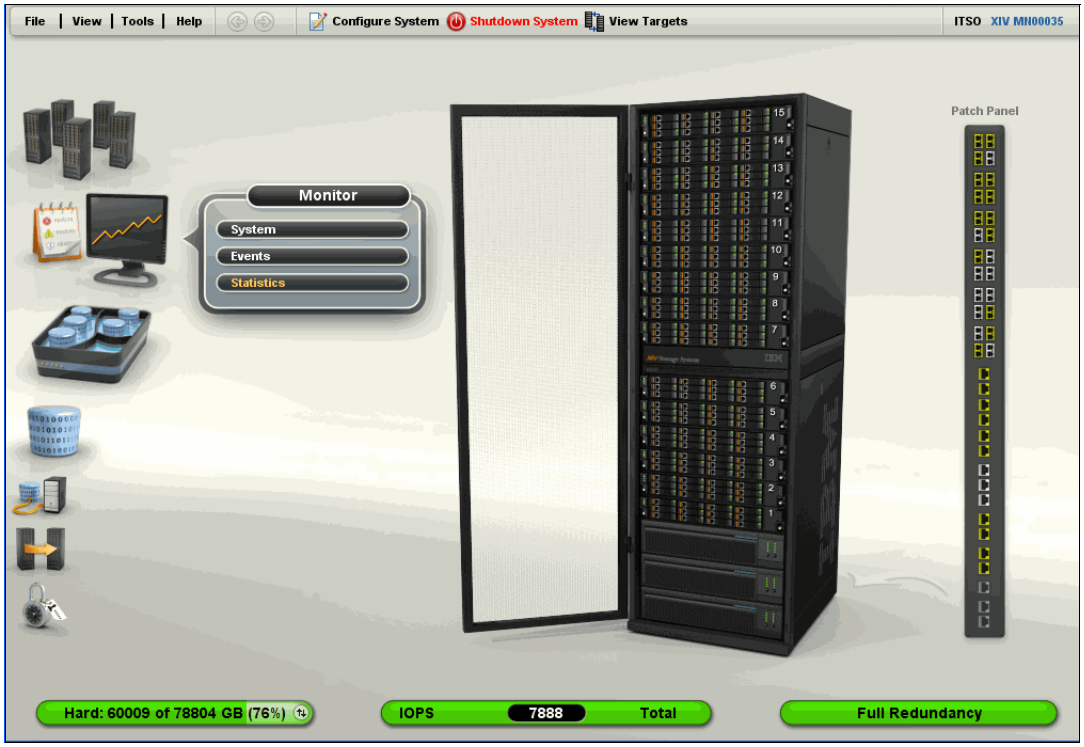


Figure 13-1 Starting the statistics monitor on the GUI

Select **Statistics** from the Monitor menu as shown in Figure 13-1 to display the Monitor default view that is shown in Figure 13-2.

Figure 13-2 shows the system IOPS for the past 24 hours:

- The X-axis of the graph represents the time and can vary from minutes to months.
- The Y-axis of the graph is the measurement selected. The default measurement is IOPS.

The statistics monitor also illustrates latency and bandwidth.

Figure 13-2 Default statistics monitor view

The other options in the statistics monitor act as filters for separating data. These filters are separated by the type of transaction (reads or writes), cache properties (hits compared to misses), or the transfer size of I/O as seen by the XIV Storage System. Refer to Figure 13-3 for a better view of the filter pane.

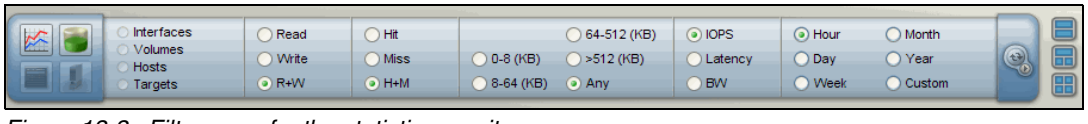


Figure 13-3 Filter pane for the statistics monitor

The filter pane allows you to select multiple items within a specific filter, for example, if you want to see reads and writes separated on the graph. By holding down Ctrl on the keyboard and selecting the read option and then the write option, you can witness both items displayed on the graph.

As shown in Figure 13-4, one of the lines represents the reads and the other line represents the writes. On the GUI, these lines are drawn in separate colors to differentiate the metrics.

This selection process can be performed on the other filter items as well.

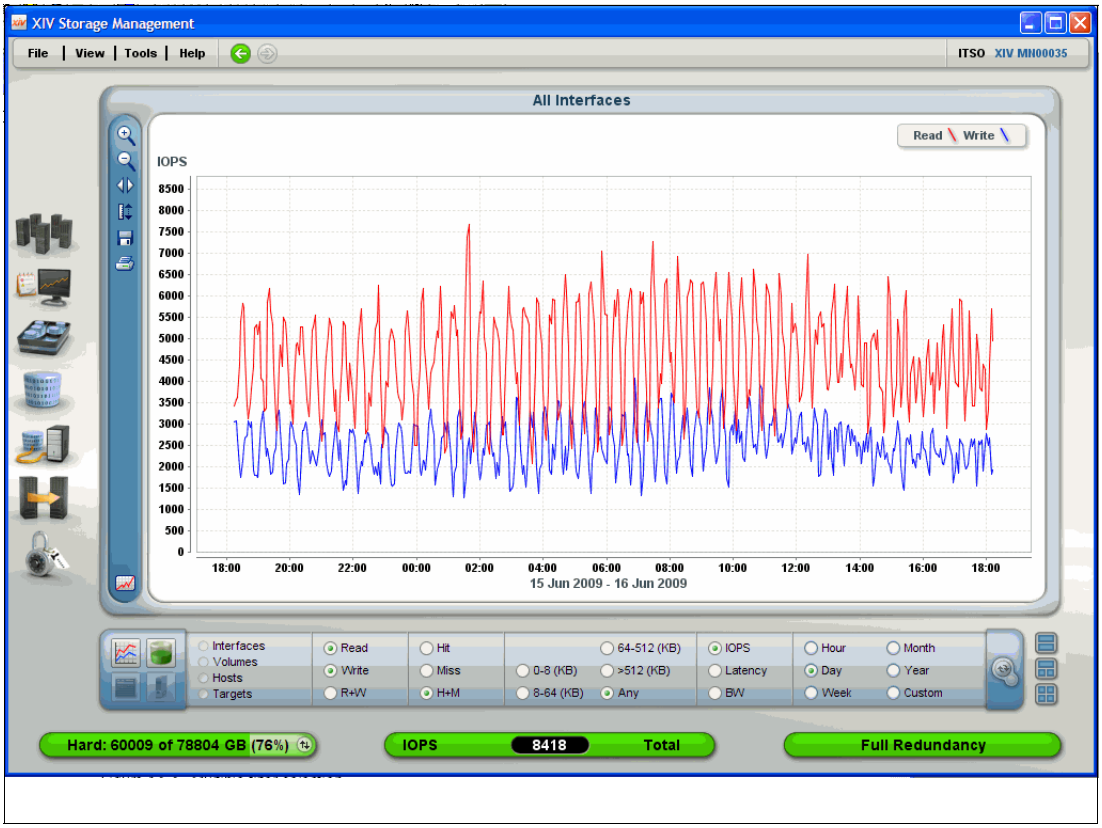


Figure 13-4 Multiple filter selection

In certain cases, the user needs to see multiple graphs at one time. On the right side of the filter pane, there is a selection to add graphs (refer to Figure 13-3 on page 306). Up to four graphs are managed by the GUI. Each graph is independent and can have separate filters.

Next, Figure 13-5 illustrates this concept. The top graph is the IOPS for the day with the reads and writes separated. The second graph displays the bandwidth for several minutes with reads and writes separated, which provides quick and easy access to multiple views of the performance metrics.

Figure 13-5 Multiple graphs using the GUI

There are several additional filters available, such as filtering by host, volumes, interfaces, or targets. These items are defined on the left side of the filter pane. When clicking one of these filters, a dialog window appears. Highlight the item, or select a maximum of four using the Ctrl key, to be filtered and then click **Click to select.** It moves the highlighted item to the lower half of the dialog box. In order to generate the graph, you must click the green check mark located on the lower right side of the dialog box. Your new graph is generated with the name of the filter at the top of the graph. Refer to Figure 13-6 for an example of this filter.

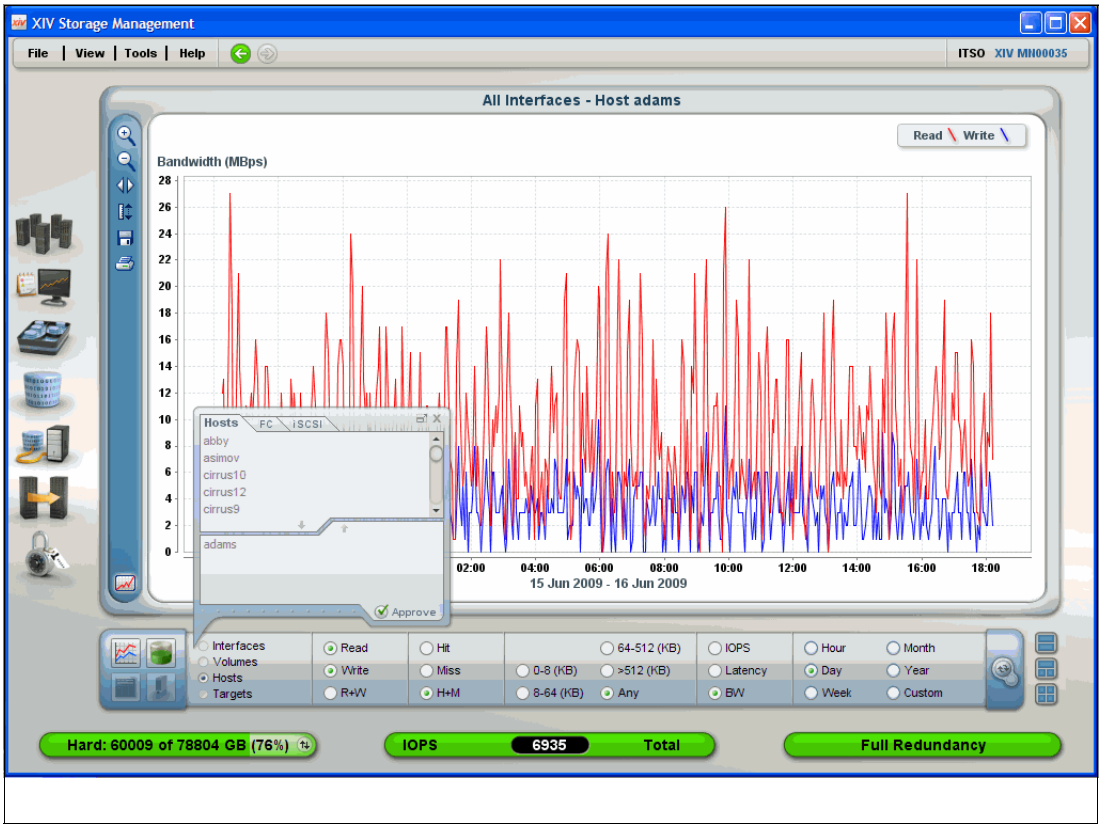


Figure 13-6 Example of a host filter

On the left side of the chart in the blue bar, there are several tools to assist you in managing the data.

Figure 13-7 shows the chart toolbar in more detail.

Figure 13-7 Chart toolbar

The top two tools (magnifying glasses) zoom in and out for the chart, and the second set of two tools adjusts the X-axis and the Y-axis for the chart. Finally, the bottom two tools allow you to export the data to a comma-separated file or print the chart to a printer.

13.3.2 Using the XCLI

The second method to collect statistics is through the XCLI operation. In order to access the e XCLI session, refer to Chapter 4, “Configuration” on page 79.

To retrieve the system time, issue the **time_list** command, and the system retrieves the current time. Refer to Example 13-1 for an example of retrieving the XIV Storage System time.

Example 13-1 Retrieving the XIV Storage System time

```
>> time_list
Time       Date         Time Zone   Daylight Saving Time
11:45:42   2009-06-16   GMT         no
```

After the system time is obtained, the **statistics_get** command can be formatted and issued.

The **statistics_get** command requires several parameters to operate. The command requires that you enter a starting or ending time point, a count for the number of intervals to collect, the size of the interval, and the units related to that size. The TimeStamp is modified from the previous **time_list** command. Example 13-2 provides a description of the command.

Example 13-2 The statistics_get command format

```
statistics_get [ host=H | host_iscsi_name=initiatorName | host_fc_port=WWPN |
target=RemoteTarget | remote_fc_port=WWPN | remote_ipaddress=IPAdress |
vol=VolName | ipinterface=IPInterfaceName | local_fc_port=ComponentId ]
< start=TimeStamp | end=TimeStamp > [ module=ComponentId ] count=N
interval=IntervalSize resolution_unit=<minute|hour|day|week|month>
```

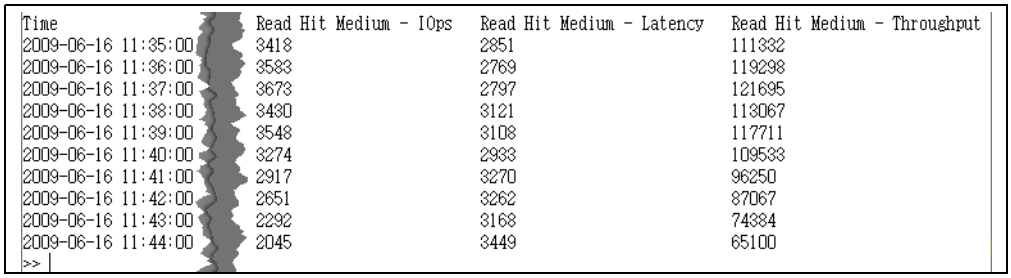
To further explain this command, assume that you want to collect 10 intervals, and each interval is for one minute. The point of interest occurred June 16 2008 roughly 15 minutes 11:45:00. It is important to note the **statistics_get command** allows you to gather the performance data from any time period.

The time stamp is formatted as *YYYY-MM-DD:hh:mm:ss*, where the *YYYY* represents a four digit year, *MM* is the two digit month, and *DD* is the two digit day. After the date portion of the time stamp is specified, you specify the time, where *hh* is the hour, *mm* is the minute, and *ss* represents the seconds.

Example 13-3 shows a typical use of this command, and Figure 13-8 shows some sample output of the statistics. The output displayed is a small portion of the data provided.

Example 13-3 The statistics_get command example

```
>> statistics_get end=2009-06-16.11:45:00 count=10 interval=1
resolution_unit=minute
```



Time	Read Hit Medium - IOps	Read Hit Medium - Latency	Read Hit Medium - Throughput
2009-06-16 11:35:00	3418	2851	111332
2009-06-16 11:36:00	3583	2769	119298
2009-06-16 11:37:00	3673	2797	121695
2009-06-16 11:38:00	3430	3121	113067
2009-06-16 11:39:00	3548	3108	117711
2009-06-16 11:40:00	3274	2933	109533
2009-06-16 11:41:00	2917	3270	96250
2009-06-16 11:42:00	2651	3262	87067
2009-06-16 11:43:00	2292	3168	74384
2009-06-16 11:44:00	2045	3449	65100

>>

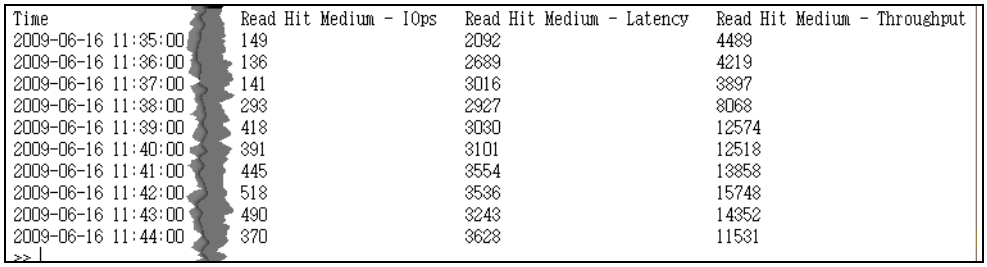
Figure 13-8 Output from statistics_get command

Extending this example, assume that you want to filter out a specific host defined in the XIV Storage System. By using the host filter in the command, you can specify for which host you want to see performance metrics, which allows you to refine the data that you are analyzing.

Refer to Example 13-4 for an example of how to perform this operation, and see Figure 13-9 for a sample of the output for the command.

Example 13-4 The statistics_get command using the host filter

```
>> statistics_get host=adams end=2009-06-16.11:45:00 count=10 interval=1
resolution_unit=minute
```



Time	Read Hit Medium - IOps	Read Hit Medium - Latency	Read Hit Medium - Throughput
2009-06-16 11:35:00	149	2092	4489
2009-06-16 11:36:00	136	2689	4219
2009-06-16 11:37:00	141	3016	3897
2009-06-16 11:38:00	293	2927	8068
2009-06-16 11:39:00	418	3030	12574
2009-06-16 11:40:00	391	3101	12518
2009-06-16 11:41:00	445	3554	13858
2009-06-16 11:42:00	518	3536	15748
2009-06-16 11:43:00	490	3243	14352
2009-06-16 11:44:00	370	3628	11531

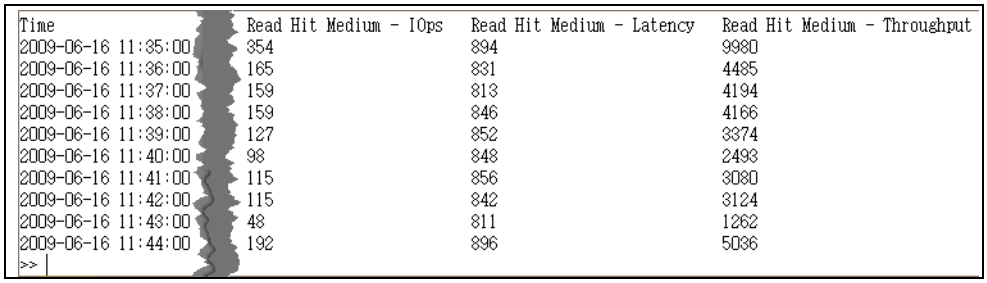
>>

Figure 13-9 Output from the statistics_get command using the host filter

In addition to the filter just shown, the **statistics_get command** is capable of filtering iSCSI names, host worldwide port names (WWPNs), volume names, modules, and many more fields. As an additional example, assume you want to see the workload on the system for a specific module. The module filter breaks out the performance on the specified module. Example 13-5 pulls the performance statistics for module 5 during the same time period of the previous examples. Figure 13-10 shows the output.

Example 13-5 The statistics_get command using the module filter

```
>> statistics_get end=2009-06-16.11:45:00 module=5 count=10 interval=1
resolution_unit=minute
```



Time	Read Hit Medium - IOps	Read Hit Medium - Latency	Read Hit Medium - Throughput
2009-06-16 11:35:00	354	894	9980
2009-06-16 11:36:00	165	831	4485
2009-06-16 11:37:00	159	813	4194
2009-06-16 11:38:00	159	846	4166
2009-06-16 11:39:00	127	852	3374
2009-06-16 11:40:00	98	848	2493
2009-06-16 11:41:00	115	856	3080
2009-06-16 11:42:00	115	842	3124
2009-06-16 11:43:00	48	811	1262
2009-06-16 11:44:00	192	896	5036

>>

Figure 13-10 Output from statistics_get command using the module filter

14

Monitoring

This chapter describes the various methods and functions that are available to monitor the XIV Storage System. It also shows how you can gather information from the system in real time, in addition to the self-monitoring, self-healing, and automatic alerting function implemented within the XIV software.

Furthermore, this chapter also discusses the Call Home function and secure remote support and repair.

14.1 System monitoring

The XIV Storage System software includes features that allow you to monitor the system:

- You can review or request at any time the current system status and performance statistics.
- You can set up alerts to be triggered when specific error conditions or problems arise in the system. Alerts can be conveyed as messages to the operator, an e-mail, or a Short Message Service (SMS) text to a mobile phone. Depending on the nature or severity of the problem, the system will automatically alert the IBM support center, which immediately initiates the necessary actions to promptly repair the system.
- You can configure IBM Tivoli Storage Productivity Center to communicate with the built in SMI-S agent the XIV Storage System.
- In addition, the optional Secure Remote Support feature allows remote monitoring and repair by IBM support.

14.1.1 Monitoring with the GUI

The monitoring functions available from the XIV Storage System GUI allow the user to easily display and review the overall system status, as well as events and several statistics.

These functions are accessible from the Monitor menu as shown in Figure 14-1.

Figure 14-1 GUI monitor functions

Monitoring the system

Selecting System from the Monitor menu shown in Figure 14-1 takes you to the system view, shown in Figure 14-2 (note that this view is also the default or main GUI window for the selected system).

The System view shows a graphical representation of the XIV Storage System rack with its components. You can click the curved arrow located at the bottom right of the picture of the rack to display a view of the patch panel.

You get a quick overview in real time of the system's overall condition and the status of its individual components. The display changes dynamically to provide details about a specific component when you position the mouse cursor over that component.

Figure 14-2 Monitoring the IBM XIV

Status bar indicators located at the bottom of the window indicate the overall operational levels of the XIV Storage System:

- The first indicator on the left shows the amount of soft or hard storage capacity currently allocated to Storage Pools and provides alerts when certain capacity thresholds are reached. As the physical, or hard, capacity consumed by volumes within a Storage Pool passes certain thresholds, the color of this meter indicates that additional hard capacity might need to be added to one or more Storage Pools.

  Clicking the icon on the right side of the indicator bar that represents up and down arrows will toggle the view between hard and soft capacity.

  Our example indicates that the system has a usable hard capacity of 79113 GB, of which 84% or 66748 GB are actually used.

  You can also get more detailed information and perform more accurate capacity monitoring by looking at Storage Pools (refer to 4.3.1, “Managing Storage Pools with the XIV GUI” on page 98).

- The second indicator in the middle, displays the number of I/O operations per second (IOPS).

- The third indicator on the far right shows the general system status and, for example, indicates when a redistribution is underway.

  In our example, the general system status indicator shows that the system is undergoing a Rebuilding phase, which was triggered because of a failing disk (Disk 7 in Module 7) as shown in Figure 14-2.

Monitoring events

To get to the Events window, select Events from the Monitor menu as shown in Figure 14-3.

Extensive information and many events are logged by the XIV Storage System. The system captures entries for problems with various levels of severity, including warnings and other informational messages. These informational messages include detailed information about logins, configuration changes, and the status of attached hosts and paths. All of the collected data can be reviewed in the Events window that is shown in Figure 14-3.

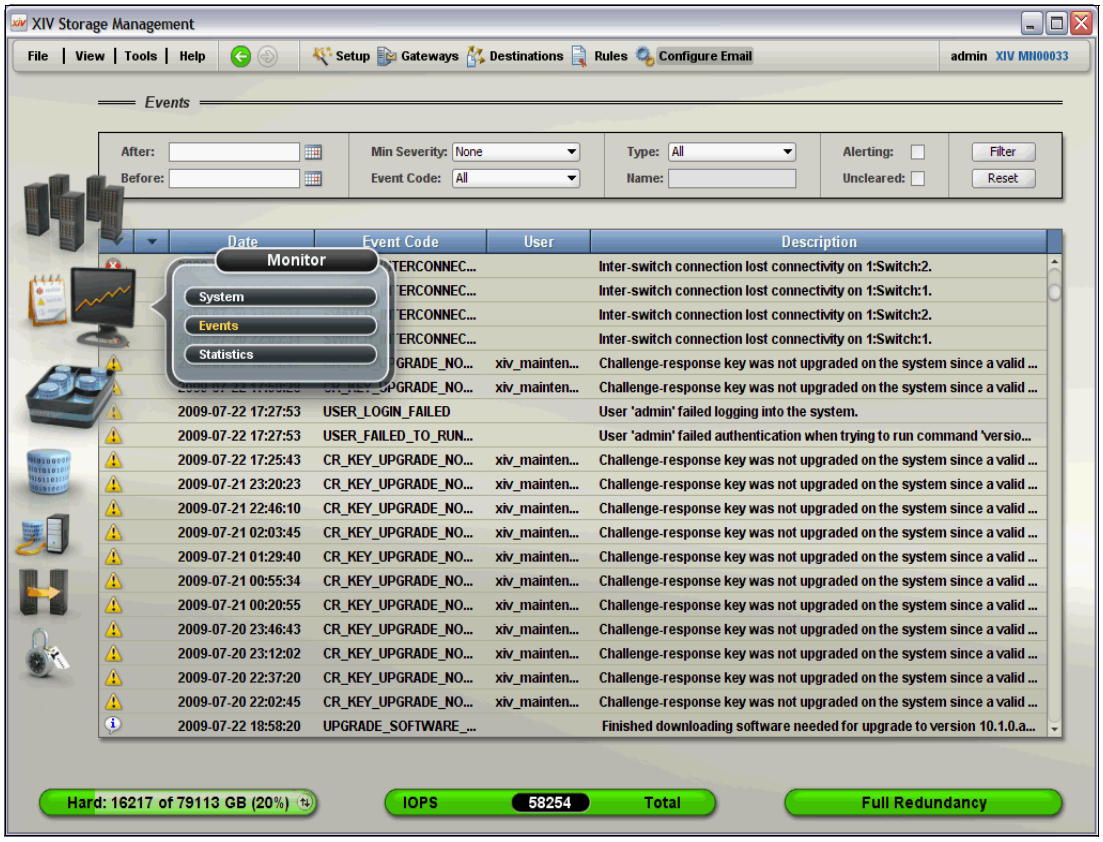


Figure 14-3 Events

Because many events are logged, the number of entries is typically huge.

To get a more useful and workable view, there is an option to filter the events logged. Without filtering the events, it might be extremely difficult to find the entries for a specific incident or information. Figure 14-4 shows the possible filter options for the events.

Figure 14-4 Event filter

If you double-click a specific event in the list, you can get more detailed information about that particular event, along with a recommendation about what eventual action to take.

Figure 14-5 show details for a critical event where a module failed. For this type of event, you must immediately contact IBM XIV support.

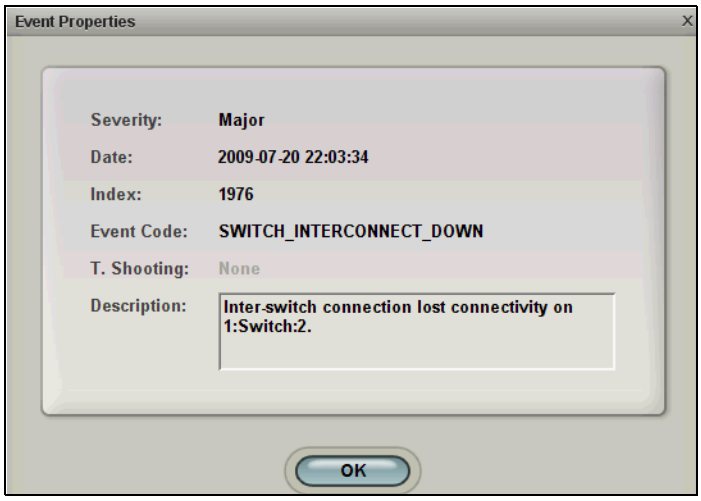


Figure 14-5 Event properties

Event severity

The events are classified into a level of severity depending on their impact on the system. Figure 14-6 gives an overview of the criteria and meaning of the various severity levels.

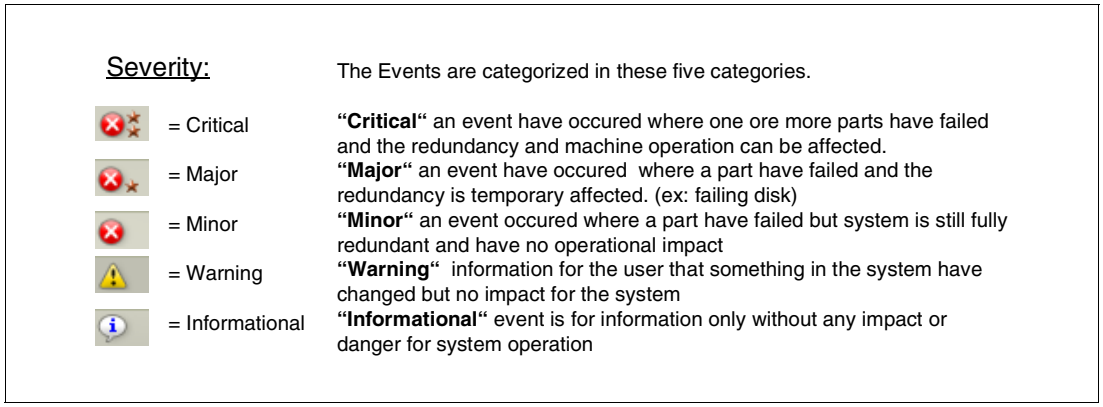


Figure 14-6 Event severity

Event configuration

The events window offers a Toolbar (refer to Figure 14-7) which contains a Setup wizard, the ability to view and modify gateways, destinations, and rules, as well as modify the email addresses for the XIV Storage system.

Clicking the Setup icon starts the Events Configuration Wizard, which guides you through the process to create gateways, add destinations, and define rules for event notification.

Figure 14-7 Event rules configuration

For further information about event notification rules, refer to 14.3, "Call Home and Remote support" on page 351.

Monitoring statistics

The Statistics monitor, which is shown in Figure 14-8, provides information about the performance and workload of the IBM XIV.

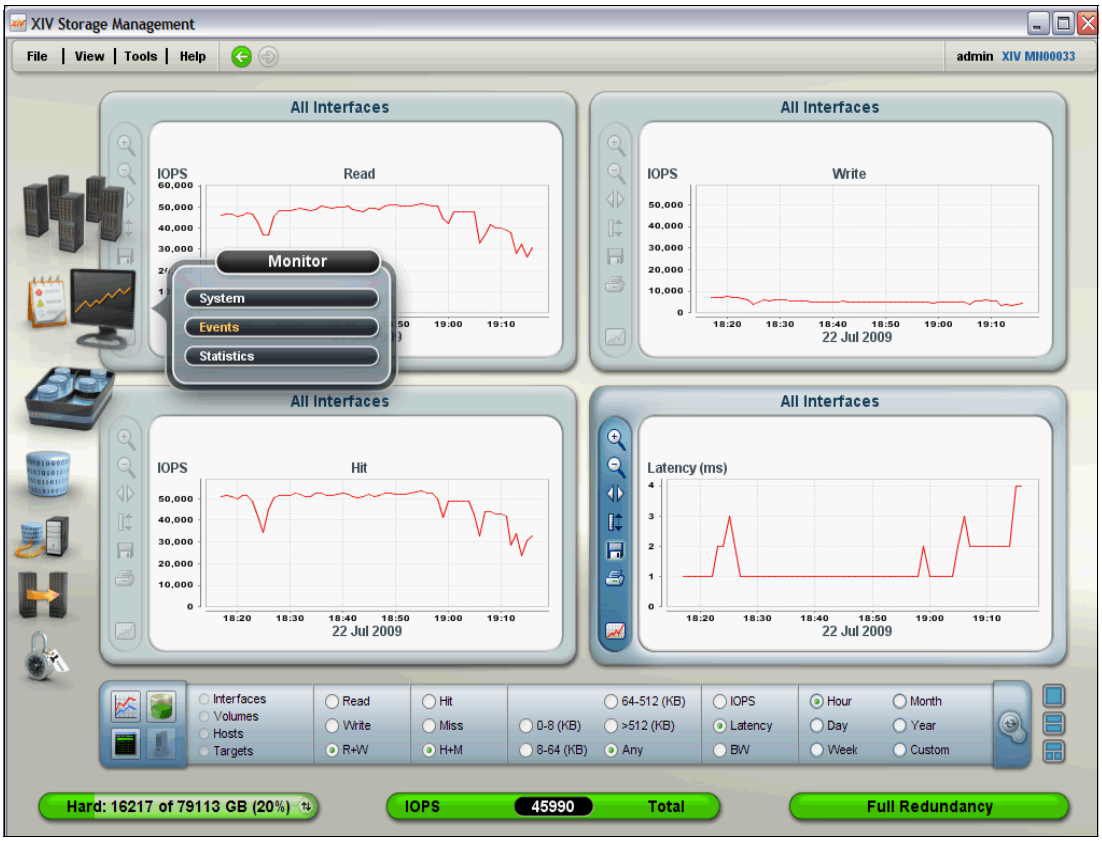


Figure 14-8 Monitor statistics

There is flexibility in how you can visualize the statistics. Options are selectable from a control pane located at the bottom of the window, which is shown in Figure 14-9.

Figure 14-9 Statistics filter

For detailed information about performance monitoring, refer to 13.3, “Performance statistics gathering” on page 305.

14.1.2 Monitoring with XCLI

The Extended Command Line Interface (XCLI) provides several commands to monitor the XIV Storage System and gather real-time system status, monitor events, and retrieve statistics. Refer also to 4.1, “IBM XIV Storage Management software” on page 80 for more information about how to set up and use the XCLI.

System monitoring

Several XCLI commands are available for system monitoring. We illustrate several commands next. For complete information about these commands, refer to the *XCLI Users Guide*, which is available at:

http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp

The **state_list** command, shown Example 14-1, gives an overview of the general status of the system. In the example, the system is operational, data is fully redundant and no shutdown is pending.

Example 14-1 The state_list command

```
>> state_list
Category          Value
shutdown_reason   No Shutdown
target_state      on
off_type          off
redundancy_status Full Redundancy
system_state      on
safe_mode         no
```

In Example 14-2, the **system_capacity_list** command shows an overview of used and free capacity system-wide. In the example, both the hard and soft usable capacity is 79113 GB, with 54735 GB of free hard capacity, 54302 GB of free soft capacity. It also shows that the all spare capacity is still available.

Example 14-2 The system_capacity_list command

```
>> system_capacity_list
Soft   Hard   Free Hard   Free Soft   Spare Modules   Spare Disks   Target Spare Modules   Target Spare Dis
79113  79113  54735       54202       1               3             1                      3
```

In Example 14-3, the **version_get** command displays the current version of the XIV code installed on the system. Knowing the current version of your system assists you in determining when upgrades are required.

Example 14-3 The version_get command

```
>> version_get
Version
10.1
```

In Example 14-4, the **time_list** command is used to retrieve the current time from the XIV Storage System. This time is normally set at the time of installation. Knowing the current system time is required when reading statistics or events. In certain cases, the system time might differ from the current time (at the user’s location), and therefore, knowing when something occurred according to the system time assists with debugging issues.

Example 14-4 The time_list command

```
>> time_list
Time      Date        Time Zone  Daylight Saving Time
08:57:15  2008-08-19  GMT        no
```

Use appropriate operating system commands to obtain local computer time. Be aware that checking time on all elements of the infrastructure (switches, storage, hosts, and so on) might be required when debugging issues.

System components status

In this section, we present several XCLI commands that are available to get the status of specific system components, such as disks, modules, or adapters.

The **component_list** command that is shown In Example 14-5 gives the status of all hardware components in the system. The filter options **filter=<FAILED | NOTOK>** is used to only return failing components. The example shows a failing disk in module 4 on position 9.

Example 14-5 The component_list command

```
>> component_list filter=NOTOK
Component ID  Status  Currently Functioning
1:Disk:4:9    Failed  no
```

As shown in Example 14-6, the **disk_list** command provides more in-depth information for any individual disk in the XIV Storage System, which might be helpful in determining the root cause of a disk failure. If the command is issued without the disk parameter, all the disks in the system are displayed.

Example 14-6 The disk_list command

```
>> disk_list disk=1:Disk:13:10
Component ID   Status  Currently Functioning   Capacity (GB)  Target Status   Model        Size    Serial
1:Disk:13:11   Failed  yes                     1 TB                           Hitachi...   942772  PAJ1W2XF

>> disk_list disk=1:Disk:13:11
Component ID   Status  Currently Functioning   Capacity (GB)  Target Status   Model        Size    Serial
1:Disk:13:11   Failed  yes                     1 TB                           Hitachi...   942772  PAJUO2YF
```

In Example 14-7, the module_list command displays details about the modules themselves. If the module parameter is not provided, all the modules are displayed. In addition to the status of the module, the output describes the number of disks, number of FC ports, and number of iSCSI ports.

Example 14-7 The module_list command

```
>> module_list module=1:Module:4
Component ID   Status Currently Functioning Target Status   Type             Data Disks FC Ports   iSCSI P
1:Module:4     OK     yes                                   p1Ohw_auxiliary  12         4          0
```

In Example 14-8, the **ups_list** command describes the current status of the Uninterruptible Power Supply (UPS) component. It provides details about when the last test was performed and the results. Equally important is the current battery charge level. A non-fully charged battery can be a cause of problems in case of power failure.

The output of the command is broken into two lines for easier reading.

Example 14-8 The ups_list command

```
>> ups_list
Component ID   Status   Currently Functioning   Input Power On   Battery Charge Level   Last Self Test Date
1:UPS:1        OK       yes                     yes              100                    06/23/2009
1:UPS:2        OK       yes                     yes              100                    06/24/2009
1:UPS:3        OK       yes                     yes              100                    06/24/2009

Last Self Test Result   Monitoring Enabled   UPS Status
Passed                  yes                  ON_LINE
Passed                  yes                  ON_LINE
Passed                  yes                  ON_LINE
```

Example 14-9 shows the **switch_list** command that is used to display the current status of the switches.

Example 14-9 The switch_list command

```
>> switch_list
Component ID   Status  Currently Functioning   AC Power State   DC Power State   Interconnect   Failed Fans
1:Switch:1     OK      yes                     OK               OK               Up             0
1:Switch:2     OK      yes                     OK               OK               Up             0
```

The **psu_list** command that is shown in Example 14-10 lists all the power supplies in each of the modules. There is no option to display an individual Power Supply Unit (PSU).

Example 14-10 The psu_list command

```
>> psu_list
Component ID  Status  Currently Functioning  Hardware Status
1:PSU:1:1     OK      yes                    OK
1:PSU:1:2     OK      yes                    OK
1:PSU:2:1     OK      yes                    OK
1:PSU:2:2     OK      yes                    OK
1:PSU:3:1     OK      yes                    OK
1:PSU:3:2     OK      yes                    OK
  .
  .
  .
  .
1:PSU:12:1    OK      yes                    OK
1:PSU:12:2    OK      yes                    OK
1:PSU:13:1    OK      yes                    OK
1:PSU:13:2    OK      yes                    OK
1:PSU:14:1    OK      yes                    OK
1:PSU:14:2    OK      yes                    OK
1:PSU:15:1    OK      yes                    OK
1:PSU:15:2    OK      yes                    OK
```

Events

Events can also be handled with XCLI. Several commands are available to list, filter, close, and send notifications for the events. There are many commands and parameters available. You can obtain detailed and complete information in the *IBM XIV XCLI User Manual*.

Next we illustrate just a few of the several options of the **event_list** command.

Several parameters can be used to sort and filter the output of the **event_list command**. Refer to Table 14-1 for a list of the most commonly used parameters.

Table 14-1 The event_list command parameters

Name	Description	Syntax and example
max_events	Lists a specific number of events	<event_list max_events=100>
after	Lists events after the specified date and time	<event_list after=2008-08-11 04:04:27>
before	Lists events before specified date and time	<event_list before 2008-08-11 14:43:47>
min_severity	Lists events with the specified and higher severities	<event_list min_severity=major>
alerting	Lists events for which an alert was sent or for which no alert was sent	<event_list alerting=no> <event_list alerting=yes>
cleared	Lists events for which an alert was cleared or for which the alert was not cleared	<event_list cleared=yes> <event_list cleared=no>

These parameters can be combined for better filtering. In Example 14-11, two filters were combined to limit the amount of information displayed. The first parameter **max_events** only allows three events to be displayed. The second parameter is the date and time that the events must not exceed. In this case, the event occurred approximately 1.5 minutes before the cutoff time.

Example 14-11 The event_list command with two filters combined

```
>> event_list max_events=5 before=2009-06-29.11:00:00
Timestamp         Severity      Code           User  Description
2009-06-29 09:40:43   Informational   HOST_MULTIPATH_OK          Host 'itso_2008' has redundant connection
2009-06-29 09:53:20   Informational   HOST_MULTIPATH_OK          Host 'isabella' has redundant connections
2009-06-29 10:32:28   Informational   HOST_MULTIPATH_OK          Host 'isabella' has redundant connections
2009-06-29 10:58:25   Informational   HOST_MULTIPATH_OK          Host 'isabella' has redundant connections
2009-06-29 10:58:35   Informational   HOST_MULTIPATH_OK          Host 'isabella' has redundant connections
```

The event list can also be filtered on severity. Example 14-12 displays all the events in the system that contain a severity level of `Major` and all higher levels, such as `Critical`.

Example 14-12 The event_list command filtered on severity

```
>> event_list min_severity=Major max_events=5
Timestamp             Severity   Code                 User   Description
2009-06-26 17:11:36   Major      TARGET_DISCONNECTED         Target named 'XIV MN00035' is no longer accessi
2009-06-26 22:30:33   Major      TARGET_DISCONNECTED         Target named 'XIV MN00035' is no longer accessi
2009-06-28 08:07:41   Major      SWITCH_INTERCONNECT_DOWN    Inter-switch connection lost connectivity on 1
2009-06-28 08:07:41   Major      SWITCH_INTERCONNECT_DOWN    Inter-switch connection lost connectivity on 1
2009-06-29 07:26:58   Critical   MODULE_FAILED               1:Module:9 failed.
```

Certain events generate an alert message and do not stop until the event has been cleared. These events are called *alerting events* and can be viewed by the GUI or XCLI with a separate command. After the alerting event is cleared, it is removed from this list, but it is still visible with the **event_list** command. See Example 14-13.

Example 14-13 The event_list_uncleared command

```
>> event_list_uncleared
No alerting events exist in the system
```

Monitoring statistics

The statistics gathering mechanism is a powerful tool. The XIV Storage System continually gathers performance metrics and stores them internally. Using the XCLI, data can be retrieved and filtered by using many metrics. Example 14-14 provides an example of gathering the statistics for 10 days, with each interval covering an entire day. The system is given a time stamp as the ending point for the data. Due to the magnitude of the data being provided, it is best to redirect the output to a file for further post-processing. Refer to Chapter 13, “Performance characteristics” on page 301 for a more in-depth view of performance.

Example 14-14 Statistics for 10 days

```
>> statistics_get count=10 interval=1 resolution_unit=day end=2009-06-29.17:15:00 > perf.out
```

The **usage_get** command is a powerful tool to provide details about the current utilization of pools and volumes. The system saves the usage every hour for later retrieval. This command works the same as the **statistics_get command**. You specify the time stamp to begin or end the collection and the number of entries to collect. In addition, you need to specify the pool name or the volume name. See Example 14-15.

Example 14-15 The usage_get command by pool

```
>> usage_get pool=ITSO_SVC max=10 start=2009-06-22.11:00:00
Time                  Volume Usage (MB)   Snapshot Usage (MB)
2009-06-22 11:00:00   0                   0
2009-06-22 12:00:00   0                   0
2009-06-22 13:00:00   0                   0
2009-06-22 14:00:00   262144              0
2009-06-22 15:00:00   262144              0
2009-06-22 16:00:00   262144              0
2009-06-22 17:00:00   262144              0
2009-06-22 18:00:00   262144              0
2009-06-22 19:00:00   262144              0
2009-06-22 20:00:00   262144              0
```

Note that the usage is displayed in MB. Example 14-16 shows that the volume `Red_vol_1` is utilizing 78 MB of space. The time when the data was written to the device is also recorded. In this case, the host wrote data to the volume for the first time on 14 August 2008.

Example 14-16 The usage_get command by volume

```
>> usage_get vol=ITSO_SVC_MDISK_01 max=10 start=2009-06-22.11:00:00
Time                  Volume Usage (MB)   Snapshot Usage (MB)
2009-06-22 11:00:00   0                   0
2009-06-22 14:00:00   46                  0
2009-06-22 15:00:00   46                  0
2009-06-22 16:00:00   46                  0
2009-06-22 17:00:00   46                  0
2009-06-22 18:00:00   46                  0
2009-06-22 19:00:00   46                  0
2009-06-22 20:00:00   46                  0
```

```
2009-06-22 21:00:00   46                  0
2009-06-22 22:00:00   46                  0
```

14.1.3 SNMP-based monitoring

So far, we have discussed how to perform monitoring based on the XIV GUI and the XCLI. The XIV Storage System also supports Simple Network Management Protocol (SNMP) for monitoring. SNMP-based monitoring tools, such as IBM Tivoli NetView® or the IBM Director, can be used to monitor the XIV Storage System.

Simple Network Management Protocol (SNMP)

SNMP is an industry-standard set of functions for monitoring and managing TCP/IP-based networks and systems. SNMP includes a protocol, a database specification, and a set of data objects. A set of data objects forms a Management Information Base (MIB).

The SNMP protocol defines two terms, *agent* and *manager*, instead of the client and server terms that are used in many other TCP/IP protocols.

SNMP agent

An SNMP agent is a daemon process that provides access to the MIB objects on IP hosts on which the agent is running. An SNMP agent, or daemon, is implemented in the IBM XIV software and provides access to the MIB objects defined in the system. The SNMP daemon can send SNMP trap requests to SNMP managers to indicate that a particular condition exists on the agent system, such as the occurrence of an error.

SNMP manager

An SNMP manager can be implemented in two ways. An SNMP manager can be implemented as a simple command tool that can collect information from SNMP agents. An SNMP manager also can be composed of multiple daemon processes and database applications. This type of complex SNMP manager provides you with monitoring functions using SNMP. It typically has a graphical user interface for operators. The SNMP manager gathers information from SNMP agents and accepts trap requests sent by SNMP agents. In addition, the SNMP manager generates traps when it detects status changes or other unusual conditions while polling network objects. IBM Director is an example of an SNMP manager with a GUI interface.

SNMP trap

A *trap* is a message sent from an SNMP agent to an SNMP manager without a specific request from the SNMP manager. SNMP defines six generic types of traps and allows you to define enterprise-specific traps. The trap structure conveys the following information to the SNMP manager:

- Agent's object that was affected
- IP address of the agent that sent the trap
- Event description (either a generic trap or enterprise-specific trap, the including trap number)
- Time stamp
- Optional enterprise-specific trap identification
- List of variables describing the trap

SNMP communication

The SNMP manager sends SNMP get, get-next, or set requests to SNMP agents, which listen on UDP port 161, and the agents send back a reply to the manager. The SNMP agent can be implemented on any kind of IP host, such as UNIX workstations, routers, network appliances, and also on the XIV Storage System.

You can gather various information about the specific IP hosts by sending the SNMP get and get-next requests, and you can update the configuration of IP hosts by sending the SNMP set request.

The SNMP agent can send SNMP trap requests to SNMP managers, which listen on UDP port 162. The SNMP trap requests sent from SNMP agents can be used to send warning, alert, or error notification messages to SNMP managers. Figure 14-10 on page 325 illustrates the characteristics of SNMP architecture and communication.

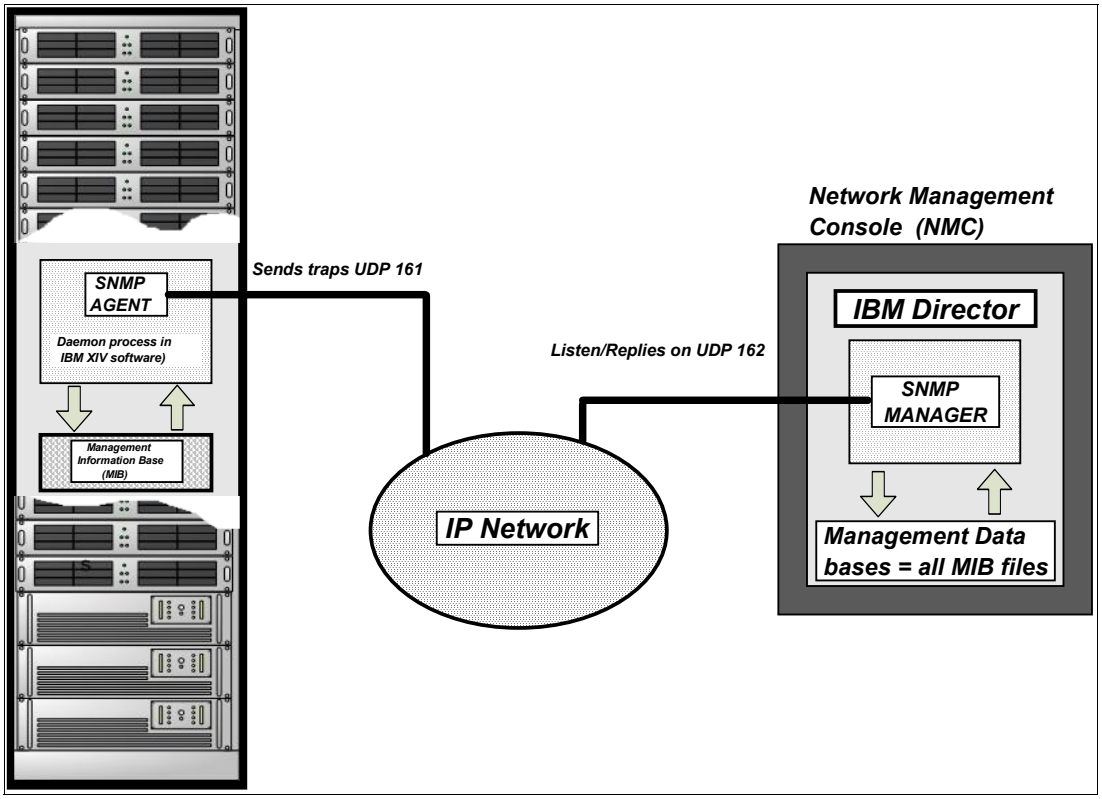


Figure 14-10 SNMP communication

You can configure an SNMP agent to send SNMP trap requests to multiple SNMP managers.

Management Information Base (MIB)

The objects, which you can get or set by sending SNMP get or set requests, are defined as a set of databases called the *Management Information Base* (MIB). The structure of MIB is defined as an Internet standard in RFC 1155. The MIB forms a tree structure.

Most hardware and software vendors provide you with extended MIB objects to support their own requirements. The SNMP standards allow this extension by using the private sub-tree, which is called an enterprise-specific MIB. Because each vendor has a unique MIB sub-tree under the private sub-tree, there is no conflict among vendors' original MIB extensions.

The XIV Storage System comes with its own specific MIB.

XIV SNMP setup

To effectively use SNMP monitoring with the XIV Storage System, you must first set it up to send SNMP traps to an SNMP manager (such as the IBM Director server), which is defined in your environment. Figure 14-11 illustrates where to start to set up the SNMP destination. Also, you can refer to “Setup notification and rules with the GUI” on page 341.

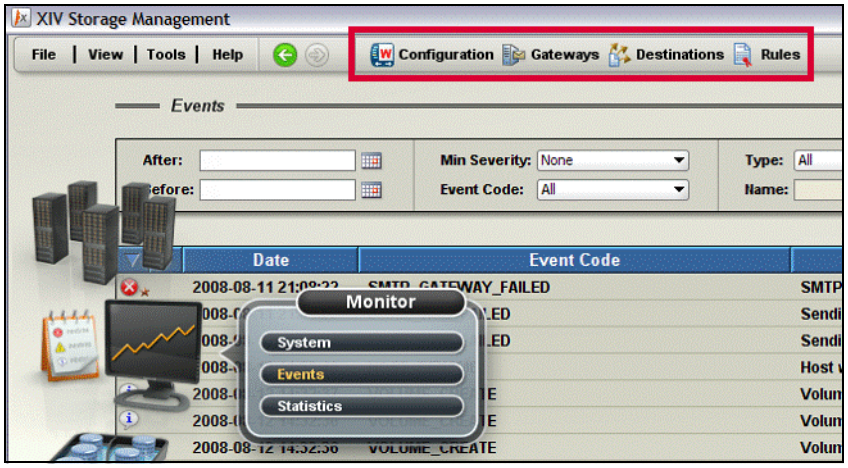


Figure 14-11 Configure destination

Configuring a new destination

Follow these steps:

1. From the XIV GUI main window, select the Monitor icon.
2. From the Monitor menu, select **Events** to display the Events window as shown in Figure 14-11.

   From the toolbar:

   a. Click **Destinations**. The Destinations dialog window opens.

   b. Select **SNMP** from the Destinations pull-down list.

   c. Click the green plus sign (+) and select **Destination** from the pop-up menu to add a destination as illustrated in Figure 14-12.

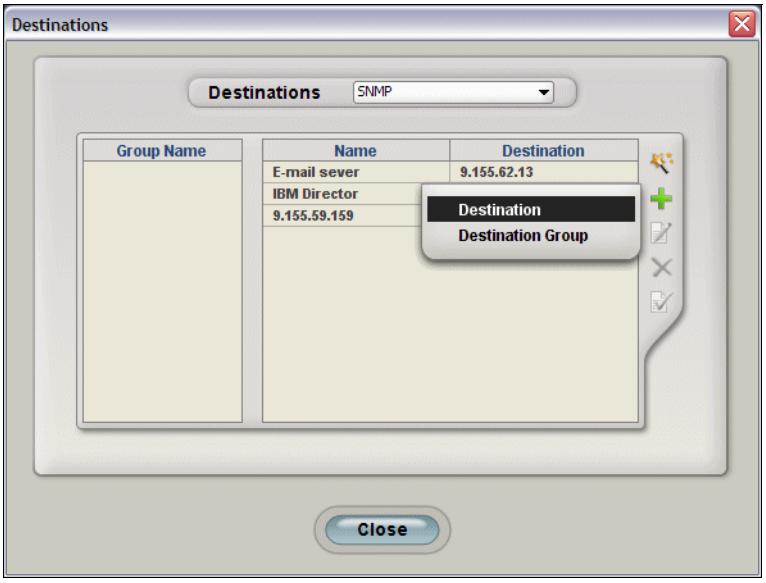


Figure 14-12 Add SNMP destination

3. The Define Destination dialog is now open. Enter a Destination Name (a unique name of your choice) and the IP or Domain Name System (DNS) of the server where the SNMP Management software is installed. Refer to Figure 14-13 on page 327.

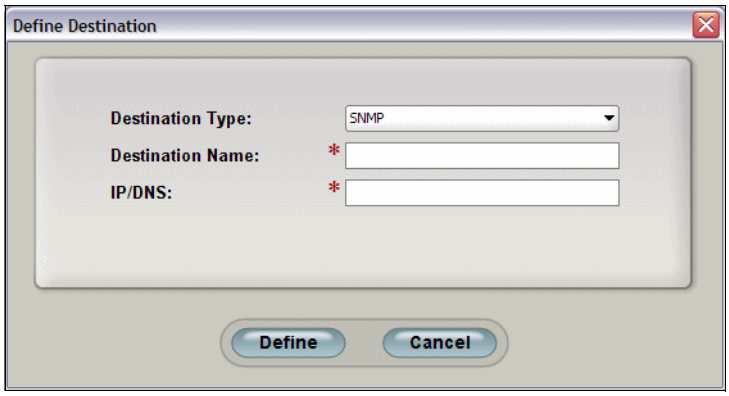


Figure 14-13 Define SNMP destination

4. Click **Define** to effectively add the SNMP Manager as a destination for SNMP traps.

Your XIV Storage System is now set up to send SNMP Traps to the defined SNMP manager. The SNMP Manager software will process the received information (SNMP traps) according to the MIB file.

Using IBM Director

In this section, we illustrate how to use IBM Director to monitor the XIV Storage System. The IBM Director is an example of a possible SNMP manager for XIV. Other SNMP Managers can be used with XIV as well.

IBM Director provides an integrated suite of software tools for a consistent, single point of management and automation. With IBM Director, IT administrators can view and track the hardware configuration of remote systems in detail and monitor the usage and performance of critical components, such as processors, disks, and memory.

All IBM clients can download the latest version of IBM Director code from the IBM Director Software Download Matrix page:

http://www.ibm.com/systems/management/director/downloads.html

For detailed information regarding the installation, setup, and configuration of IBM Director, refer to the documentation available at:

http://www.ibm.com/systems/management/director/

Compile MIB file

After you have completed the installation of your IBM Director environment, you prepare it to manage the XIV Storage System by compiling the provided MIB file.

Make sure to always use the latest MIB file provided. To compile the MIB file in your environment:

1. At the IBM Director Console window, click **Tasks** → **SNMP Browser** → **Manage MIBs**, as shown in Figure 14-14.

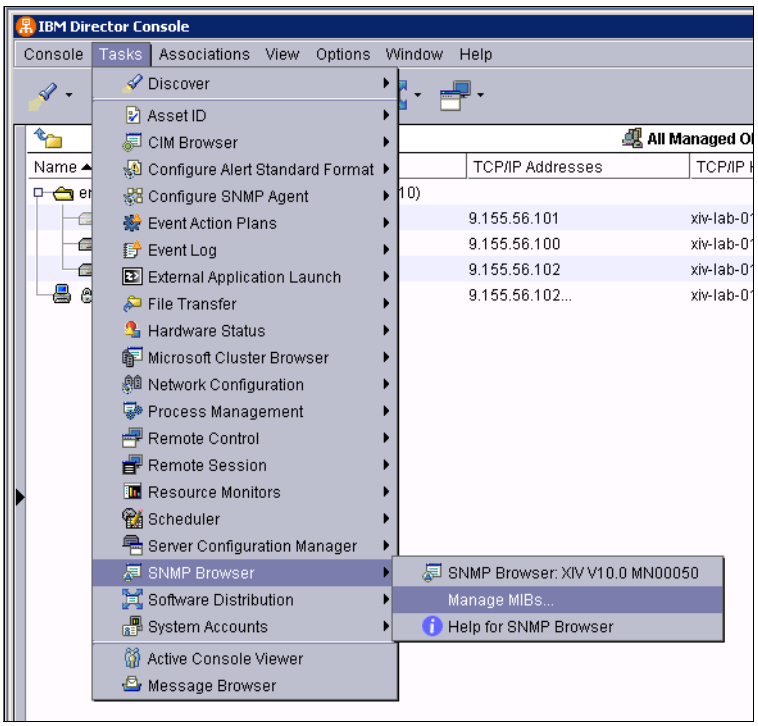


Figure 14-14 Manage MIBs

2. In the MIB Management window, click **File** → **Select MIB to Compile.**
3. In the Select MIB to Compile window that is shown in Figure 14-15, specify the directory and file name of the MIB file that you want to compile, and click **OK**. A status window indicates the progress.

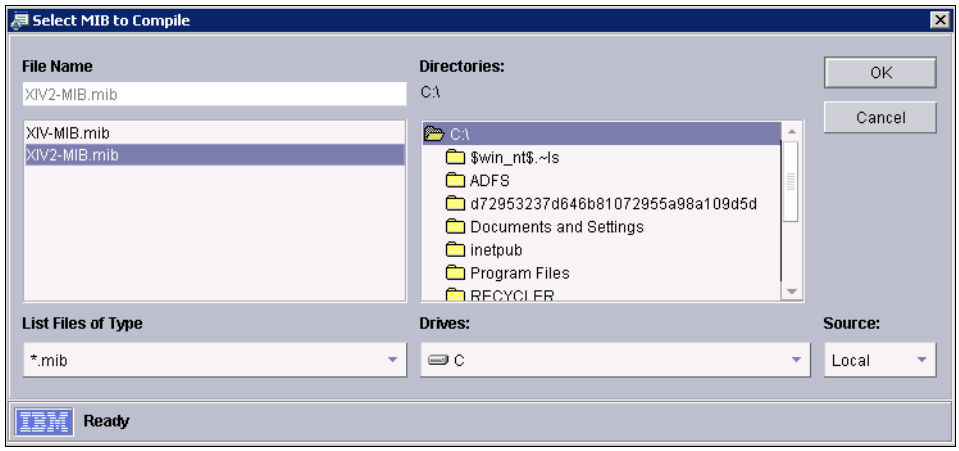


Figure 14-15 Compile MIB

When you compile a new MIB file, it is also automatically loaded in the *Loaded MIBs file directory* and is ready for use.

To load an already compiled MIB file:

- In the MIB Management window, click **File** → **Select MIB** to load.
- Select the MIB (that you to load) in the Available MIBs window, click **Add**, **Apply**, and **OK**.

This action will load the selected MIB file, and the IBM Director is ready to be configured for monitoring the IBM XIV.

Discover the XIV Storage System

After loading the MIB file into the IBM Director, the next step is to discover the XIV Storage Systems in your environment. Therefore, configure the IBM Director for auto-discover.

1. From the IBM Director Console window, select **Options** → **Discovery Preferences**, as shown in Figure 14-16.

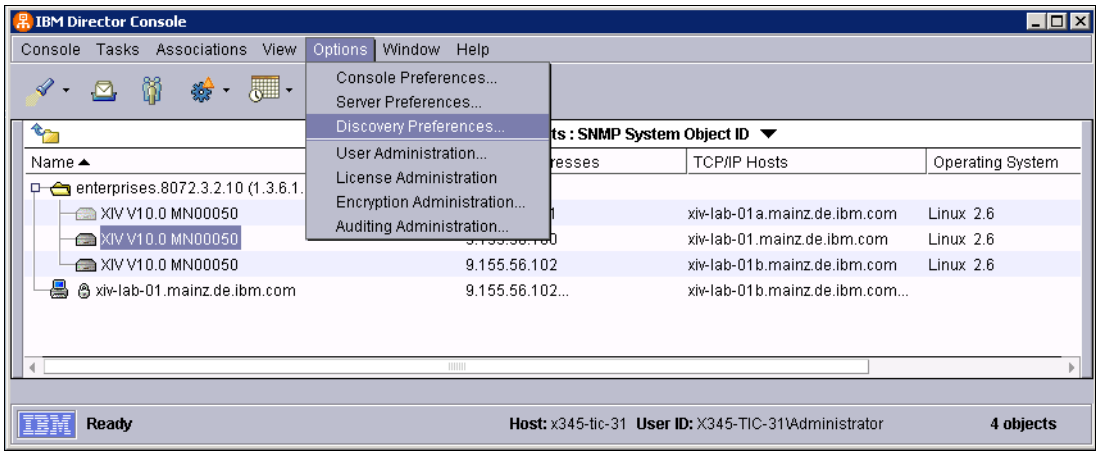


Figure 14-16 Discovery Preferences

2. In the Discovery Preferences window that is shown in Figure 14-17, follow these steps to discover XIV Storage Systems:

   a. Click the **Level 0: Agentless System** tab.

   b. Click **Add** to bring up a window to specify whether you want to add a single address or an address range. Select **Unicast Range**.

   Note: Because each XIV system is presented through three IP addresses, select Unicast Range when configuring the auto-discovery preferences.

   c. Next, enter the address range for the XIV systems in your environment. You also set the Auto-discover period and the Presence Check period.

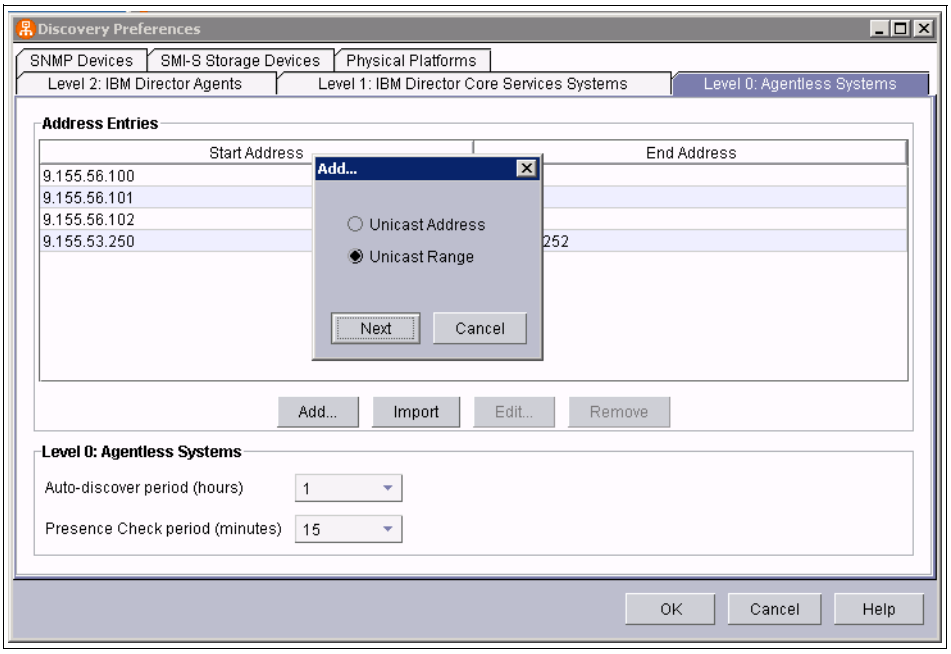


Figure 14-17 Discover Range

3. After you have set up the Discovery Preferences, the IBM Director will discover the XIV Storage Systems and add them to the IBM Director Console as seen in Figure 14-18.

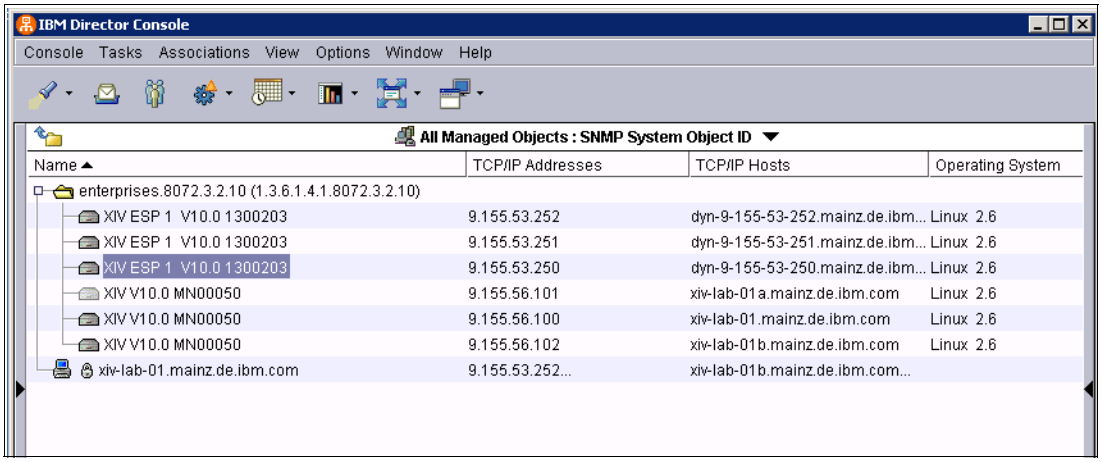


Figure 14-18 IBM Director Console

At this point, the IBM Director and IBM Director Console are ready to receive SNMP traps from the discovered XIV Storage Systems.

With the IBM Director, you can display general information about your IBM XIVs, monitor the Event Log, and browse more information.

General System Attributes

Double-click the entry corresponding to your XIV Storage System in the IBM Director Console window to display the General System Attributes as illustrated in Figure 14-19. This window gives you a general overview of the system status.

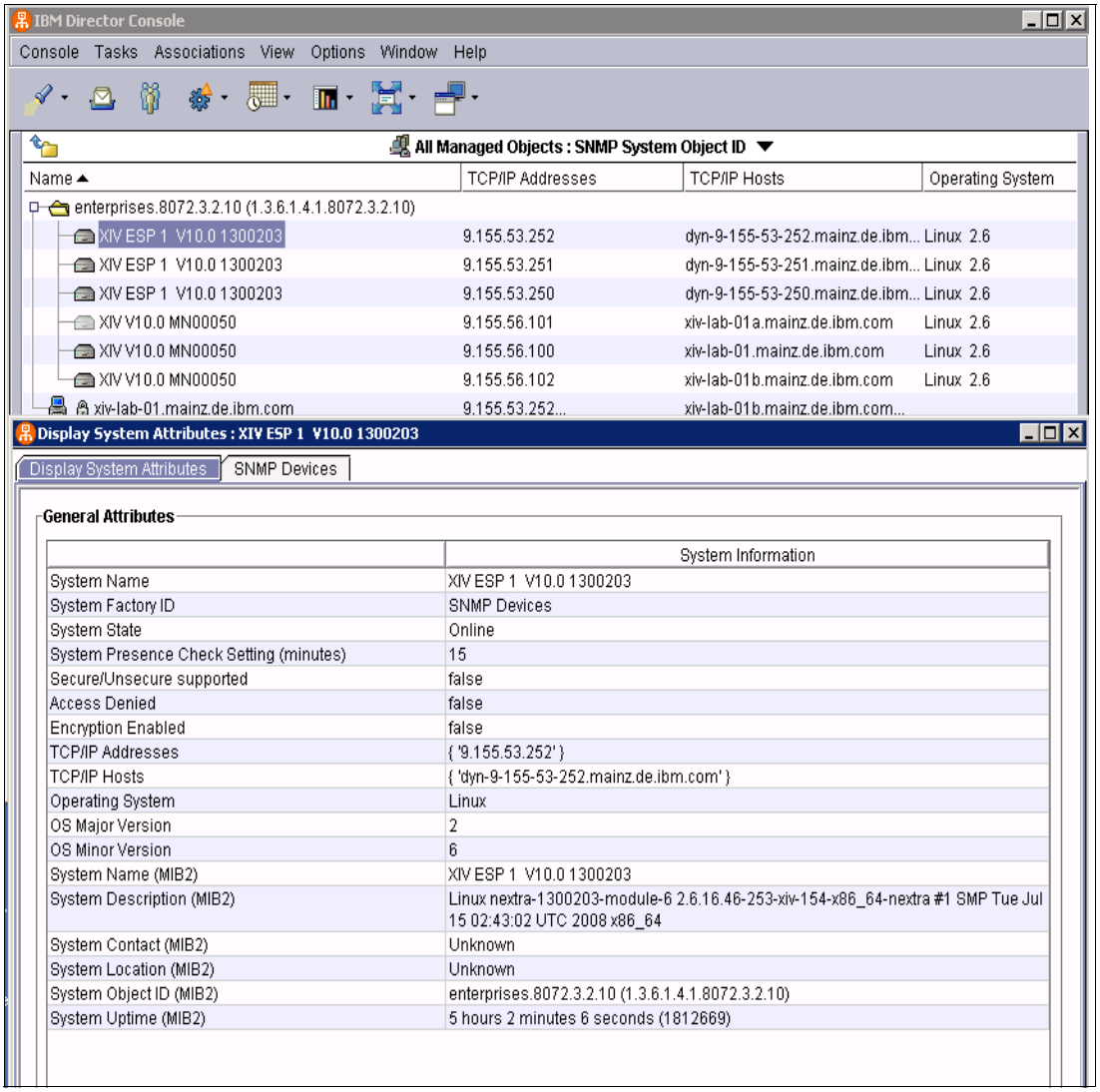


Figure 14-19 General System Attributes

Event Log

To open the Event Log, right-click the entry corresponding to your XIV Storage System and select Event Log from the pop-up menu that is shown in Figure 14-20.

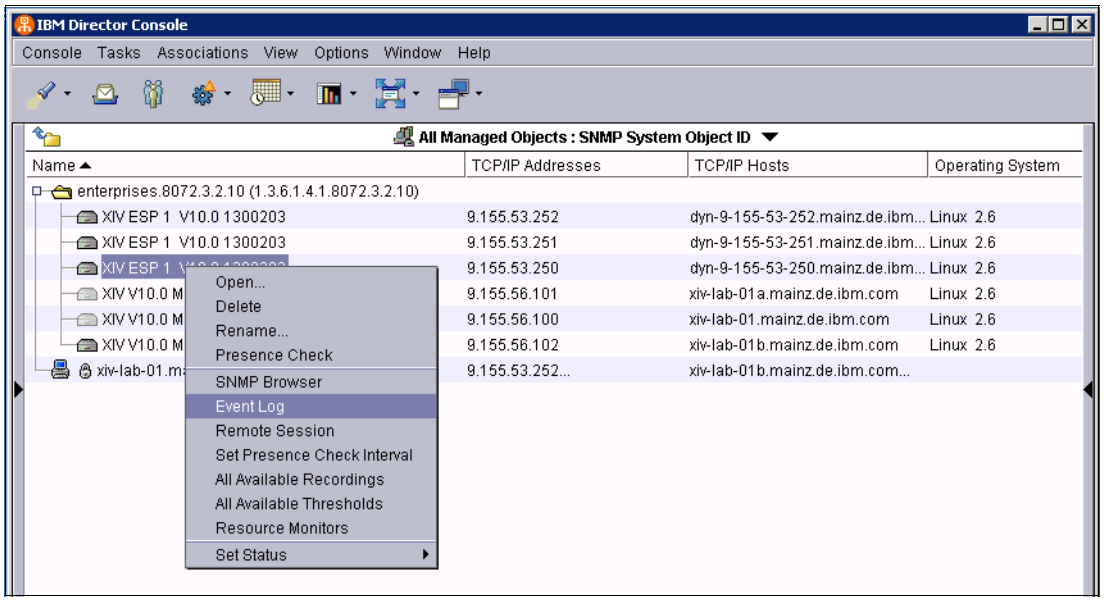


Figure 14-20 Select Event Log

The Event Log window can be configured to show the events for a defined time frame or to limit the number of entries to display. Selecting a specific event will display the Event Details in a pane on the right side of the window as shown in Figure 14-21.

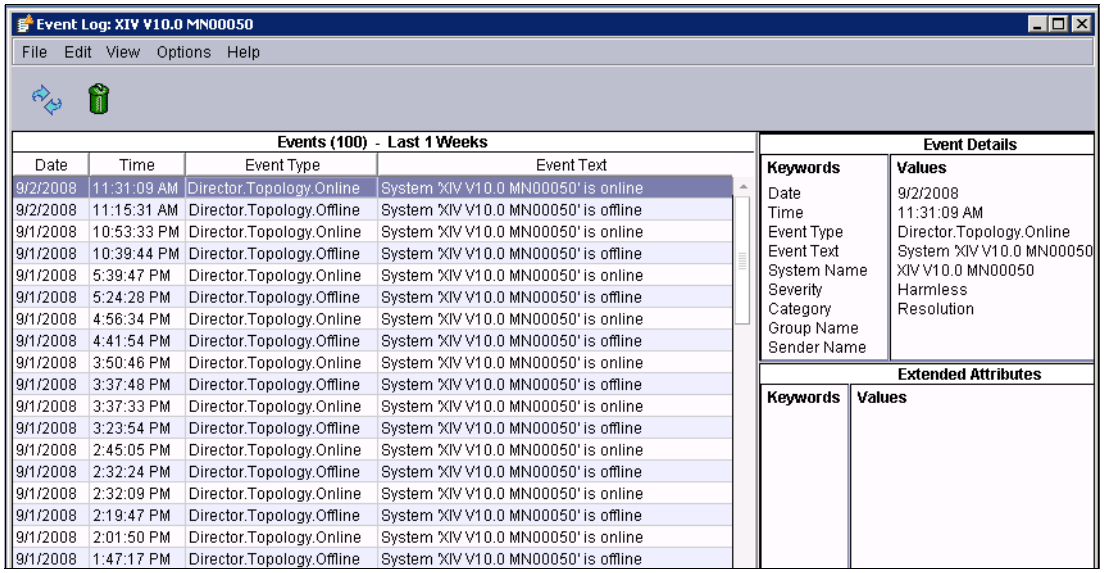


Figure 14-21 IBM Director Event Log

Event actions

Based on the SNMP traps and the events, you can define different Event Actions with the Event Actions Builder as illustrated in Figure 14-22. Here, you can define several actions for the IBM Director to perform in response to specific traps and events.

IBM Director offers a wizard to help you define an Event Action Plan. Start the wizard by selecting **Tasks** → **Event Action Plans** → **Event Action Plan Wizard** in the IBM Director Console window. The Wizard will guide you through the setup.

The window in Figure 14-22 shows that the IBM Director will send, for all events, an e-mail (to a predefined e-mail address or to a group of e-mail addresses).

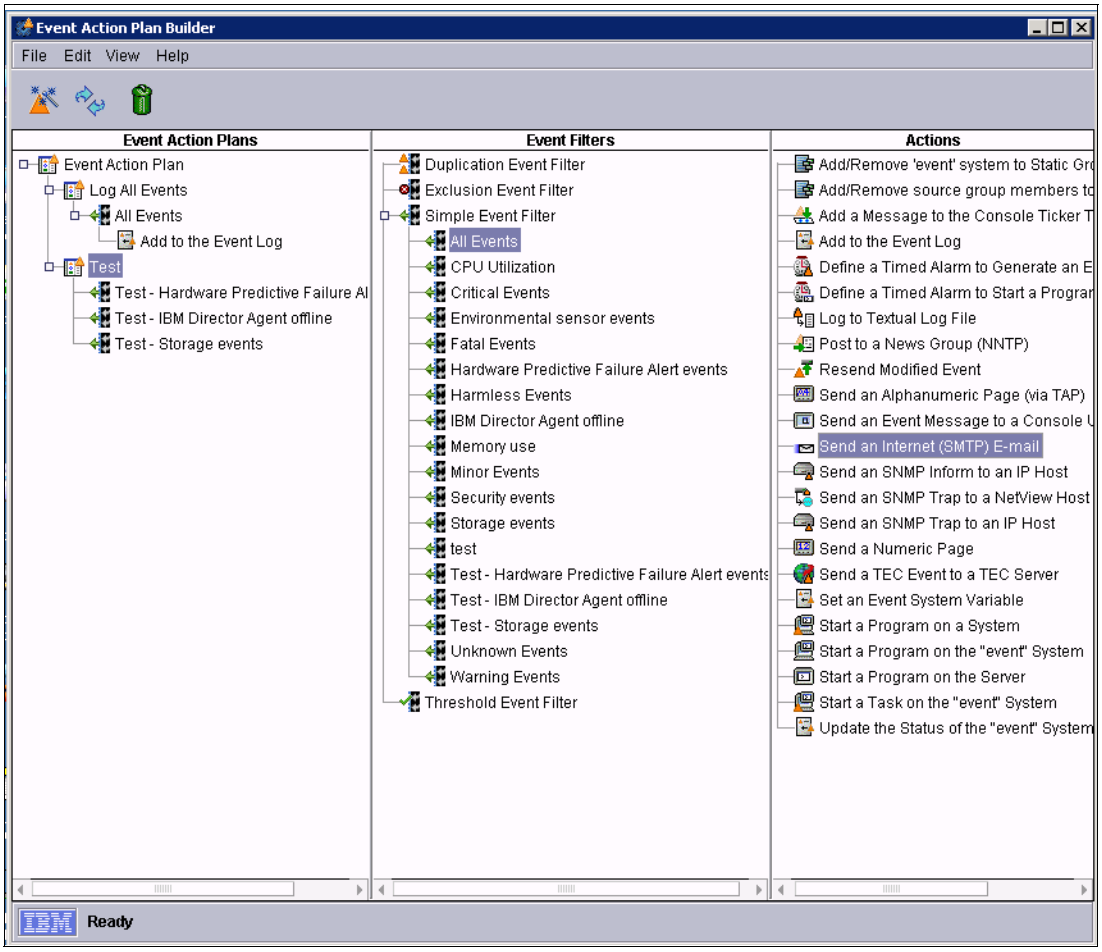


Figure 14-22 Event Action Plan Builder

14.1.4 Using Tivoli Storage Productivity Center

Starting with version 10.1 of the software, XIV supports integration with the Tivoli Storage Productivity Center (TPC) v4.1 or higher.

For detailed information about TPC 4.1, refer to the IBM Redbooks publication, *IBM Tivoli Storage Productivity Center V4.1 Release Guide*, SG24-7725.

TPC is an integrated suite for managing storage systems, capacity, storage networks, even replication. The IBM XIV Storage system includes a Common Information Model Object Manager (CIMOM) agent (SMI-S compliant) that can be used directly by TPC 4.1. The built-in agent provides increased performance and reliability and makes it easier to integrate XIV in a TPC environment.

The CIM agent provides detailed information regarding the configuration of the XIV device, the Storage Pools, Volumes, Disks, Hosts and Host Mapping as well as the device itself. It also provides information on the CIM agent service. TPC collects the information and stores in the TPC database.

For now, TPC can only perform read operations against XIV.

Set up and discover XIV system in TPC

TPC manages and monitors the XIV through its CIM agent (embedded in the XIV code).

Discovery phase

The the first step in managing an XIV device by TPC is the process of discovering XIV CIM agents and XIVs managed by these CIM agents.

The XIV CIM agent will publish itself as a service within the SLP Service Agent (SA). This agent broadcasts its address to allow a directory look-up of the CIM agents that have registered with it. This then allows TPC to query for the IP address, namespace, and supported profiles for the XIV CIM agent, thus discovering it.

If only specific devices should be monitored by TPC, we recommend to disable the automatic discovery. This is done by deselecting the field **Scan local subnet** as shown in Figure 14-23.

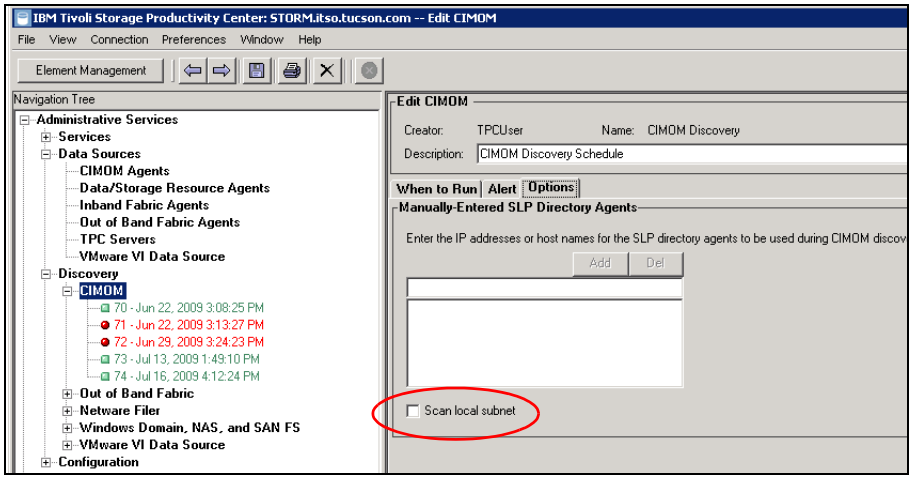


Figure 14-23 Deselecting autodiscovery of the CIM agents

The CIMOM discovery usually takes a few minutes. The CIMOM discovery can be run on a schedule. How often you run it depends on how dynamic your environment is. It must be run to detect a new subsystem. The CIMOM discovery also performs basic health checks of the CIMOM and subsystem.

For a manual discovery, perform the following steps to set up the CIMOM from TPC:

1. Select **Administrative Services** → **Data Sources** → **CIMOM Agents** and select **Add CIMOM**.

2. As shown in Figure 14-4, enter the required information:
   - Host: The IP address of the CIMOM. For XIV, this corresponds to the IP address or a fully qualified domain name of the XIV system
   - Port: The port on which the CIMOM is connected. By default, this is 5989 for a secure connection and 5988 for an unsecured connection. For XIV, use the default value of 5989
   - Interoperability Namespace: the CIM namespace for this device. For example: “\root\ibm”.
   - Display Name: The name of the CIMOM, as specified by the CIMOM provider. This name will appear in the IBM Tivoli Storage Productivity Center interface.
   - Description: The optional description for the CIM agent.
3. Click **Save** to add the CIMOM to the list and test availability of the connection.

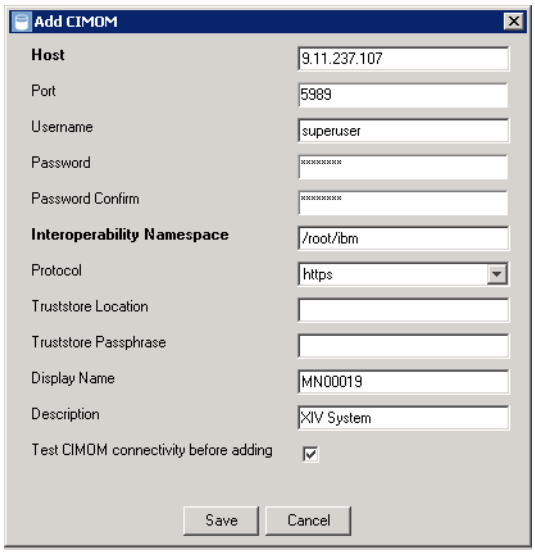


Figure 14-24 Define XIV CIMOM in TPC

4. When the test has completed, the new CIMON is added to the list as shown in Figure 14-25.

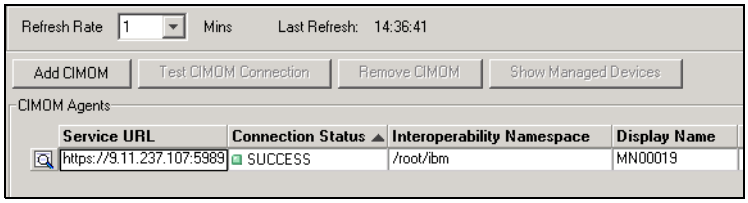


Figure 14-25 CIMOM listed

After the CIMOM has been added to the list of devices, the initial CIMOM discovery can be executed:

1. Go to **Administrative Services** → **Discovery**.
2. Deselect the field **Scan Local Subnet** in the folder Options.
3. Select **When to Run** → **Run Now**.
4. Save this setting to start the CIMOM discovery.

The initial CIMOM discovery will be listed in the Navigation Tree. Selecting this entry allows you to verify the progress of the discovery and the details about actions done while probing the systems. After the discovery has completed, the entry in the navigation tree will change from blue to green or red depending on the success (or not) of the discovery.

After the initial setup action, future discoveries should be scheduled. As shown in Figure 14-26, this can be set up by the following actions:

1. Specify the start time and frequency on the When to Run tab.
2. Select **Run Repeatedly**.
3. Save the configuration.

The CIMOM discoveries will now run at the time intervals configured.

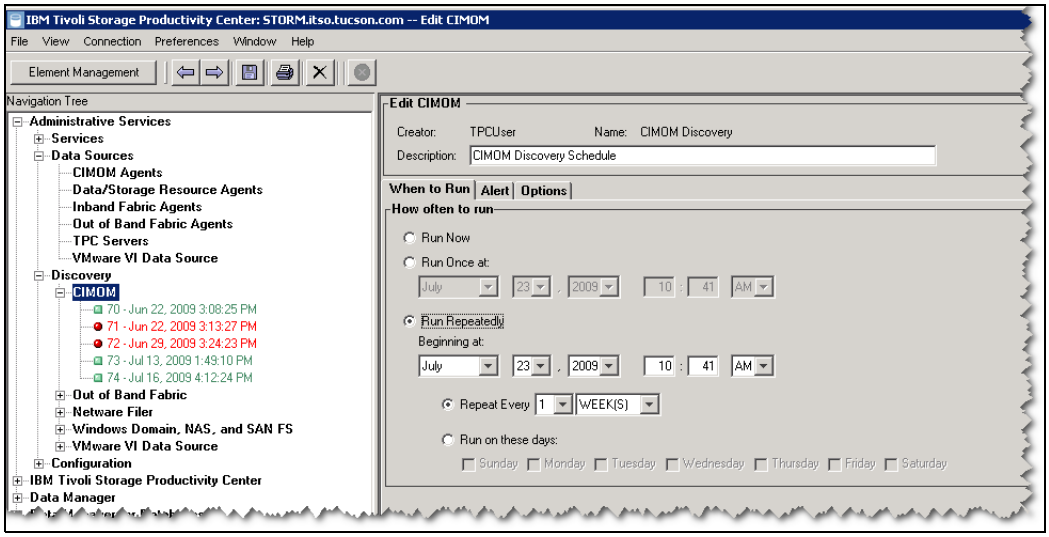


Figure 14-26 Setup repeatable CIMOM Discoveries

Probing phase

After TPC has been made aware of the XIV CIMOM, the storage subsystem must be probed to collect detailed information. Probes use agents to collect statistics, including data about drives, pools, volumes. The results of probe jobs are stored in the repository and are used in TPC to supply the data necessary for generating a number of reports, including Asset, Capacity, and Storage Subsystem reports.

To configure a probe, from TPC, perform the following steps:

1. Go to **IBM Total Productivity Center** → **Monitoring**.
2. Right-click **Probes** and select **Create Probe**.
3. In the next window (Figure 14-27) specify the systems to probe in the What to Probe tab. To add a system to a probe, double-click the subsystem name to add it to the Current Selection list.
4. Select when to probe the system, assign a name to the probe, and save the session.

Tip: Configure individual probes for every XIV system, but set them to run at different times.

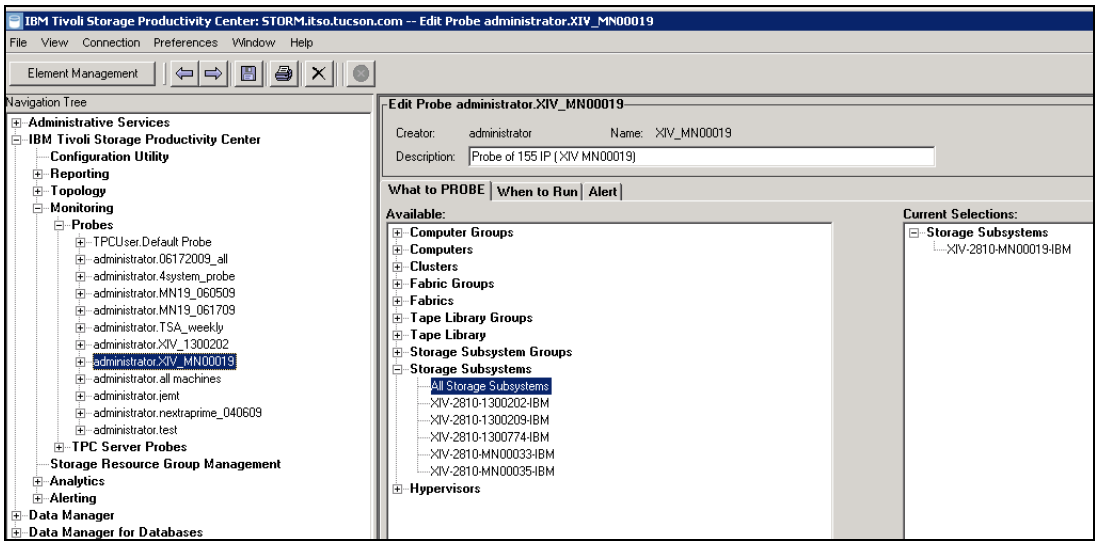


Figure 14-27 Configuring a new Probe

The CIM agent uses the smis_user, a predefined XIV user with read-only access, to gather capacity and configuration data from the XIV Storage system.

Configuration information and reporting

In Figure 14-28 you can see the a list of several XIV subsystems as reported in TPC.

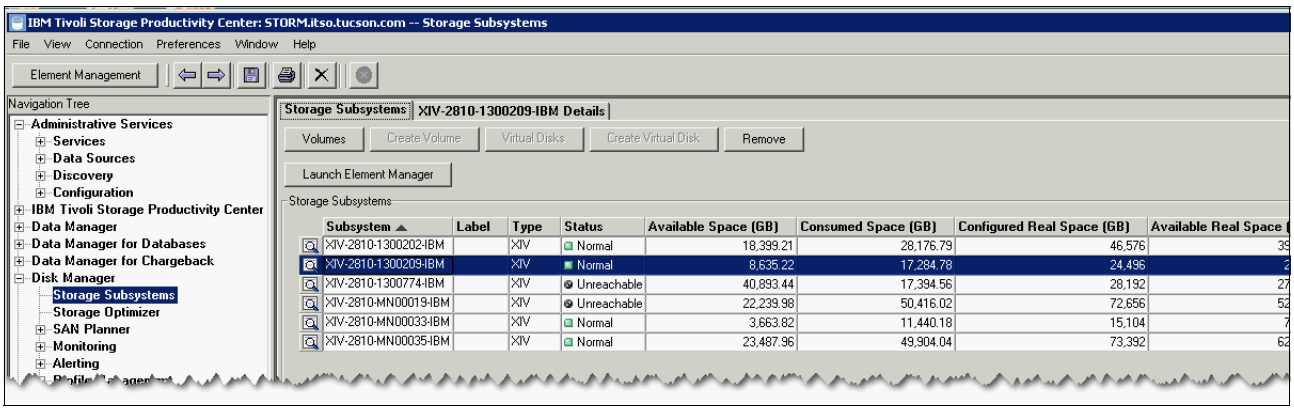


Figure 14-28 List storage subsystems

XIV Storage Subsystem TPC reports

Tivoli Storage Productivity Center version 4.1 includes basic capacity and asset information in tabular reports as well as in Topology Viewer. In addition, LUN Correlation information is available.

TPC probes collect the following information from XIV systems:

- Storage Pools
- Volumes
- Disks
- Ports
- Host definitions, LUN Mapping & Masking information

Note: Space is calculated differently in the XIV Graphical User Interface (GUI) and the Command Line Interface (CLI) than in TPC. XIV defines 1 Gigabyte as 10^9 = 1,000,000,000 Bytes, while TPC defines 1 Gigabyte as 2^{30} = 1,073,741,824 Bytes.

This is why capacity information might seem different (wrong) when comparing XIV GUI with TPC GUI, when in fact it is the exact same size.

Because the XIV Storage Subsystems provide thin provisioning by default, additional columns for the thin provisioning properties of Volumes, Pools, and Subsystems were introduced to the TPC GUI.

Note that the TPC terminology of *configured space* accords with XIV's terminology of soft' capacity, while TPC terminology of *real space* accords with XIV's terminology of hard space.

Additional *Configured Real Space* and *Available Real Space* columns were introduced to report on the hard capacity of a subsystem, while the pre-existent *Consumed Space* and *Available Space* columns now report on the soft capacity of a subsystem in the following reports:

- Storage Subsystem list under **Disk Manager** → **Storage Subsystems**
- Storage Subsystem Details panel under **Disk Manager** → **Storage Subsystems**
- Storage Subsystem Details panel under **Data Manager** → **Reporting** → **Asset** → **By Storage Subsystem**
- **Data Manager** → **Reporting** → **Asset** → **System-wide** → **Storage Subsystems**
- **Data Manager** → **Reporting** → **TPC-wide Storage Space** → **Disk Space** → **By Storage Subsystem (Group)**

See Figure 14-29 for an example of the Storage Subsystem Details panel:

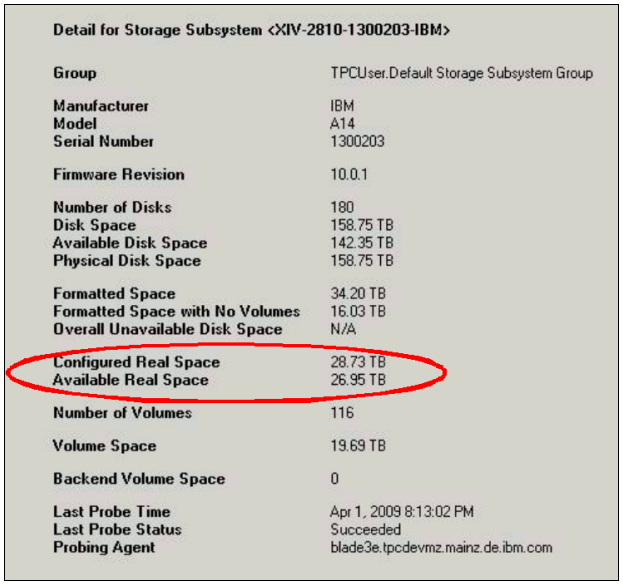


Figure 14-29 Storage Subsystem Details panel

Configured Real Space and *Available Real Space* columns, reporting on the hard capacity of a storage pool, were also added to the following report:

- Storage Pool Details panel under **Data Manager** → **Reporting** → **Asset** → **By Storage Subsystem** → **<Subsystem Name>** → **Storage Pools**

See Figure 14-30 for an example of the Storage Pool Details panel:

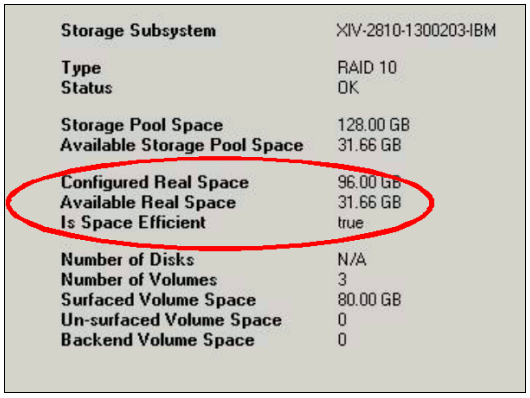


Figure 14-30 Storage Pool Details panel

A *Volume Real Space* column was added to report on the hard capacity of a volume, while the pre-existent *Volume Space* columns report on the soft capacity of a volume in the following reports:

- Volume Details panel under **Disk Manager** → **Storage Subsystems** → **Volumes**
- **Disk Manager** → **Reporting** → **Storage Subsystems** → **Volumes**
- **Disk Manager** → **Reporting** → **Storage Subsystems** → **Volume to HBA Assignment**
- Added **Backend Volume Real Space** for XIV volumes as backend volumes under **Disk Manager** → **Reporting** → **Storage Subsystems** → **Volume to Backend Volume Assignment**
- Volume Details panel under **Data Manager** → **Reporting** → **Asset** → **By Storage Subsystem** → **<Subsystem Name>** → **Volumes**
- **Data Manager** → **Reporting** → **Asset** → **System-wide** → **Volumes**

See Figure 14-31 for an example of the Volume Details panel:

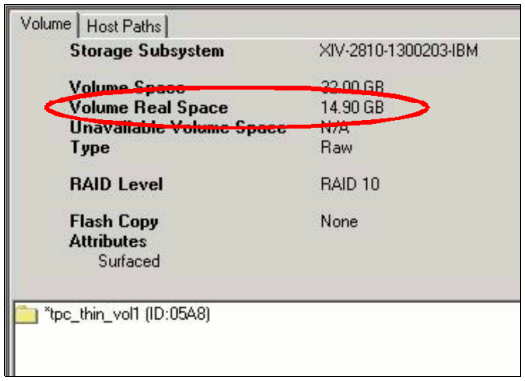


Figure 14-31 Volume Details panel

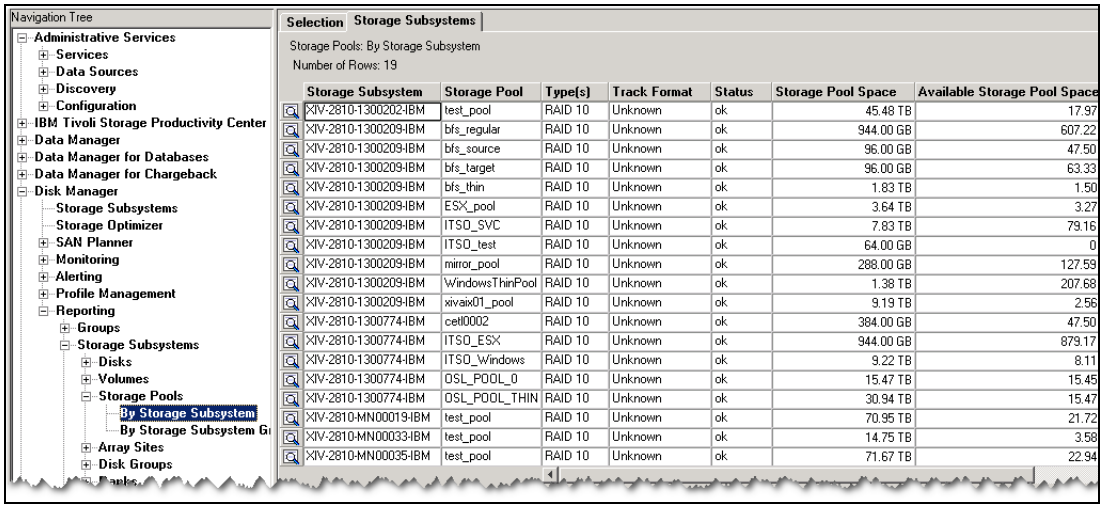
Due to the XIV architecture and the fact that each volume resides on all disks, some of the reports in the TPC GUI will not provide meaningful information for XIV Storage Subsystems. Correlation of disks and volumes, for example under the **Data Manager** → **Reporting** → **Asset** → **By Storage Subsystem** branch, is not possible - TPC will not report any volumes under the branch of a particular disk.

Also, because XIV Storage pools are used to group volumes but not disks, no disks will be reported for a particular storage pool under the reporting branch mentioned above.

Finally, the following reports will not contain any information for XIV Storage Subsystems:

- **Disk Manager → Reporting → Storage Subsystems → Computer Views → By Computer** (Relate Computers to Disks)
- **Disk Manager → Reporting → Storage Subsystems → Computer Views → By Computer Group** (Relate Computers to Disks)
- **Disk Manager → Reporting → Storage Subsystems → Computer Views → By Filesystem/Logical Volume** (Relate Filesystems/Logical Volumes to Disks)
- **Disk Manager -> Reporting -> Storage Subsystems -> Computer Views -> By Filesystem Group** (Relate Filesystems/Logical Volumes to Disks)
- **Disk Manager -> Reporting -> Storage Subsystems -> Storage Subsystem Views -> Disks** (Relate Disks to Computers)

Figure 14-32 illustrates how TPC can report on XIV storage p[ools.



Storage Subsystem	Storage Pool	Type(s)	Track Format	Status	Storage Pool Space	Available Storage Pool Space
XIV-2810-1300202-IBM	test_pool	RAID 10	Unknown	ok	45.48 TB	17.97
XIV-2810-1300209-IBM	bfs_regular	RAID 10	Unknown	ok	944.00 GB	607.22
XIV-2810-1300209-IBM	bfs_source	RAID 10	Unknown	ok	96.00 GB	47.50
XIV-2810-1300209-IBM	bfs_target	RAID 10	Unknown	ok	96.00 GB	63.33
XIV-2810-1300209-IBM	bfs_thin	RAID 10	Unknown	ok	1.83 TB	1.50
XIV-2810-1300209-IBM	ESX_pool	RAID 10	Unknown	ok	3.64 TB	3.27
XIV-2810-1300209-IBM	ITSO_SVC	RAID 10	Unknown	ok	7.83 TB	79.16
XIV-2810-1300209-IBM	ITSO_test	RAID 10	Unknown	ok	64.00 GB	0
XIV-2810-1300209-IBM	mirror_pool	RAID 10	Unknown	ok	288.00 GB	127.59
XIV-2810-1300209-IBM	WindowsThinPool	RAID 10	Unknown	ok	1.38 TB	207.68
XIV-2810-1300209-IBM	xivaix01_pool	RAID 10	Unknown	ok	9.19 TB	2.56
XIV-2810-1300774-IBM	cetl0002	RAID 10	Unknown	ok	384.00 GB	47.50
XIV-2810-1300774-IBM	ITSO_ESX	RAID 10	Unknown	ok	944.00 GB	879.17
XIV-2810-1300774-IBM	ITSO_Windows	RAID 10	Unknown	ok	9.22 TB	8.11
XIV-2810-1300774-IBM	OSL_POOL_0	RAID 10	Unknown	ok	15.47 TB	15.45
XIV-2810-1300774-IBM	OSL_POOL_THIN	RAID 10	Unknown	ok	30.94 TB	15.47
XIV-2810-MN00019-IBM	test_pool	RAID 10	Unknown	ok	70.95 TB	21.72
XIV-2810-MN00033-IBM	test_pool	RAID 10	Unknown	ok	14.75 TB	3.58
XIV-2810-MN00035-IBM	test_pool	RAID 10	Unknown	ok	71.67 TB	22.94

Figure 14-32 XIV Storage pools seen in TPC

These queries, when combined with SCSI inquiry data TPC collects from the hosts, allow TPC to correlate LUNs reported by the IBM XIV Storage System to LUNs as seen by the host systems.

Also, when the IBM XIV Storage System is providing storage to the IBM System Storage SAN Volume controller (SVC), TPC can correlate LUNs reported by the IBM XIV Storage System to SVC managed disks (MDisks).

Element Manager Launch

If the XIV GUI is installed on the TPC server, TPC provides the ability to launch the XIV management software by simply clicking the “Launch Element Manager” button.

14.2 XIV event notification

The XIV Storage System allows you to send alerts via email and SMS messages. You can configure the system using very flexible rules to ensure that notification is sent to the correct person, or group of people, according to the several different parameters. This event notification is similar to, but not quite the same as XIV Call Home, discussed in 14.3, “Call Home and Remote support”

Setting up event notification

Configuration options are available from the XIV GUI. You have the flexibility to create a detailed events notification plan based on specific rules. This flexibility allows the storage administrator to decide, for instance, where to direct alerts for various event types. All these settings can also be done with XCLI commands.

Setup notification and rules with the GUI

To set up e-mail or SMS notification and rules:

1. From the XIV GUI main window, select the Monitor icon. From the Monitor menu, select **Events** to display the Events window as shown in Figure 14-33.

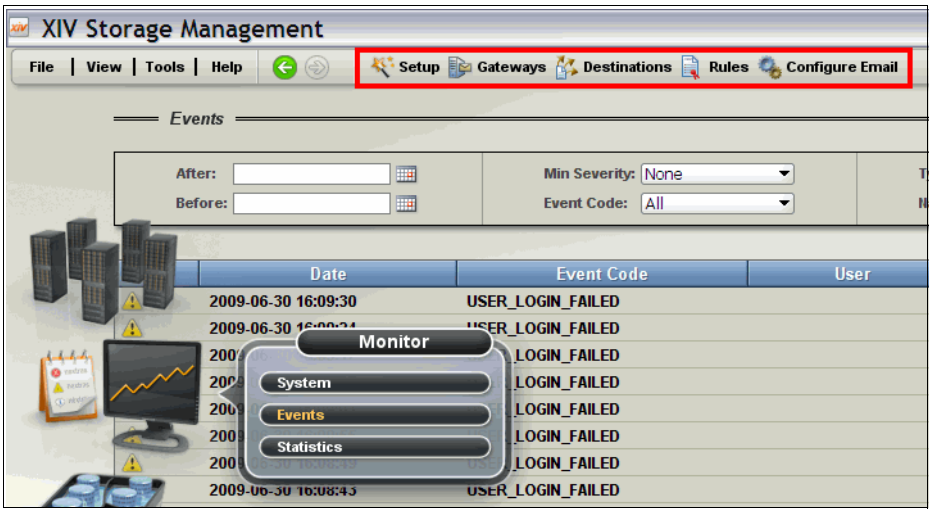


Figure 14-33 Setup notification and rules

2. From the toolbar, click **Setup** to invoke the Events Configuration wizard. The wizard will guide you through the configuration of Gateways, Destinations, and Rules.
3. The wizard Welcome window is shown in Figure 14-34.

Figure 14-34 Events Configuration wizard

Gateways

The wizard will first take you through the configuration of the gateways. Click **Next** or **Gateway** to display the Events Configuration - Gateway dialog as shown in Figure 14-35.

Figure 14-35 Define Gateway panel

Click **Define Gateway**. The Gateway Create Welcome panel shown in Figure 14-36 appears. Click **Next**.

Figure 14-36 Configure Gateways

The Gateway Create - Select gateway type panel displays as shown in Example 14-37.

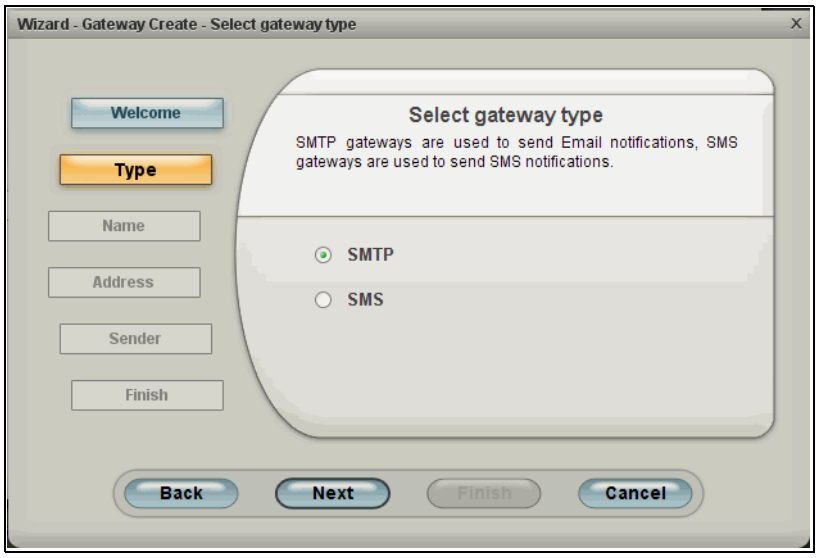


Figure 14-37 Select gateway type

The wizard is asking for the type of the gateway, either SMTP for e-mail notification or SMS if an alert or information will initiate an SMS. Click either SMTP or SMS.

The next steps differ for SMTP and SMS. Our illustration from now on is for SMTP. However, the steps to go through for SMS are similarly self-explanatory. Click **Next**.

Enter the gateway name of the SMTP Gateway, click **Next**. Enter the IP address or DNS name of the SMTP gateway for the gateway address. Click **Next**. The SMTP Sender Email address panel as shown in Figure 14-38 appears.

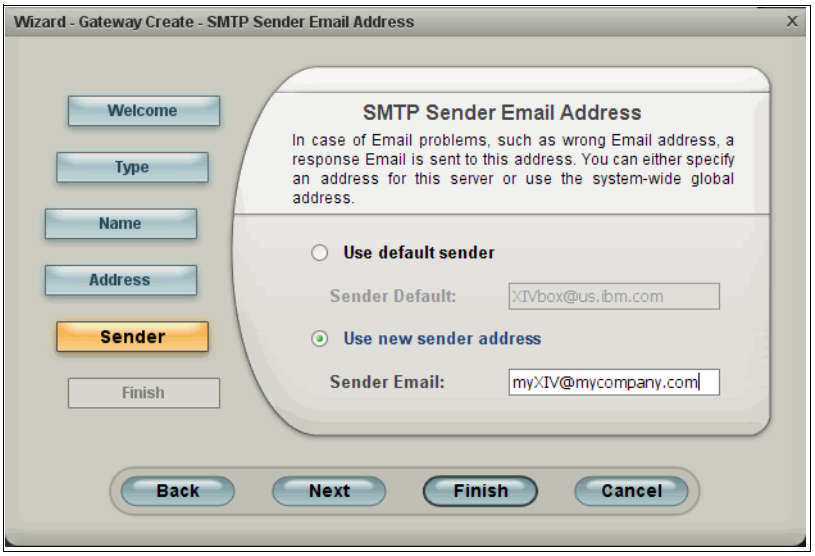


Figure 14-38 SMTP Sender Email Address

This allows you to set the sender email address. You can use the default, or enter a new address. In case of e-mail problems, such as the wrong e-mail address, a response e-mail will be sent to this address. Depending on how your email server is configured, you might need to use an authorized address in order to ensure proper delivery of notifications. Click **Finish**.

The Create Gateway summary panel is displayed as shown in Figure 14-39.

Figure 14-39 Create the Gateway - Summary

This panel allows you to review the information you entered. If all is correct, click **Create**. If not, you can click **Back** until you are at the information that needs to be changed, or just click one of the buttons on the left, to take you directly to the information that needs to be changed.

Destinations

Next, the Events Configuration wizard will guide you through the setup of the destinations where you configure e-mail addresses or SMS receivers. Figure 14-40 shows the Destination panel of the Events Configuration Wizard.

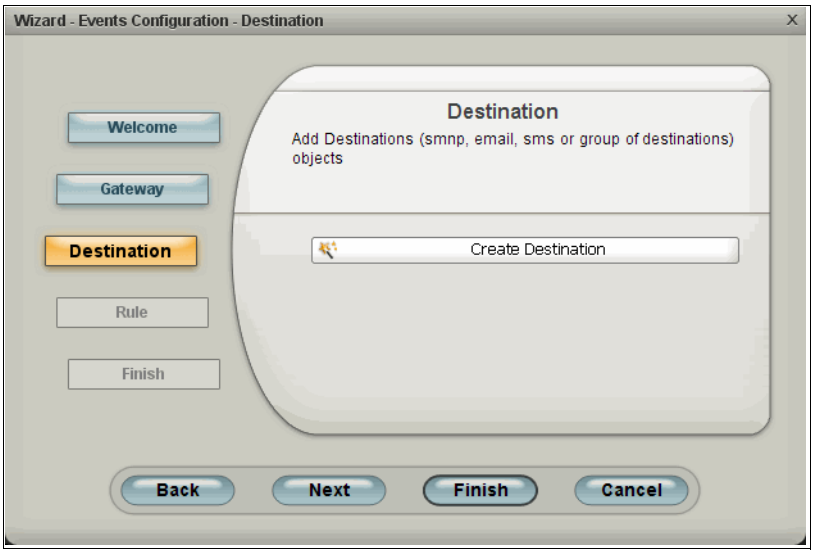


Figure 14-40 Add Destination

Click **Create Destination** to display the Welcome Panel, as shown in Figure 14-41.

Figure 14-41 Destination Create

Click **Next** to proceed. The Select Destination type panel, shown in Figure 14-42, is displayed.

On this panel, you configure:

- Type: Event notification destination type can be either a destination group (containing other destinations), SNMP manager for sending SNMP traps, e-mail address for sending e-mail notification, or mobile phone number for SMS notification:
  - SNMP
  - EMAIL
  - SMS
  - Group of Destinations

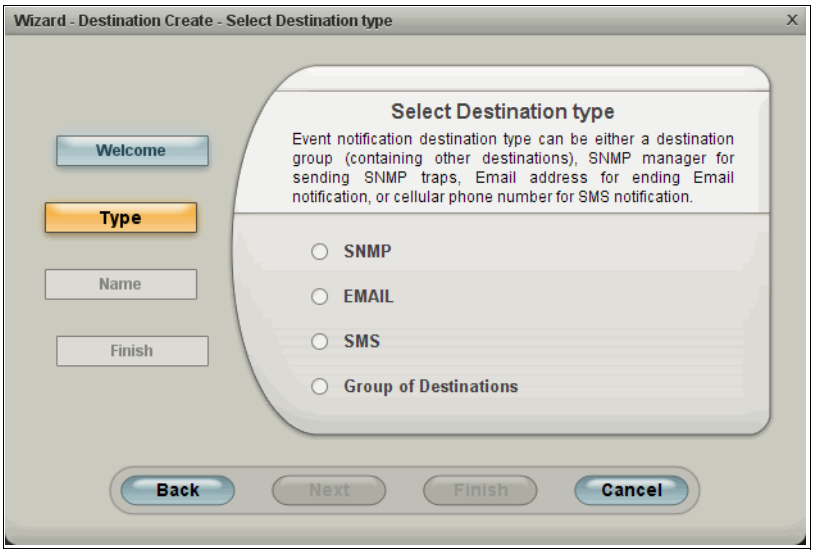


Figure 14-42 Select destination type

Depending on the selected type, the remaining configuration information required differs but is self-explanatory.

Rules

The final step in the Events Creation Wizard is creating a rule. A rule determines what notification is sent. It is based on event severity, event code or both. Click **Create Rule** as shown in Figure 14-43.

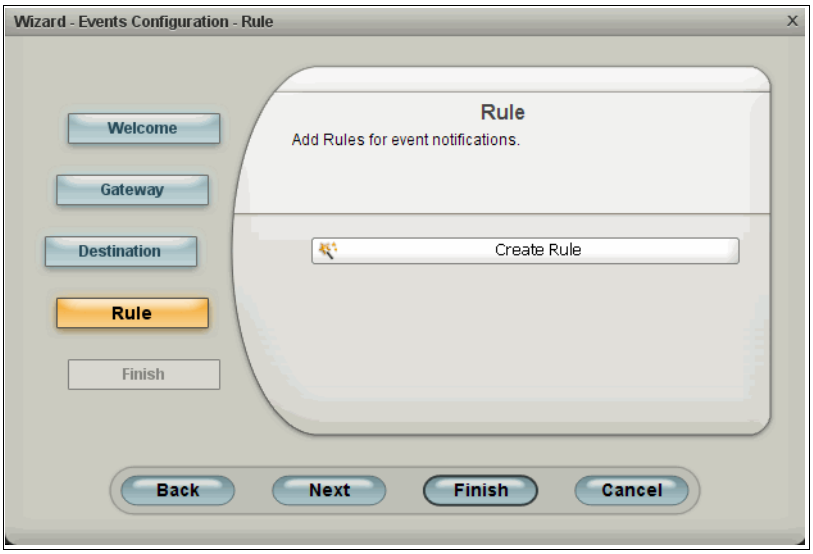


Figure 14-43 Create Rule

The Welcome panel is displayed. Click **Next**. The Rule Create - Rule name panel shown in Figure 14-44 is displayed.

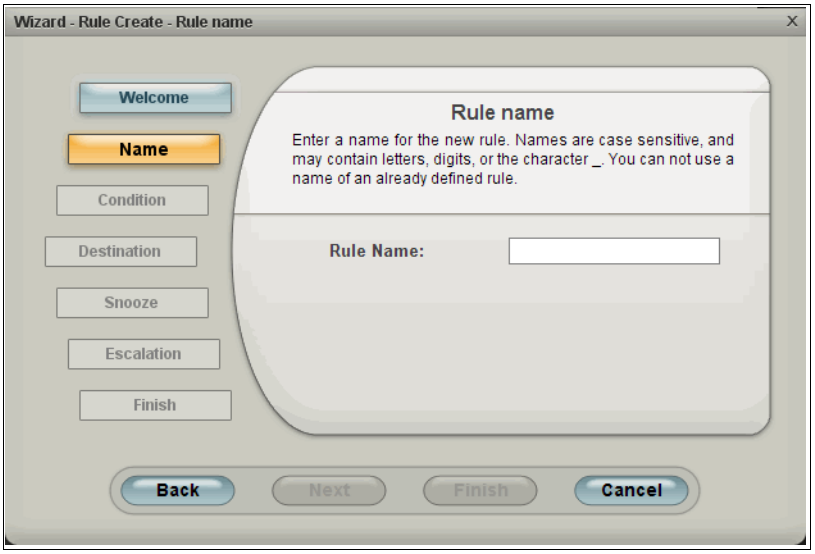


Figure 14-44 Rule name

To define a rule, configure:

- Rule Name: Enter a name for the new rule. Names are case-sensitive and can contain letters, digits, or the underscore character (_). You cannot use the name of an already defined rule.
- Rule condition setting: Select the severity if you want the rule to be triggered by severity, event code if you want the rule be triggered by event, or both severity and event code for events that might have multiple severities depending on a threshold of certain parameters:
  - Severity only
  - Event Code only
  - Both severity and event code
- Select the severity trigger: Select the minimum severity to trigger the rule's activation. Events of this severity or higher will trigger the rules.
- Select the event code trigger: Select the event code to trigger the rule's activation.
- Rule destinations: Select destinations and destination groups to be notified when the event's condition occurs. Here, you can select one or more existing destinations or also define a new destination (refer to Figure 14-45).

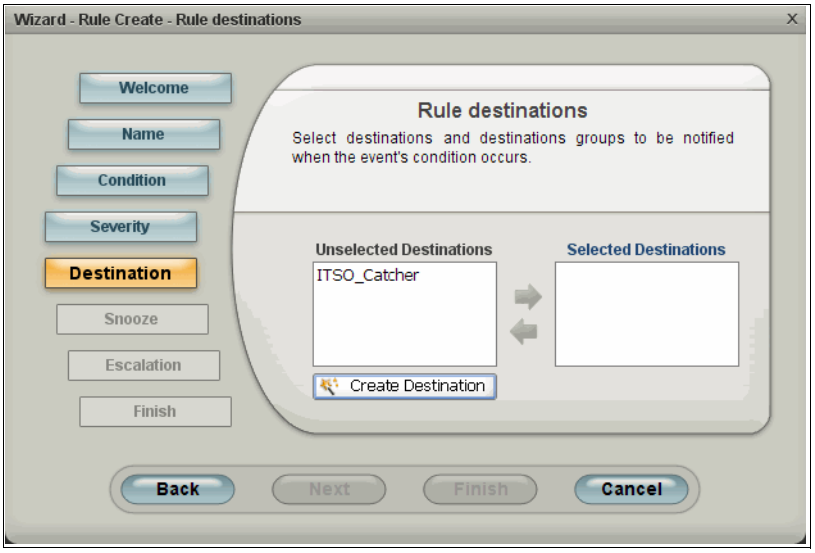


Figure 14-45 Select destination

- Rule snooze: Defines whether the system repeatedly alerts the defined destination until the event is cleared. If so, a snooze time must be selected. Either:
  - Check **Use snooze timer**
  - Snooze time in minutes
- Rule escalation: Allows the system to send alerts via other rules if the event is not cleared within a certain time. If so, an escalation time and rule must be specified:
  - Check **Use escalation rule**
  - Escalation Rule
  - Escalation time in minutes
  - Create Escalation Rule

A summary panel shown in Figure 14-46 allows you to review the information you entered. Go back if you need to make changes, or if all is correct, click **Create**.

Figure 14-46 Rule Create

Setting up notification and rules with the XCLI

You use the same process to set up the XIV Storage System for notification with the XCLI as you used with the GUI. The three-step process includes all the required configurations to allow the XIV Storage System to provide notification of events:

- Gateway
- Destination
- Rules

The gateway definition is used for SMTP and SMS messages. There are several commands used to create and manage the gateways for the XIV Storage System. Example 14-17 shows an SMTP gateway being defined. The gateway is named `test` and the messages from the XIV Storage System are addressed to `xiv@us.ibm.com`.

When added, the existing gateways are listed for confirmation. In addition to gateway address and sender address, the port and reply to address can also be specified. There are several other commands that are available for managing a gateway.

Example 14-17 The smtpgw_define command

```
>> smtpgw_define smtpgw=test address=test.ibm.com from_address=xiv@us.ibm.com
Command executed successfully.

>> smtpgw_list
Name                Address         Priority
ITSO Mail Gateway   us.ibm.com      1
test                test.ibm.com    2
```

The SMS gateway is defined in a similar method. The difference is that the fields can use tokens to create variable text instead of static text. When specifying the address to send the SMS message, tokens can be used instead of hard-coded values. In addition, the message body also uses a token to have the error message sent instead of a hard-coded text.

Example 14-18 provides an example of defining a SMS gateway. The tokens available to be used for the SMS gateway definition are:

- {*areacode*}: This escape sequence is replaced by the destination's mobile or cellular phone number area code.
- {*number*}: This escape sequence is replaced by the destination's cellular local number.
- {*message*}: This escape sequence is replaced by the text to be shown to the user.
- \{, \}, \\: These symbols are replaced by the {, } or \ respectively.

Example 14-18 The smsgw_define command

```
>> smsgw_define smsgw=test email_address={areacode}{number}@smstest.ibm.com
subject_line="XIV System Event Notification" email_body={message}
Command executed successfully.
>> smsgw_list
Name  Email Address                        SMTP Gateways
test  {areacode}{number}@smstest.ibm.com  all
```

When the gateways are defined, the destination settings can be defined. There are three types of destinations:

- SMTP or e-mail
- SMS
- SNMP

Example 14-19 provides an example of creating a destination for all three types of notifications. For the e-mail notification, the destination receives a test message every `Monday` at `12:00`. Each destination can be set to receive notifications on multiple days of the week at multiple times.

Example 14-19 Destination definitions

```
>> dest_define dest=emailtest type=EMAIL email_address=test@ibm.com smtpgws=ALL
heartbeat_test_hour=12:00 heartbeat_test_days=Mon
Command executed successfully.

>> dest_define dest=smstest type=SMS area_code=555 number=5555555 smsgws=ALL
Command executed successfully.

>> dest_define dest=snmptest type=SNMP snmp_manager=9.9.9.9
Command executed successfully.

>> dest_list
Name           Type   Email Address  Area Code Phone Number SNMP Manager            User
ITSO_Catcher   SNMP                                         itsocatcher.us.ibm.com
smstest        SMS                    555       5555555
snmptest       SNMP                                         9.9.9.9
emailtest      EMAIL  test@ibm.com
```

Finally, the rules can be set for which messages can be sent. Example 14-20 provides two examples of setting up rules. The first rule is for SNMP and e-mail messages and all messages, even informational messages, are sent to the processing servers. The second example creates a rule for SMS messages. Only critical messages are sent to the SMS server, and they are sent every 15 minutes until the error condition is cleared.

Example 14-20 Rule definitions

```
>> rule_create rule=emailtest min_severity=informational dests=emailtest,snmptest
Command executed successfully.

>>rule_create rule=smstest min_severity=critical dests=smstest snooze_time=15
Command executed successfully.

>> rule_list
Name         Minimum Severity   Event Codes   Except Codes   Destinations         Active   Escalation Only
ITSO_Major   Major              all                          ITSO_Catcher         yes      no
emailtest    Informational      all                          emailtest,snmptest   yes      no
smstest      Critical           all                          smstest              yes      no
```

Example 14-21 shows illustrations of deleting rules, destinations, and gateways. It is not possible to delete a destination if a rule is using that destination, and it is not possible to delete a gateway if a destination is pointing to that gateway.

Example 14-21 Deletion of notification setup

```
>> rule_delete -y rule=smstest
Command executed successfully.
>> dest_delete -y dest=smstest
Command executed successfully.
>> smsgw_delete -y smsgw=test
Command executed successfully.
```

14.3 Call Home and Remote support

The Call Home function allows the XIV Storage System to send event notifications to the XIV Support Center. This enables both proactive and failure notifications to be sent directly to IBM for analysis. The XIV support center will take appropriate action, up to dispatching an IBM service representative with a replacement part, or engaging level 2 or higher to ensure complete problem determination and solution.

14.3.1 Call Home

Call Home is always configured to use SMTP, and is only configured by qualified IBM service personnel, typically when the XIV is first installed.

14.3.2 Remote support

The XIV Storage System is repaired by trained IBM service personnel, either remotely with the help of the IBM XIV remote support center, or on-site by an IBM SSR. When problems arise, a remote support specialist can connect to the system to analyze the problem, repair it remotely if possible, or assist the IBM SSR who is on-site.

The remote support center has three ways to connect the system. Depending on the client's choice, the support specialist can either connect by

- Using a modem dial-up connection, using a analog phone line provided by the client
- Using a secure, high-speed connection through the Internet, by modifying firewall access for the XIV Storage System
- Using the XIV Remote Support Center (XRSC), which allows the customer to initiate a secure connection from the XIV to IBM. Using XRSC, the XIV system makes a connection to an external XRSC server. Using an internal XRSC server, the XIV Support Center can connect to the XIV, through the connection made to the external server.

  See for details about the XIV Remote Support Center recommended solution.

Note: We highly recommend that the XIV Storage System be connected to the client's public network using XSRC secure high-speed connection.

These possibilities are depicted in Figure 14-47. In case of problems, the remote specialist is able to analyze problems and also assist an IBM SSR dispatched on-site in repairing the system or in replacing field-replaceable units (FRUs).

To enable remote support, you must allow an external connection, such as either:

- A telephone line
- An Internet connection through your firewall that allows IBM to use a VPN connection to your XIV Storage System.

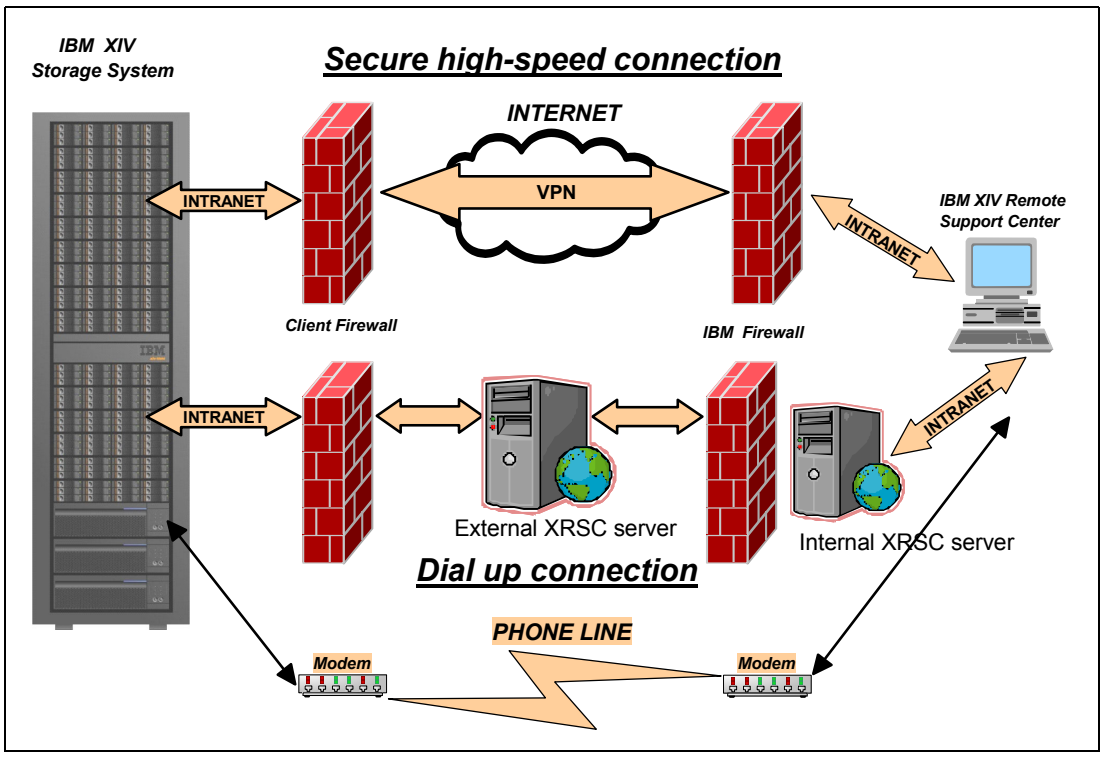


Figure 14-47 Remote Support connections

XIV Remote Support Center Connection

XRSC uses the high speed Internet connection, but gives the client the ability to initiate an outbound SSH call to a secure IBM server.

The XIV Remote Support Center comprises XIV internal functionality together with a set of globally deployed supporting servers to provide secure IBM support access to the XIV system when necessary and when authorized by the customer personnel.

The XIV Remote Support Center was designed with security as a major concern, while keeping the system architecture simple and easy to deploy. It relies on standard, proven technologies and minimizes the logic (code) that must reside either on the External XRSC server or on customer machines.

The XRSC includes extensive auditing features that further enhance security.

Underlying architecture

The XIV Remote Support mechanism has three components (refer toFigure 14-48):

- Remote Support Client (machine internal):

  The Remote Support Client is a software component inside the XIV system that handles remote support connectivity. It relies only on a single outgoing TCP connection, and has no capability to receive inbound connections of any kind. The Client is controlled using XCLI, and is charged with starting a connection, terminating a connection (due to time-out or customer request) and re-trying the connection in case it terminates unexpectedly.

- Remote Support Center Front Server (Internet):

  Front Servers are located on an IBM DMZ of the Internet and receive connections from the Remote Support Client and the IBM XIV Remote Support Back Server. Front Servers are security-hardened machines which provide a minimal set of services, namely, maintaining

connectivity to connected Clients and to the Back Server. They are strictly inbound, and never initiate anything on their own accord. No sensitive information is ever stored on the Front Server, and all data passing through the Front Server from the Client to the Back server is encrypted so that the Front Server or a malicious entity in control of a Front Server cannot access this data.

- Remote Support Center Back Server (IBM Intranet):

  The Back Server manages most of the logic of the system. It is located within IBM's Intranet. The Back Server is access controlled. Only IBM employees authorized to perform remote support of IBM XIV are allowed to use it, and only through specific support interfaces, not with a CLI or a GUI shell. The Back Server is in charge of authenticating a support person, providing the support person with a UI through which to choose a system to support based on the support person's permissions and the list of systems currently connected to the Front Servers and managing the remote support session as it progresses (logging it, allowing additional support persons to join the session, and so on). The Back Server maintains connection to all Front Servers. Support people connect to the Back Server using any SSH client or an HTTPS connection with any browser.

Figure 14-48 provides a representation of the data flow of the XIV to IBM Support.

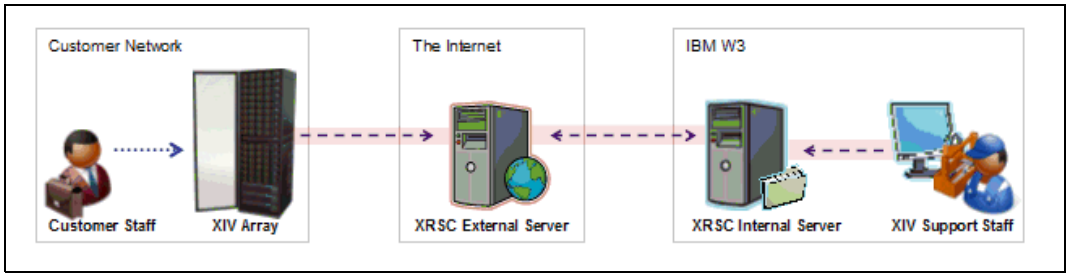


Figure 14-48 XIV Remote Support Center

To initiate the remote connection process, the following steps are performed:

1. Customer initiates an Internet based SSH connection to XRSC either via the GUI or XCLI
2. XRSC identifies the XIV Storage System and marks it as “connected”
3. Support personnel connects to XRSC using SSH over the IBM Intranet
4. XRSC authenticates the support person against the IBM Intranet
5. XRSC then displays the connected customer system available to the support personnel
6. The IBM Support person then chooses which system to support and connect to
   - Only permitted XIV systems are shown
   - IBM Support personnel log their intended activity
7. A fully recorded support session commences
8. When complete, the support person terminates the session and the XRSC disconnects the XIV array from the remote support system.

14.3.3 Repair flow

In case of system problems, IBM XIV support center will be notified by a hardware or software call generated by a notification from the system or by a user’s call.

Based on this call, the remote support center will initiate the necessary steps to repair the problem according to the flow depicted in Figure 14-49.

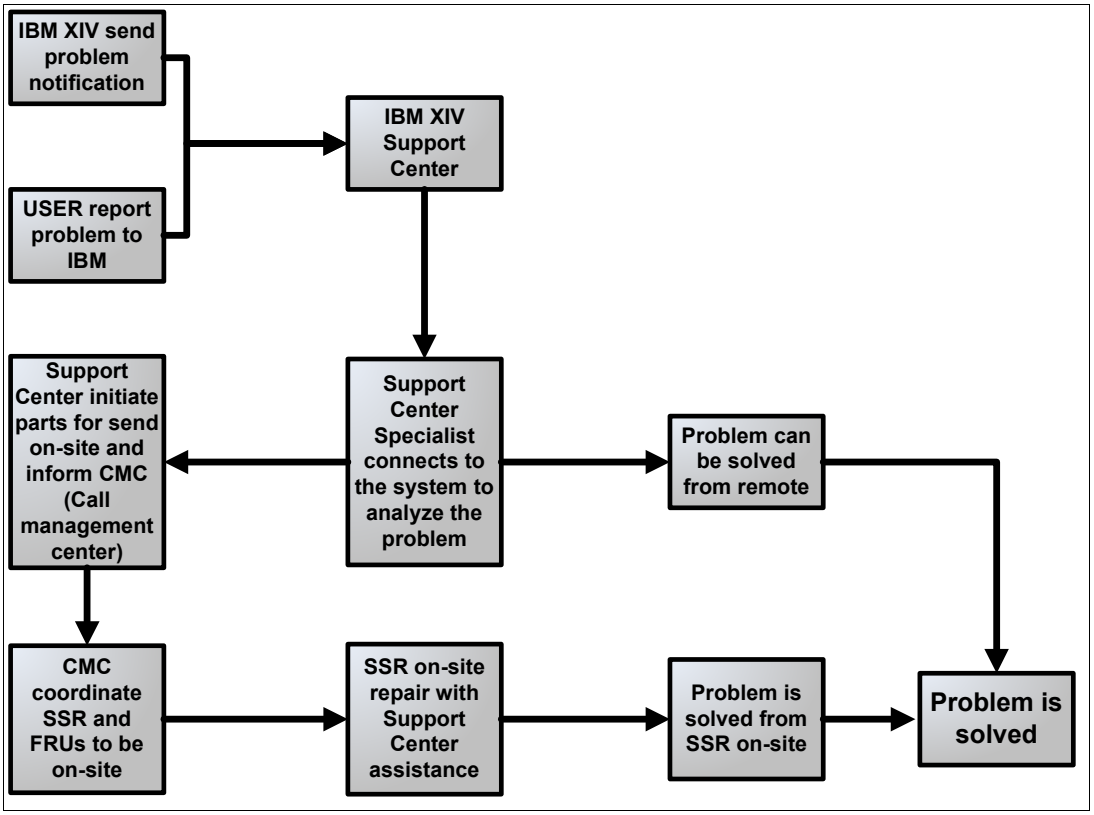


Figure 14-49 Problem Repair Flow

Either a call from the user or an e-mail notification will generate an IBM internal problem record and alert the IBM XIV Support Center. A Support Center Specialist will remotely connect to the system and evaluate the situation to decide what further actions to take to solve the reported issue:

- Remote Repair: Depending on the nature of the problem, a specialist will fix the problem while connected.
- Data Collection: Start to collect data in the system for analysis to develop an action plan to solve the problem.
- On-site Repair: Provide an action plan, including needed parts, to the call management center (CMC) to initiate an IBM SSR repairing the system on-site.
- IBM SSR assistance: Support the IBM SSR during on-site repair via remote connection.

The architecture of the IBM XIV is self-healing. Failing units are logically removed from the system automatically, which greatly reduces the potential impact of the event and results in service actions being performed in a fully redundant state.

For example, if a disk drive fails, it will be automatically removed from the system. The process has a minimal impact on performance, because only a small part of the available resources has been removed. The rebuild time is fast, because most of the remaining drives will participate in redistributing the data.

Due to this self-healing mechanism, with most failures, there is no need for urgent action and service can be performed at a convenient time. The IBM XIV will be in a fully redundant state, which mitigates issues that might otherwise arise if a failure occurs during a service action.

A

Additional LDAP information

This appendix provides additional LDAP related information:

- Creating user accounts in Active Directory
- Creating user accounts in SUN Java Directory
- Securing LDAP communication with SSL
  - Windows Server SSL configuration
  - SUN Java Directory SSL configuration
- Certificate Authority setup

Creating user accounts in Microsoft Active Directory

Creating an account in Microsoft Active Directory for use by XIV LDAP authentication is no different than creating any regular user account. The only exception is the designated "*description*" attribute (field). This field must be populated with the predefined value in order for the authentication process to work.

Start Active Directory Users and Computer by selecting **Start** → **Administrative Tools** → **Active Directory Users and Computers**

Right mouse click on "Users" container, select **New** → **User.** The New Object User dialog window opens as seen in Figure A-1.

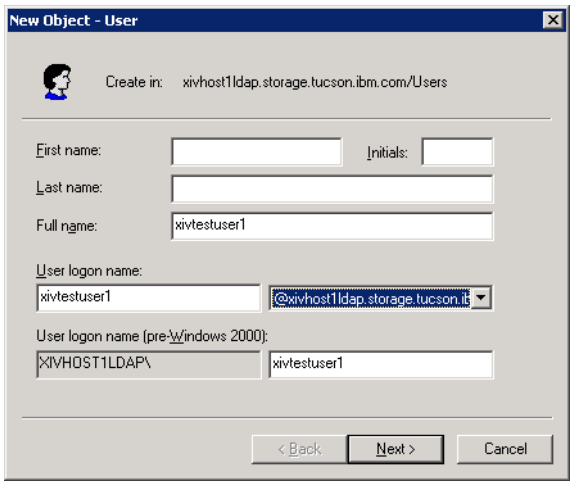


Figure A-1 Creating Active Directory user account

The value entered in "Full name" is what XIV will use as the User name. The only other mandatory field in this form is "User logon name". For simplicity the same `xivtestuser1` value is entered into both fields. Other fields can also be populated but it is not required.

Proceed with creating the account by clicking **Next.** A new dialog window, shown in Figure A-2 is displayed.

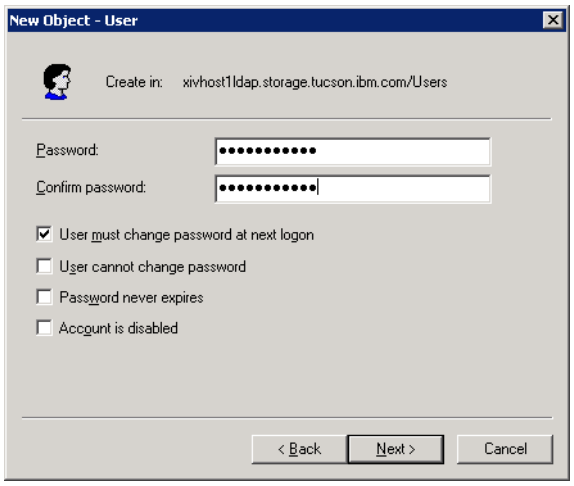


Figure A-2 Assigning password

Note that by default, the password is set to “User must change password at next login”. After the account is created, the user must logon to a server that is part of the Active Directory managed domain to change the password. After the password is changed, all the security rules and policies related to password management are in effect, such as password expiration, maintaining password change history, verifying password complexity, and so on.

Note: If the password initially assigned to an Active directory user is not changed - XIV will not authenticate that user.

Complete the account creation by pressing **Next** → **Finish**.

Proceed with populating the “*description*” field with predefined value for XIV category (role) mapping by selecting the “*xivtestuser1*” user name followed by right mouse click and selecting “Properties”, as illustrated in Figure A-3.

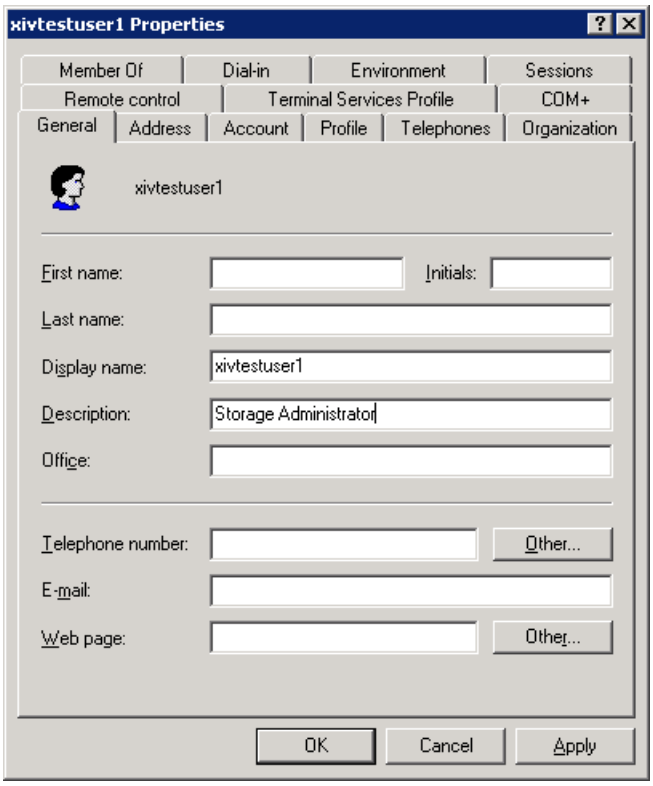


Figure A-3 Entering predefined value into Description field

Complete the account information update by pressing **OK**

After the user account is created in Active Directory, its accessibility can be verified from any of the available LDAP clients. In our case we used the OpenLDAP client, as shown in Example A-1.

Example: A-1 Active Directory account verification using OpenLDAP client

```
$ ldapsearch -x  -H "ldap://xivhost1.xivhost1ldap.storage.tucson.ibm.com:389"  -D
'CN=xivtestuser1,CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com' -w
pass2remember -b 'CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com'
cn=xivtestuser1

dn: CN=xivtestuser1,CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com
```

```
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: user
cn: xivtestuser1
description: Storage Administrator
distinguishedName: CN=xivtestuser1,CN=Users,DC=xivhost1ldap,DC=storage,DC=tucs
 on,DC=ibm,DC=com
instanceType: 4
whenCreated: 20090622172440.0Z
whenChanged: 20090622180134.0Z
displayName: xivtestuser1
uSNCreated: 98467
uSNChanged: 98496
name: xivtestuser1
objectGUID:: apHajqyazEyALYHDAJrjNA==
userAccountControl: 512
badPwdCount: 0
codePage: 0
countryCode: 0
badPasswordTime: 128901682350000000
lastLogoff: 0
lastLogon: 128901682415312500
pwdLastSet: 128901672940468750
primaryGroupID: 513
objectSid:: AQUAAAAAAAUVAAAAn59TxndIlskwvBQmdAQAAA==
accountExpires: 9223372036854775807
logonCount: 3
sAMAccountName: xivtestuser1
sAMAccountType: 805306368
userPrincipalName: xivtestuser1@xivhost1ldap.storage.tucson.ibm.com
objectCategory: CN=Person,CN=Schema,CN=Configuration,DC=xivhost1ldap,DC=storag
 e,DC=tucson,DC=ibm,DC=com
```

The **ldapsearch command** syntax might appear overly complex and its output difficult for interpretation. However this might be the easiest way to verify that the account was created as expected. The **ldapsearch** command can also be very useful for troubleshooting purposes when you are unable to communicate with Active directory LDAP server.

Here is a brief explanation of the **ldapsearch** command line parameters:

`-H 'ldap://xivhost1.xivhost1ldap.storage.tucson.ibm.com:389'` - specifies that the LDAP search query must be sent to "`xivhost1.xivhost1ldap.storage.tucson.ibm.com`" server using port number 389.

`-D 'CN=xivtestuser1,CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com'` - the query is issued on behalf of user "`xivtestuser1`" registered in "`Users`" container in "`xivhost1ldap.storage.tucson.ibm.com`" Active Directory domain.

`-w pass2remember` - is the current password of the user "`xivtestuser1`" (after the initially assigned password was changed to this new password).

`-b 'CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com'` - Base_DN, the location in the directory where to perform the search, the "`Users`" container in the "`xivhost1ldap.storage.tucson.ibm.com`" Active Directory domain.

`cn=xivtestuser1` - specifies what object to search for

The output of the **ldapsearch** command shows the structure of the LDAP object retrieved from the LDAP repository. We do not need to describe every attribute of the retrieved object, however at least two attributes should be checked to validate the response:

```
name: xivtestuser1
description: Storage Administrator
```

The fact that **ldapsearch** returns the expected results in our example indicates that:

1. The account is indeed registered in Active Directory
2. The distinguished name (DN) of the LDAP object is known and valid
3. The password is valid
4. The designated attribute "description" has a predefined value assigned "Storage Administrator"

When the Active Directory account verification is completed, we can proceed with configuring the XIV System for LDAP authentication mode. At this point we still have a few unassigned LDAP related configuration parameters in our XIV System as can be observed in Example A-2.

Example: A-2 Remaining XIV LDAP configuration parameters

```
>> ldap_config_get
Name                      Value
base_dn
xiv_group_attrib          description
third_expiration_event    7
version                   3
user_id_attrib            objectSiD
current_server
use_ssl                   no
session_cache_period
second_expiration_event   14
read_only_role            Read Only
storage_admin_role        Storage Administrator
first_expiration_event    30
bind_time_limit           0
```

base_dn - base DN (distinguished name), the parameter which specifies where in the Active Directory LDAP repository that a user can be located. In our example we use "CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com" as base DN, see Example A-1 on page 357.

current_server - is read-only parameter and can not be populated manually.. It will get updated by the XIV system after the initial contact with LDAP server is established.

session_cache_period - duration in minutes the XIV system keeps user credentials in its cache before discarding the cache contents. If a user repeats the login attempt within session_cache_period minutes from the first attempt, authentication will be done from the cache content without contacting LDAP server for user credentials.

bind_time_limit - the timeout value in seconds after which the next LDAP server on the ldap_list_servers is called. The default value for this parameter is 0. It must be set to a non-zero value in order for bind (establishing LDAP connection) to work. The rule also applies to configurations where the XIV System is configured with only a single server on the ldap_list_servers list.

The populated values are shown in Example A-3.

Example: A-3 Completing and verifying LDAP configuration on XIV

```
>> ldap_config_set
base_dn="CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com"
session_cache_period=10 bind_time_limit=30
Command executed successfully.

$ xcli -c "XIV MN00019" -u ITSO -p redb00k  ldap_config_get
Name                       Value
base_dn CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com
xiv_group_attrib           description
third_expiration_event     7
version                    3
user_id_attrib             objectSiD
current_server
use_ssl                    no
session_cache_period       10
second_expiration_event    14
read_only_role             Read Only
storage_admin_role         Storage Administrator
first_expiration_event     30
bind_time_limit            30
```

To complete our description e of the LDAP related configuration parameters (at the XIV system), we should discuss the parameters that had default values assigned and did not have to be set explicitly. Those are:

`version` - version of LDAP protocol used, default “3”. This parameter should never be changed. Both LDAP products - Active Directory and Sun Java Services Directory Server Enterprise Edition support LDAP protocol version 3

`user_id_attrib` - LDAP attribute set to identify the user (in addition to user name) when recording user operations in the XIV event log, Default `objectSiD` value corresponds to the existing attribute name in Active Directory LDAP object class.

`use_ssl` - indicates if secure (SSL encrypted) LDAP communication is mandated. Default value is **”**`no`**”.** If set to **”**`yes`**”** without configuring both sides for SSL encrypted communication, will result in failing LDAP authentication at the XIV system

`first_expiration_event` - number of days before expiration of certificate to set first alert (severity “warning”). This parameter should be set to a number of days that would give you enough time to generate and deploy new security certificate.

second_expiration_event - number of days before expiration of certificate to set second alert (severity “warning”)

`third_expiration_event` - number of days before expiration of certificate to set third alert (severity “warning”)

Now that all configuration and verification steps are completed, the XIV System is ready for the LDAP mode to be activated.

Creating user accounts in SUN Java Directory

Creating an account in SUN Java Directory can be done in many different ways using various LDAP clients. For illustration purposes we used the LDAP GUI client that is part of Java System Directory Service Control Center - web tool that is part of the SUN Java Directory Server product suit.

The designated *description* attribute must be populated with the predefined value in order for the authentication process to work. From the SUN Java Directory LDAP Server perspective, assigning value to the *description* attribute is not mandatory and will not be enforced by the server itself. LDAP server will allow creating an account with no value assigned to the attribute. However this attribute value is required by the XIV system for establishing LDAP role mapping. This field must be populated with the predefined value in order for the authentication process to work.

To launch Java System Directory Service Control Center point your browser to the ip address of your SUN Java Directory LDAP Server, for a secure connection on port 6789:

In our example we use the following URL for accessing Java System Directory Service Control Center: "`https://xivhost2.storage.tucson.ibm.com:6789`"

Before the first user account can be created, the LDAP administrator must create a suffix. A suffix (also known as a naming context) is a DN that identifies the top entry in the directory hierarchy. SUN Java Directory LDAP server can have multiple suffixes, each identifying a locally held directory hierarchy. For example, o=ibm or in our specific example dc=xivauth

To create a suffix, login to the Java Console using your own userid and password and select "Directory Service Control Center (DSCC)" link in "Services" section. Authenticate to "Directory Service Manager" application. In "Common Tasks" tab select **Directory Entry Management → Create New Suffix or Replication Topology**.

1. Enter Suffix Name. Specify the new suffix DN; In our example `dc=xivauth`, click **Next**.
2. Choose Replication Options. Accept the default "Do Not Replicate Suffix" (LDAP replication is beyond the scope for this book).
3. Choose Server(s). Select "`xivhost2.storage.tucson.ibm.com:389`**"** in Available Servers list and click **Add.** The server name should appear in Chosen Servers list. Click **Next**.
4. Choose Settings. Accept the default "Use Default Settings".
5. Choose Database Location Options. Accept the default database location, click **Next**.
6. Choose Data Options. Select "Create Top Entry for the Suffix. Click **Next**.
7. Review settings for the suffix about to be created and click **Finish** if they are correct.

After the new suffix creation is confirmed, you can proceed with LDAP entry creation.

To create new LDAP entry login to the Java Console using your own and password, select "Directory Service Control Center (DSCC)" link in "Services" section. Authenticate to "Directory Service Manager" application. In "Common Tasks" tab select **Directory Entry Management → Create New Entry**.

The New Entry configuration wizard should now be launched, as follows:

1. The first step of the process is selecting a server instance. SUN Java Directory allows you to create multiple instances of an LDAP server. However the only instance that uses port 389 for non-SSL LDAP and port 636 for SSL LDAP communication can be used for XIV LDAP authentication services. In step one select an instance configured on port 384, as illustrated in Figure A-4.

Figure A-4 Selecting Directory Server Instance

2. The second step (see Figure A-5) is selecting the new entry location. The LDAP administrator determines the location of a new entry. Unlike the Active Directory LDAP repository, where location is directly linked to the domain name, SUN Java Directory LDAP server provides greater flexibility in terms of placement for the new entry.

   The location of all entries for XIV accounts must be the same because the XIV system has only one LDAP configuration parameter that specifies the location. In our example we use `dc=xivauth` as the entry location for XIV user accounts. For simplicity in this example the location name is the same as the suffix name. There are certain similarities between Windows file system and LDAP directory structures. You can think of LDAP suffixes as drive letters. A drive letter can contain directories but you can also put files onto the root directory on a drive letter. In our example we put new account at the level (by analogy) of a root directory, the `dc=xivauth` location.

   As your LDAP repository grows, it might no longer be practical to put types of entries into the same location. In this case, just like with Windows file system, you would create subdirectories and place new entries there and so on. And the LDAP equivalent of what has become a directory hierarchy in your file system is called the Directory Information Tree (DIT). After the entry location is selected and the XIV System is configured to point to that location, all the new account entries can only be created in that location.

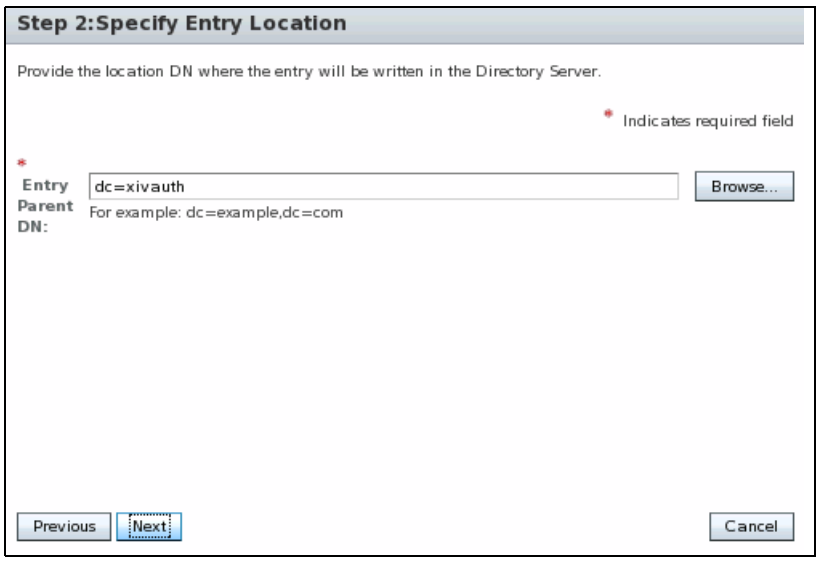


Figure A-5 Selecting entry location

3. Step 3 of the process is to select an object class for the new entry. And again, unlike a predefined object class for a user account in Active Directory LDAP, SUN Java Directory LDAP presents you with a choice of different object class types.

   The LDAP object class describes the content and purpose of the object. It also contains a list of attributes, such as a name, surname or telephone number. Traditionally `inetOrgPerson` object class type is used for LDAP objects describing personal information hence its name - Internet Organizational Person. To be compatible with XIV System an object class must include a minimal set of attributes. These attributes are:

   – uid - user identifier (user name)
   – userPassword - user password
   – description (configurable) - LDAP role mapping attribute

   You can select a different object class type as long as it contains the same minimal set of attributes. Note that object class type can enforce certain rules, for instance some attributes can be designated as mandatory in which case a new LDAP object can not be created without assigning a value to that attribute. In case of `inetOrgPerson` object there are two mandatory attributes - `cn` “Full Name” also called “Common Name” and sn - “Full Name” also called “Surname”. Although it is possible to populate these two objects with different values for simplicity reasons we will be using the `uid` value to populate both `cn` and `sn` attributes. See Figure A-6.

4. Step 4 of the process is entering the attribute values. The first field “Naming Attribute” must remain “uid”; XIV is using that attribute name for account identification. Then, we populate the mandatory attributes with values as described in the previous step. You can also choose to populate other optional attributes and store their values in the LDAP repository; However XIV will not use those attributes. refer to Figure A-7.

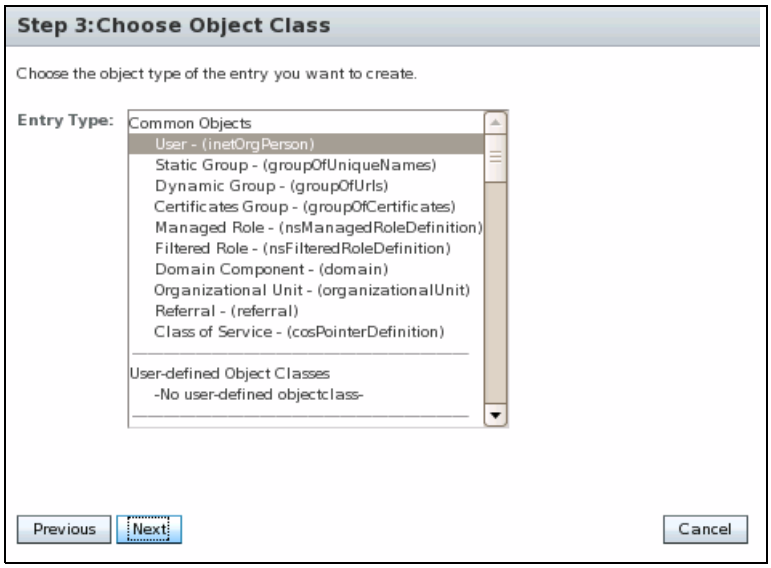


Figure A-6 Object class selection

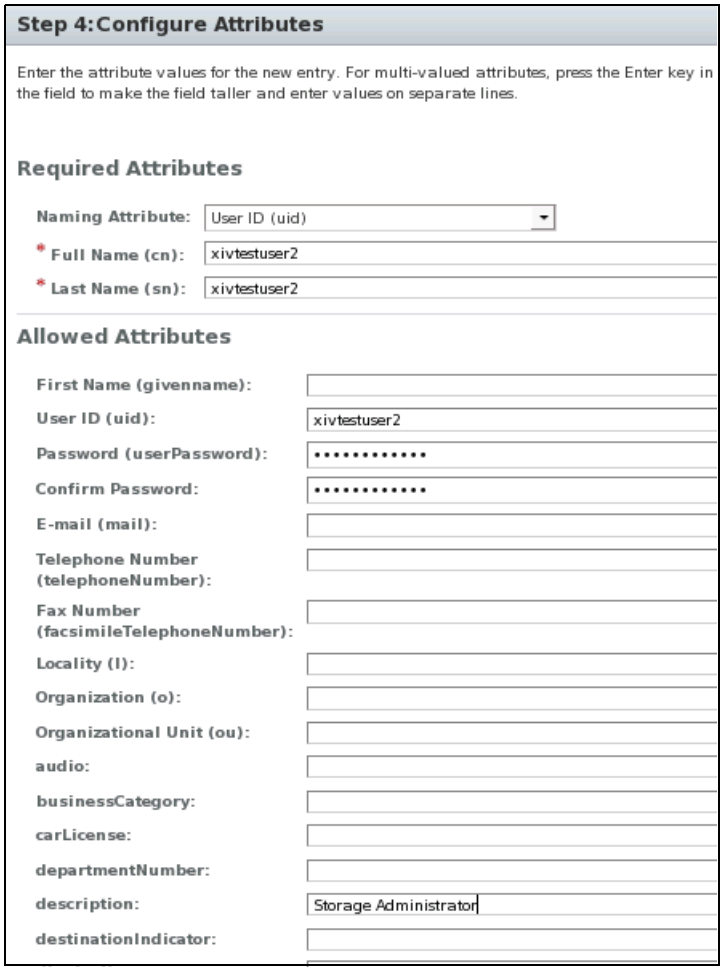


Figure A-7 Entering object attribute values

5. Step 5 is the last step of the process. A Summary panel (Figure A-8) shows what you have selected and entered in the previous steps. You can go back and change parameters if you choose to do so. Otherwise you proceed with the entry creation.

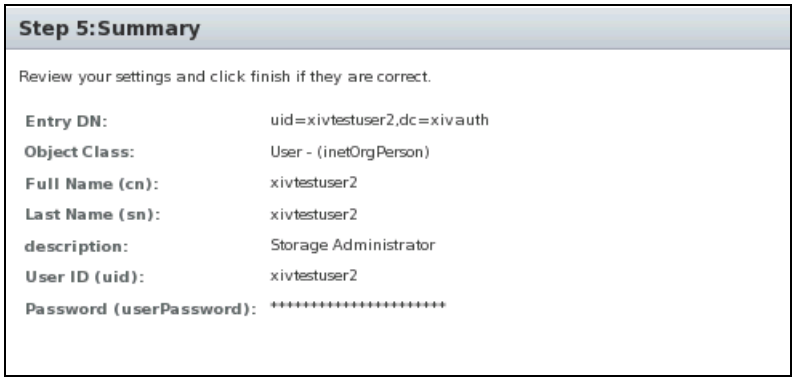


Figure A-8 Reviewing entry settings

If all the information was entered correctly you should get an “Operation completed Successfully” message on the next panel, shown in Figure A-9. If the operation failed for one reason or another you need to go back and make necessary changes before resubmitting your request.

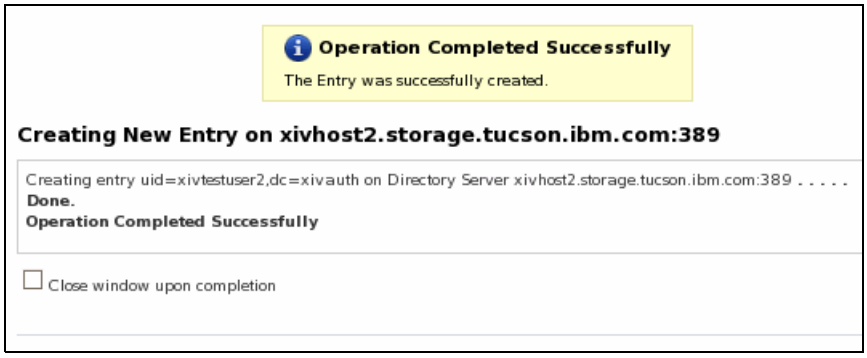


Figure A-9 Entry creation confirmation

After the user account was created in SUN Java Directory LDAP its accessibility can be verified using any of the available LDAP clients. In our example (Example A-4), we use the SUN Java Directory LDAP client.

Example: A-4 SUN Java Directory account verification using SUN Java Directory LDAP client

```
$ /opt/sun/dsee6/bin/ldapsearch -b dc=xivauth -h xivhost2.storage.tucson.ibm.com
-D uid=xivtestuser2,dc=xivauth -w pwd2remember uid=xivtestuser2

dn: uid=xivtestuser2,dc=xivauth
uid: xivtestuser2
description: Storage Administrator
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
sn: xivtestuser2
cn: xivtestuser2
```

The **ldapsearch command** syntax might appear overly complex and its output difficult for interpretation. However, this might be the easiest way to verify that the account was created as expected. The **ldapsearch** command can also be very useful for troubleshooting purposes when you are unable to communicate with Active directory LDAP server.

Here is a brief explanation of the **ldapsearch** command line parameters:

`-h xivhost2.storage.tucson.ibm.com` - specifies that the LDAP search quesry must be sent to "`xivhost2.xivhost1ldap.storage.tucson.ibm.com`" server using default port 389

`-b dc=xivauth` - Base_DN, the location in the Directory Information Tree (DIT).

`-D uid=xivtestuser2,dc=xivauth` - the quesry is issued on behalf of user `xivtestuser2` located in `dc=xivauth` SUN Directory repository.

`-w pwd2remember` - is the current password of the user `xivtestuser2`

`uid=xivtestuser2` - specifies what object to search

The output of the "`ldapsearch`" command shows the structure of the object found. We do not need to describe every attribute of the returned object however at least two attributes should be checked to validate the response:

```
uid: xivtestuser2
description: Storage Administrator
```

The fact that **ldapsearch** returns the expected results in our example indicates that:

1. The account is registered in SUN Java Directory.
2. We know where in the SUN Java Directory repository the account is located.
3. We know the valid password.
4. Designated attribute "`description`" has predefined value assigned "`Storage Administrator`".

When the SUN Java Directory account verification is completed we can proceed with configuring XIV System for LDAP authentication mode. At this point we still have a few unassigned LDAP related configuration parameters in our XIV System as can be observed in Example A-5.

Example: A-5 Remaining XIV LDAP configuration parameters

```
>> ldap_config_get
Name                     Value
base_dn
xiv_group_attrib         description
third_expiration_event   7
version                  3
user_id_attrib           objectSiD
current_server
use_ssl                  no
session_cache_period
second_expiration_event  14
read_only_role           Read Only
storage_admin_role       Storage Administrator
first_expiration_event   30
bind_time_limit          0
```

base_dn - base DN (distinguished name) ,the parameter which specifies where in SUN Java Directory DIT a user can be located. In our example we use “dc=xivauth” as base DN

user_id_attrib - LDAP attribute set to identify the user (in addition to user name) when recording user operations in the XIV event log. The default value for the attribute is objectSiD which is suitable for Active Directory but not for SUN Java Directory LDAP. objectSiD attribute is not defined in inetOrgPerson object class used by SUN Java Directory. In our example we set it to uid.

current_server - is read-only parameter and can not be populated manually.. It will get updated by XIV System after the initial contact with LDAP server is established.

session_cache_period - duration in minutes XIV System keeps user credentials in its cache before discarding the cache contents. If a user repeats login attempt within session_cache_period minutes from the first attempt - authentication will be done based on the cache content without contacting LDAP server for user credentials.

bind_time_limit - the timeout value in seconds after which the next LDAP server on the ldap_list_servers is called. Default value for this parameter is 0. It must be set to a non-zero value in order for bind (establishing LDAP connection) to work. The rule also applies to configurations where XIV System is configured with only a single server on the ldap_list_servers list.

The populated values are shown in Example A-6.

Example: A-6 Completing and verifying LDAP configuration on XIV

```
$ xcli -c "ARCXIVJEMT1" -u admin -p s8cur8pwd ldap_config_set base_dn="dc=xivauth"
user_id_attrib=uid session_cache_period=10 bind_time_limit=30
Command executed successfully.

$ xcli -c "XIV MN00019" -u admin -p s8cur8pwd ldap_config_get
Name                      Value
base_dn                    dc=xivauth
xiv_group_attrib           description
third_expiration_event     7
version                    3
user_id_attrib             sid
current_server
use_ssl                    no
session_cache_period       10
second_expiration_event    14
read_only_role             Read Only
storage_admin_role         Storage Administrator
first_expiration_event     30
bind_time_limit            30
```

To complete our description e of the LDAP related configuration parameters (at the XIV system), we should discuss the parameters that had default values assigned and did not have to be set explicitly. Those are:

`version` - version of LDAP protocol used, default “3”. This parameter should never be changed. Both LDAP products - Active Directory and Sun Java Services Directory Server Enterprise Edition support LDAP protocol version 3

`user_id_attrib` - LDAP attribute set to identify the user (in addition to user name) when recording user operations in the XIV event log

`use_ssl` - indicates if secure (SSL encrypted) LDAP communication is mandated. If set to “yes” without configuring both sides for SSL encrypted communication will result in failing LDAP authentication on XIV System

`first_expiration_event` - number of days before expiration of certificate to set first alert (severity “warning”). This parameter should be set to a number of days so it would give you enough time to generate and deploy new security certificate.

`second_expiration_event` - number of days before expiration of certificate to set second alert (severity “warning”)

`third_expiration_event` - number of days before expiration of certificate to set third alert (severity “warning”)

Securing LDAP communication with SSL

In any authentication scenario, information is exchanged between the LDAP server and XIV system where access is being sought. Security Socket Layer (SSL) can used to implement secure communications between the LDAP client and server. LDAPS (LDAP over SSL, the secure version of LDAP protocol) allows secure communications between the XIV system and LDAP server with encrypted SSL connections. This allows a setup where user passwords never appear on the wire in clear text.

SSL provides methods for establishing identity using X.509 certificates and ensuring message privacy and integrity using encryption. In order to create an SSL connection the LDAP server must have a digital certificate signed by a trusted certificate authority (CA). Companies have the choice of using a trusted third-party CA or creating their own certificate authority. In this scenario, the `xivauth.org` CA will be used for demonstration purposes.

To be operational SSL, has to be configured on both the client and the server. Server configuration includes generating a certificate request, obtaining a server certificate from a certificate authority (CA), and installing the server and CA certificates.

Windows Server SSL configuration

To configure SSL for LDAP on a Windows Server, you must install the MMC snap-in to manage local certificates, create a certificate request (CER), have CER signed by a CA, import the signed certificate into the local keystore, import a CA certificate as a trusted root CA, and then reboot the server for the new configuration to take effect.

Installation of the local certificates MMC snap-in

Install the certificate snap-in for MMC to allow you to manage the certificates in your local machine keystore. The procedure to install the MMC snap-in is as follows:

1. Start the Management Console (MMC) by selecting **Start**, then **Run**. Type mmc /a and select **OK**.
2. Select the **File** → **Add/Remove Snap-in** menu to open the Add/Remove Snap-in dialog.
3. Select the **Add** button to open the Add Standalone Snap-In dialog. Select the **Certificates** snap-in and then click **Add**.
4. Select the **Computer Account** option to manage system-wide certificates. **Click Next** to continue.
5. Select the **Local Computer** option to manage certificates on the local computer only. Select the **Finish**, **Close**, and then click **OK** to complete the snap-in installation.
6. Select **File** → **Save as** and save the console configuration in the %SYSTEMROOT%\system32 directory with a file name of `localcert.msc`.
7. Create a shortcut in the Administrative Tools folder in your Start menu by right-clicking the **Start** menu and then **Open All Users**. Select the **Program** folder and then the **Administrative Tools** folder.
8. Select the **File** → **New** → **Shortcut** menu. Then enter the location of the saved console, %SYSTEMROOT%\system32\localcert.msc, in the Type the location of the item field. Click **Next** to continue.
9. Enter the name of the new shortcut, Certificates (Local Computer), in the “Type a name for this shortcut” field.
10. To start the local certificate management tool, select **Start** → **Administrative tools** → **Certificates** (**Local Computer)**.

When the local certificate management tool starts, it will appear as shown in Figure A-10. The certificates used by Active Directory are located in the **Console Root** → **Certificates (Local Computer)** → **Personal** → **Certificates** folder. The list of trusted root certificates authorities is located in the **Console Root** → **Certificates (Local Computer)** → **Trusted Certification Authorities** → **Certificates** folder. See Figure A-10.

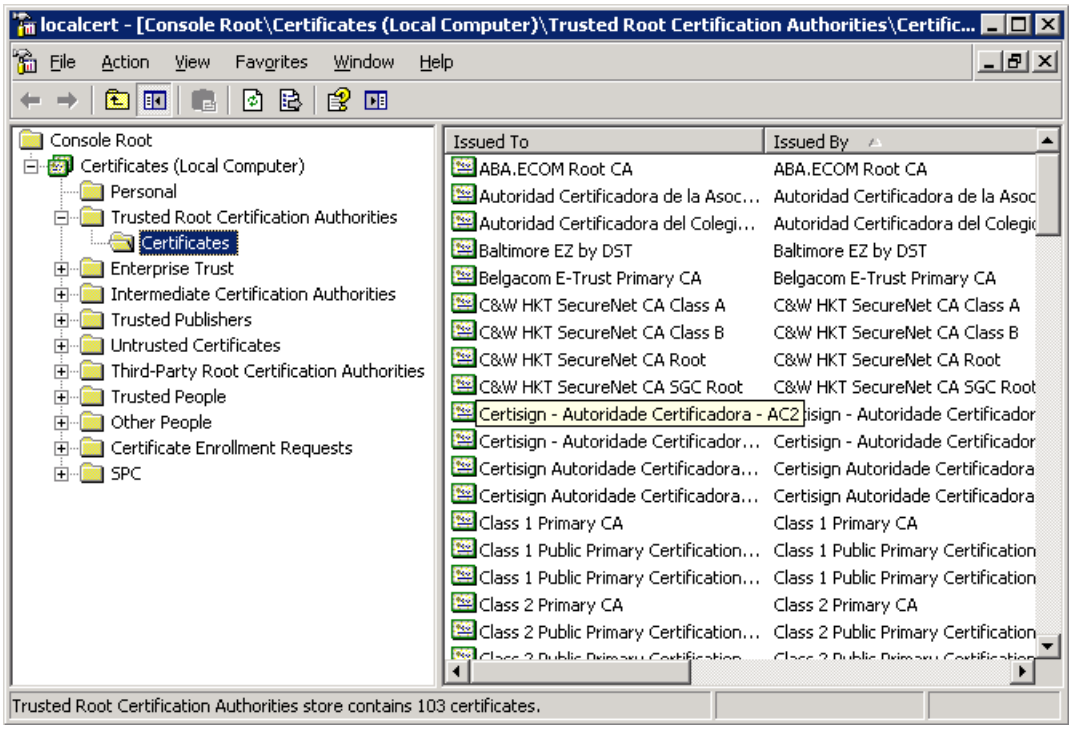


Figure A-10 Windows certificate MMC snap-in (local computer)

Generating a Windows Server certificate request

You must use the **certreq** command to generate a certificate request. The **certreq** command uses a text instruction file, which specifies the attributes needed to generate a certificate. It contains attributes such as the subject's common name, certificate key length, and additional key usage extensions. The Active Directory requires that the certificate meet the following requirements:

- The private key and certificate for the local machine must be imported into the local computer's personal keystore.
- The fully qualified domain name (FQDN) for the Active Directory must appear
- in the common name (CN) in the subject field or DNS entry in the subject alternative name extension.
- The certificate must be issued by a CA that the Active Directory server and the XIV system trust.
- The certificate must contain the enhanced key usage extension that specifies the server authentication object identifier (OID) 1.3.6.1.5.5.7.3.1. This OID indicates that the certificate will be used as a SSL server certificate.

Example A-7 shows the text instruction file used to generate the certificate for the `xivhost1ldap.storage.tucson.ibm.com` domain controller. The subject field is set to CN=`xivhost1.xivhost1ldap.storage.tucson.ibm.com`, which is the FQDN of the domain controller. You then use the **certreq** command to generate the certificate request file.

Example: A-7 Text instruction file for certificate request generation

```
[Version]

Signature="$Windows NT$

[NewRequest]

Subject = "CN=xivhost1.xivhost1ldap.storage.tucson.ibm.com"
KeySpec = 1
KeyLength = 1024
; Can be 1024, 2048, 4096, 8192, or 16384.
; Larger key sizes are more secure, but have
; a greater impact on performance.
Exportable = TRUE
MachineKeySet = TRUE
SMIME = False
PrivateKeyArchive = FALSE
UserProtected = FALSE
UseExistingKeySet = FALSE
ProviderName = "Microsoft RSA SChannel Cryptographic Provider"
ProviderType = 12
RequestType = PKCS10
KeyUsage = 0xa0

[EnhancedKeyUsageExtension]

OID=1.3.6.1.5.5.7.3.1 ; this is for Server Authentication

C:\SSL\> certreq -new xivhost1_cert_req.inf xivhost1_cert_req.pem
C:\SSL\>
```

Signing and importing Windows server certificate

After the CER is generated (`xivhost1_cert_req.pem`), you must send the request to the certificate authority to be signed. For more information about signing this certificate see “Signing a certificate for xivhost1 server” on page 384. After the signed certificate is returned, you must import the certificate into the local machines’s personal keystore.

Example A-8 shows how to import the signed certificate using the **certreq** command. Confirm that the certificate is imported correctly by using the **certutil** command.

Example: A-8 Accepting the signed certificate into local certificate keystore

```
C:\>certreq -accept xivhost1_cert.pem
C:\SSL>certutil -store my
================ Certificate 0 ================
Serial Number: 01
Issuer: E=ca@xivstorage.org, CN=xivstorage, O=xivstorage, L=Tucson, S=Arizona,
C=US
Subject: CN=xivhost1.xivhost1ldap.storage.tucson.ibm.com
Non-root Certificate
Cert Hash(sha1): e2 8a dd cc 84 47 bc 49 85 e2 31 cc e3 23 32 c0 ec d2 65 3a
  Key Container =
227151f702e7d7b2105f4d2ce0f6f38e_8aa08b0a-e9a6-4a73-9dce-c84e45aec165
  Provider = Microsoft RSA SChannel Cryptographic Provider
Encryption test passed
```

```
CertUtil: -store command completed successfully.
```

Importing a Certificate Authority certificate

Until the `xivstorage.org` CA is designated as a trusted root, any certificate signed by that CA will be untrusted. You must import the CA's certificate, using the local certificate management tool, into the Trusted Certification Authorities folder in the local keystore.

To start the local certificate management tool, select **Start** → **Administrative tools** → **Certificates** (**Local Computer)**.

1. After the certificate tool opens, select the **/Console Root/Certificates (Local Computer)/Trusted Certification Authorities** folder.
2. Start the certificate import wizard by selecting the **Action** → **All Tasks** → **Import** menu. Click **Next** to continue.
3. Select the file you want to import. The `xivstorage.org` CA certificate is located in the cacert.pem file. Click **Next** to continue.
4. Select the **Place all certificates in the following store** option and make sure the certificate store field is set to **Trusted Root Certification Authorities**.Click **Next** to continue.
5. The CA certificate is now imported. Click **Finish** to close the wizard.

After the CA and server certificates are imported into the local keystore, you can then use the local certificate management tool to check whether the certificates are correctly imported. Open the **Console Root** → **Certificates (Local Computer)** → **Personal** → **Certificates** folder and select the certificate issued to xivhost1.xivhost1ldap.storage.tucson.ibm.com.

Figure A-11 shows that the certificate which was issued to xivhost1.xivhost1ldap.storage.tucson.ibm.com is valid and was issued by the xivstorage CA. The certificate has a corresponding private key in the keystore. The "Ensures the identity of the remote computer" text indicates that the certificate has the required server authentication key usage defined.

To check the xivstorage certificate, open the **Console Root** → **Certificates (Local Computer)** → **Trusted Certification Authorities** → **Certificates** folder and select the certificate issued by xivstorage. Figure A-12 shows that the certificate issued to and by the xivstorage CA is valid.

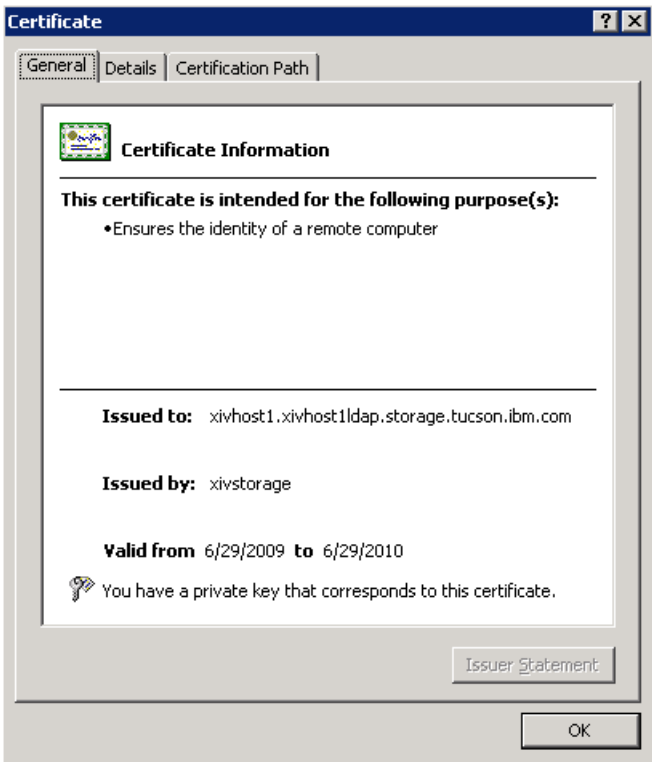


Figure A-11 Certificate information dialog

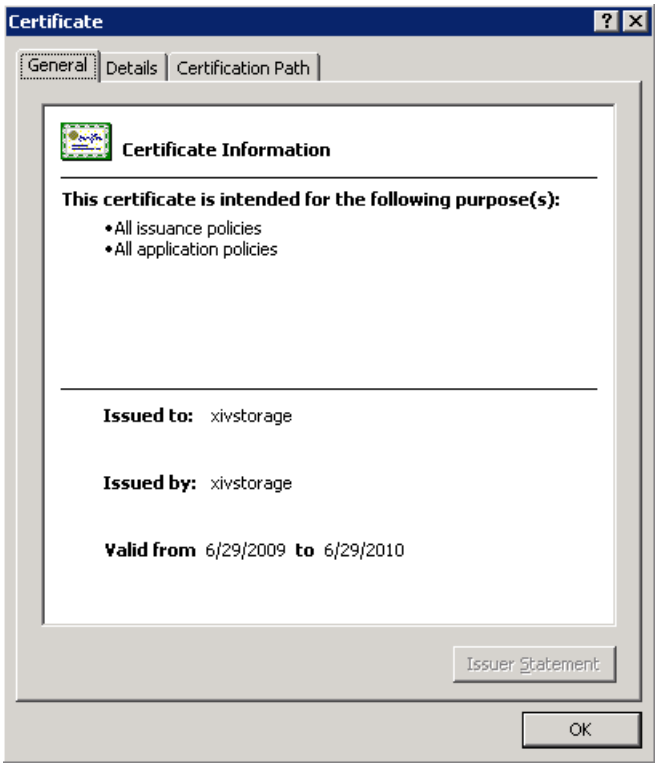


Figure A-12 Certificate information dialog for xivstorage certificate authority

Low-level SSL validation using the openssl command

The easiest way to test the low-level SSL connection to the LDAP server is by using the openssl s_client command with the -showcerts option. This command will connect to the specified host and list the server certificate, the certificate authority chain, supported ciphers, SSL session information, and verify return code. If the SSL connection worked, the openssl s_client command result in the verify return code will be 0 (Ok).

Example A-9 shows the output of the **openssl s_client** command connecting Linux server (xivstorage.org) to the Active Directory server xivhost1.xivhost1ldap.storage.tucson.ibm.com. This command connects to the Active Directory server using the secure LDAP port (636).

Example: A-9 Low-level SSL validation using the openssl s_client

```
openssl s_client  -host xivhost1.xivhost1ldap.storage.tucson.ibm.com -port 636
-CAfile cacert.pem  -showcerts
   ...
   ---
Server certificate
subject=/CN=xivhost1.xivhost1ldap.storage.tucson.ibm.com
issuer=/C=US/ST=Arizona/L=Tucson/O=xivstorage/CN=xivstorage/emailAddress=ca@xivsto
rage.org
---
   ...
New, TLSv1/SSLv3, Cipher is RC4-MD5
Server public key is 1024 bit
SSL-Session:
    Protocol  : TLSv1
    Cipher    : RC4-MD5
    Session-ID: 9E240000CE9499A4641F421F523ACC347ADB91B3F6D3ADD5F91E271B933B3F4F
    Session-ID-ctx:
    Master-Key:
F05884E22B42FC4957682772E8FB1CA7772B8E4212104C28FA234F10135D88AE496187447313149F2E
89220E6F4DADF3
    Key-Arg   : None
    Krb5 Principal: None
    Start Time: 1246314540
    Timeout   : 300 (sec)
    Verify return code: 0 (ok)
```

Note: In order to complete the configuration of SSL for the Active Directory, you must reboot the Windows server.

Basic secure LDAP validation using the ldapsearch command

After you have confirmed that the SSL connection is working properly, you should confirm that you are able to search your LDAP directory using LDAP on a secure port. This will confirm that the LDAP server can communicate using an SSL connection.

In our example we use OpenLDAP client for SSL connection validation. CA certificate needs to be added to the key ring file used by OpenLDAP client. TLS_CERTS option in OpenLDAP configuration file (typically /etc/openldap/ldap.conf) specifies the file that contains certificates for all of the Certificate Authorities the client will recognize. See Example A-10.

Example: A-10 Testing LDAP over SSL using ldapsearch command

```
# /usr/bin/ldapsearch -x  -H
“ldaps://xivhost1.xivhost1ldap.storage.tucson.ibm.com:636” -D
'CN=xivtestuser1,CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com' -w
pass2remember -b 'CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com'

dn: CN=xivtestuser1,CN=Users,DC=xivhost1ldap,DC=storage,DC=tucson,DC=ibm,DC=com
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: user
cn: xivtestuser1
description: Storage Administrator
distinguishedName: CN=xivtestuser1,CN=Users,DC=xivhost1ldap,DC=storage,DC=tucs
 on,DC=ibm,DC=com
...
# search result
search: 2
result: 0 Success
```

The URI format used with “-H” option specifies that LDAPS (LDAP over SSL) must be used on port 636 (LDAP secure port).

SUN Java Directory SSL configuration

This section illustrates the use of an SSL protocol for communication with the SUN Java Directory.

Creating SUN Java Directory certificate request

To configure SSL for SUN Java LDAP Directory, you must create a certificate request (CER), have CER signed by a CA, import the signed certificate into the local keystore, import a CA certificate as a trusted root CA, and then restart the LDAP server for the new configuration to take effect.

Generating a SUN Java Directory server certificate request

To generate a certificate request using SUN Java Web Console tool:

- Point your web browser HTTPS port 6789, in our example “https://xivhost2.storage.tucson.ibm.com:6789”.
- Login to the system and select “Directory Service Control Center (DSCC)” application and Authenticate to Directory Service Manager.
- Select **Directory Servers** → **xivhost2.storage.tucson.ibm.com:389** → **Security** → **Certificates** → **Request CA-Signed Certificate.** Fill out the certificate request form. Sample of certificate request form is shown in Figure A-13.

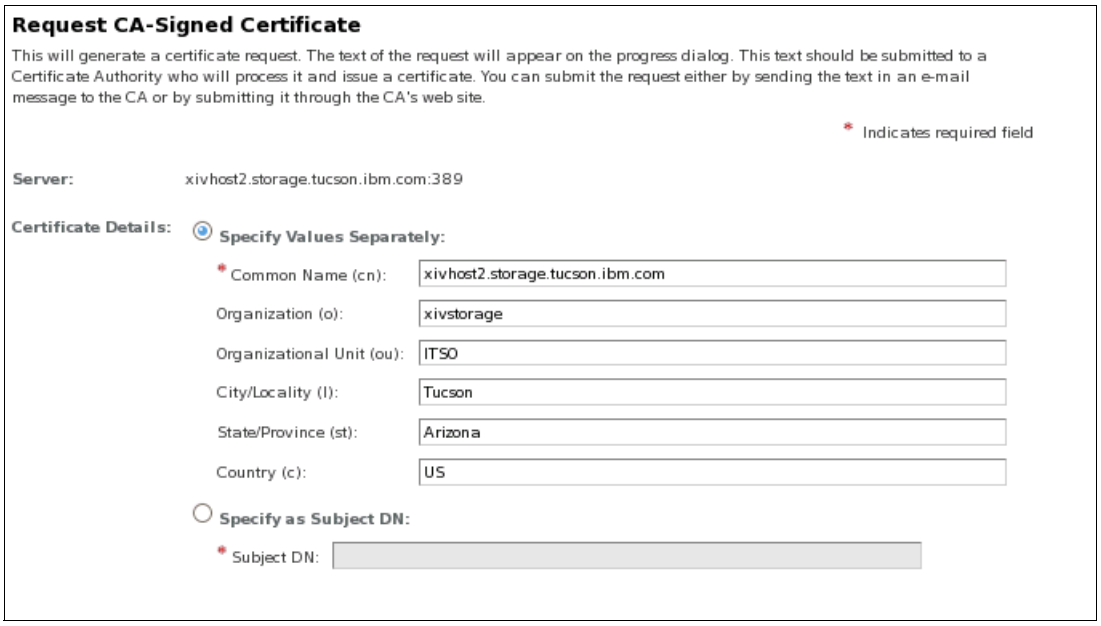


Figure A-13 Certificate request

Copy the generated certificate shown in Figure A-14 request into `xivhost2_cert_req.pem` file.

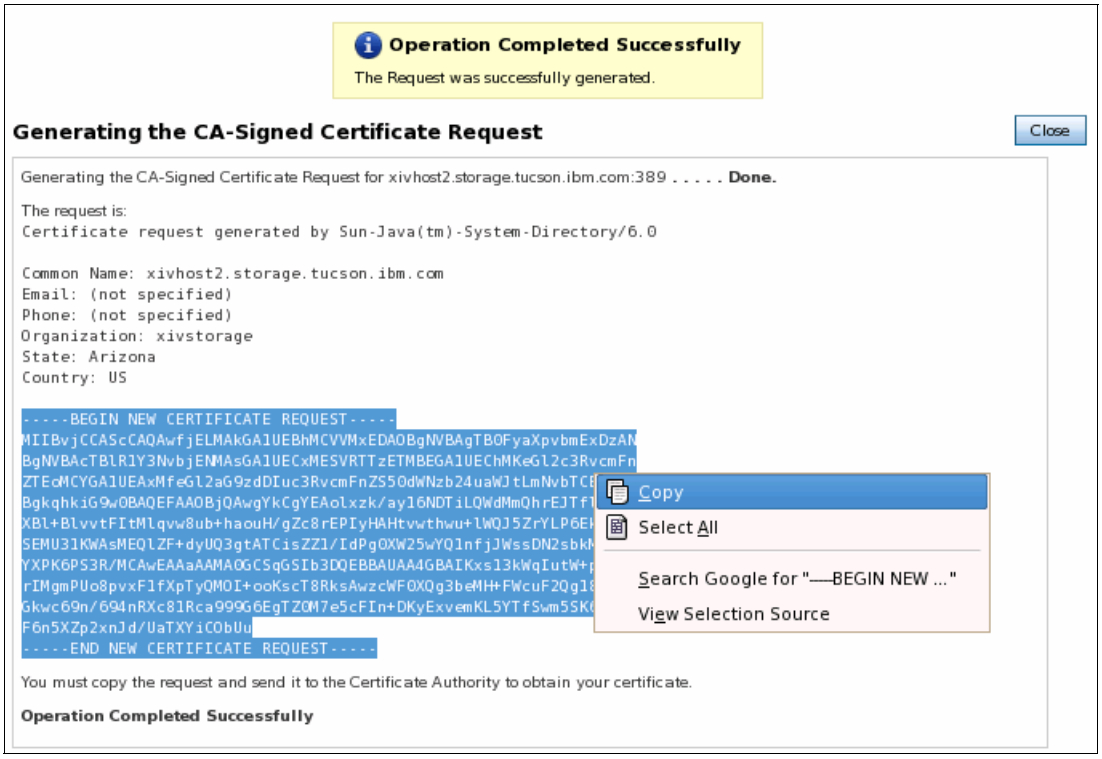


Figure A-14 Generated certificate

Signing and importing a server certificate

After the CER is generated (`xivhost2_cert_req.pem`), you must send the request to the certificate authority to be signed. For more information about signing this certificate, see “” on page 384. After the signed certificate `xivhost2_cert.pem` file is returned, you must import the certificate into the local machine’s personal keystore.

To add signed certificate in “Directory Service Manager” manager application, select **Directory Servers** → **xivhost2.storage.tucson.ibm.com:389** → **Security** → **Certificates** → **Add**.

Copy and paste certificate stored in `xivhost2_cert.pem` file as shown in Figure A-15.

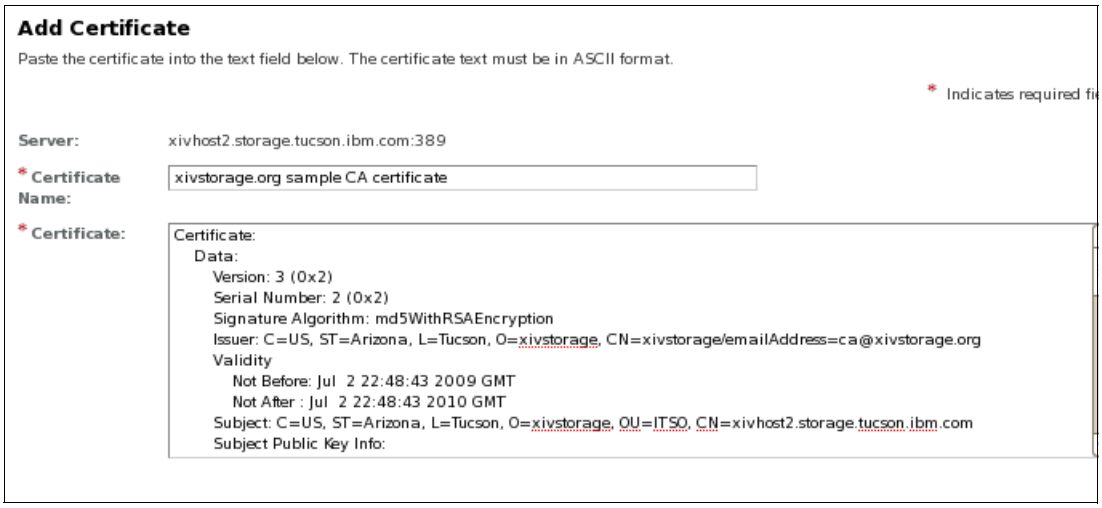


Figure A-15 Adding signed certificate

Importing a Certificate Authority certificate

Until the `xivstorage.org` CA is designated as a trusted root, any certificate signed by that CA will be untrusted. You must import the CA’s certificate using “Directory Service Manager” manager application by selecting **Directory Servers** → **xivhost2.storage.tucson.ibm.com:389** → **Security** → CA Certificates → **Add.**

Copy and paste Certificate Authority certificate stored in `cacert.pem` file as shown in Figure A-16.

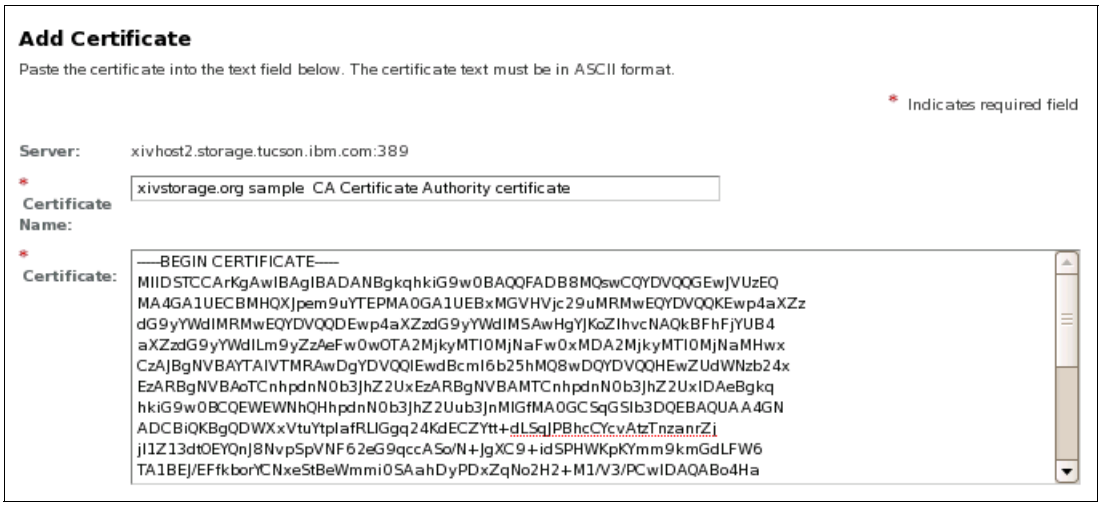


Figure A-16 Importing Certificate Authority certificate

After the CA and signed certificates are imported into the local keystore, you can use the local certificate management tool to check whether the certificates are correctly imported.

Open the **Directory Servers** → **xivhost2.storage.tucson.ibm.com:389** → **Security** → **Certificates** and click the link `xivstorage.org sample CA certificate`.

Figure A-17 shows that the certificate issued to xivhost2.storage.tucson.ibm.com is valid and was issued by the xivstorage Certificate Authority.

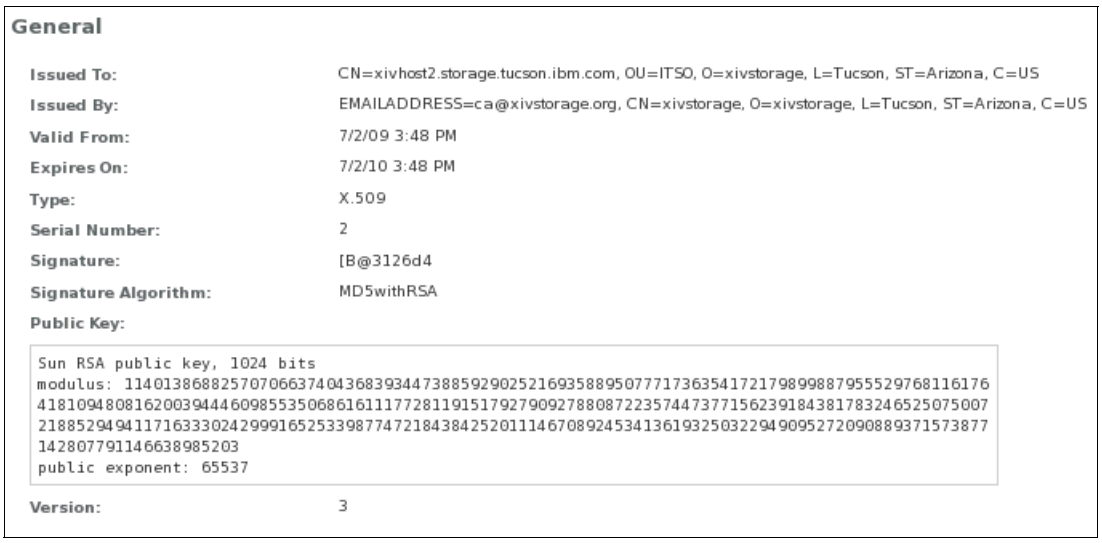


Figure A-17 Signed SUN Java Directory certificate information

To check the xivstorage certificate, open the **Directory Servers** → **xivhost2.storage.tucson.ibm.com:389** → **Security** → **CA Certificates** and click on `xivstorage.org sample CA Certificate Authority certificate` link. Figure A-18 shows that the certificate issued to and by the xivstorage CA is valid.

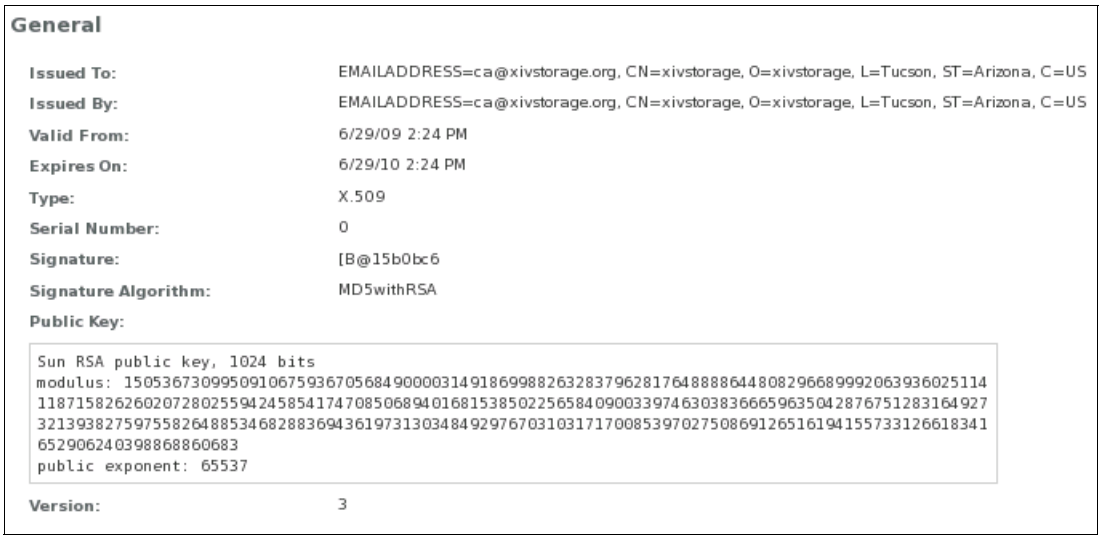


Figure A-18 Certificate information for xivstorage certificate authority

To activate imported certificate open the **Directory Servers** → **xivhost2.storage.tucson.ibm.com:389** → **Security**. In General tab in “Certificate” field expand scroll down list an select `xivstorage.org sample CA certificate` as shown in Figure A-19.

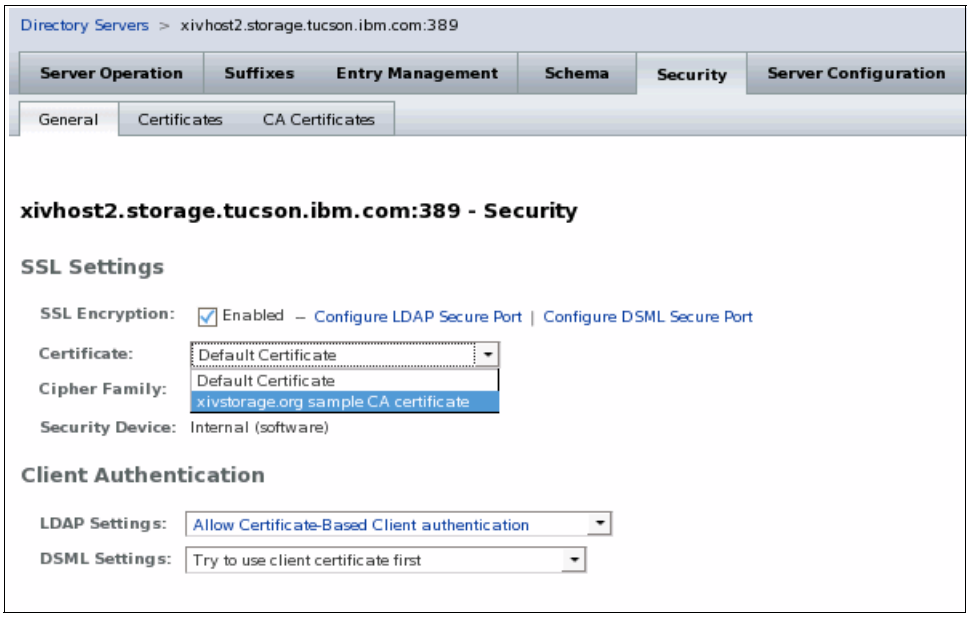


Figure A-19 New signed certificate activation

As depicted in Figure A-20, you will be prompted to restart the LDAP server in order for the new certificate to take effect

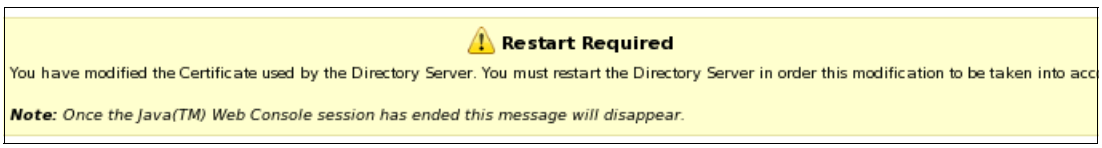


Figure A-20 Manual restart request after activating new certificate

Low-level SSL validation using the openssl command

The easiest way to test the low-level SSL connection to the LDAP server is by using the openssl s_client command with the -showcerts option. This command will connect to the specified host and list the server certificate, the certificate authority chain, supported ciphers, SSL session information, and verify return code. If the SSL connection worked, the openssl s_client command result in the verify return code will be 0 (Ok).

Example A-11 shows the output of the **openssl s_client** command connecting Linux server (xivstorage.org) to the SUN Java Directory server `xivhost2.storage.tucson.ibm.com`. This command connects to the SUN Java Directory server using the secure LDAP port (636).

Example: A-11 Low-level SSL validation using the openssl s_client

```
# openssl s_client  -host xivhost2.storage.tucson.ibm.com -port 636 -CAfile
cacert.pem  -showcerts
   ...
---
Server certificate
subject=/C=US/ST=Arizona/L=Tucson/O=xivstorage/OU=ITSO/CN=xivhost2.storage.tucson.
ibm.com
issuer=/C=US/ST=Arizona/L=Tucson/O=xivstorage/CN=xivstorage/emailAddress=ca@xivsto
rage.org
---
Acceptable client certificate CA names
/O=Sun Microsystems/CN=Directory Server/CN=636/CN=xivhost2.storage.tucson.ibm.com
/C=US/ST=Arizona/L=Tucson/O=xivstorage/CN=xivstorage/emailAddress=ca@xivstorage.or
g
---
SSL handshake has read 2144 bytes and written 328 bytes
---
New, TLSv1/SSLv3, Cipher is AES256-SHA
Server public key is 1024 bit
Compression: NONE
Expansion: NONE
SSL-Session:
    Protocol  : TLSv1
    Cipher    : AES256-SHA
    Session-ID: 48B43B5C985FE1F6BE3F455F8350A4155DD3330E6BD09070DDCB80DCCB570A2E
    Session-ID-ctx:
    Master-Key:
1074DC7ECDD9FC302781C876B3101C9C618BB074020D7062E7EA3AB794CA9C5D1A33447EE254288CEC
86BBB6CD264DCA
    Key-Arg   : None
    Krb5 Principal: None
    Start Time: 1246579854
    Timeout   : 300 (sec)
    Verify return code: 0 (ok)
---
```

Basic secure LDAP validation using the ldapsearch command

After you have confirmed that the SSL connection is working properly, you should verify that you are able to search your LDAP directory using LDAPS on port 636. This will confirm that the LDAP server can communicate using SSL connection.

In Example A-12, we use OpenLDAP client for SSL connection validation. CA certificate needs to be added to key ring file used by OpenLDAP client. TLS_CERTS option in OpenLDAP configuration file (typically /etc/openldap/ldap.conf) specifies the file that contains certificates for all of the Certificate Authorities the client will recognize.

Example: A-12 Testing LDAP over SSL using ldapsearch command

```
# /usr/bin/ldapsearch -x  -H “ldaps://xivhost2.storage.tucson.ibm.com:636” -D
'uid=xivtestuser2,dc=xivauth' -w pwd2remember -b 'dc=xivauth'

# extended LDIF
#
# LDAPv3
# base <dc=xivauth> with scope subtree
# filter: uid=xivtestuser2
# requesting: ALL
#

# xivtestuser2, xivauth
dn: uid=xivtestuser2,dc=xivauth
uid: xivtestuser2
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
sn: xivtestuser2
cn: xivtestuser2
description: custom_role_01

# search result
search: 2
result: 0 Success
```

The URI format used with “-H” option specifies that LDAPS (LDAP over SSL) must be used on port 636 (LDAP secure port).

Certificate Authority setup

This section describes the setup and use of the certificate authority that was used with all example scenarios in this book to issue certificates.

OpenSSL comes with most Linux distributions by default. Information about OpenSSL can be found at the OpenSSL Web site:

`http://www.openssl.org`

Creating the CA certificate

To set up the CA for the `xivstorage.org` domain we need to make some assumptions. We modify the `openssl.cnf` to reflect these assumptions to the CA. The file can be found at `/usr/share/ssl/openssl.cnf` and the interesting sections are shown in Example A-13.

Example: A-13 openssl.cnf

```
[ CA_default ]

dir             = /root/xivstorage.orgCA# Where everything is kept
certs           = $dir/certs            # Where the issued certs are kept
crl_dir         = $dir/crl              # Where the issued crl are kept
database        = $dir/index.txt        # database index file.
new_certs_dir   = $dir/newcerts         # default place for new certs.
certificate     = $dir/cacert.pem       # The CA certificate
serial          = $dir/serial           # The current serial number
crl             = $dir/crl.pem          # The current CRL
private_key     = $dir/private/cakey.pem# The private key
RANDFILE        = $dir/private/.rand    # private random number file
x509_extensions = usr_cert              # The extentions to add to the cert
name_opt        = ca_default            # Subject Name options
cert_opt        = ca_default            # Certificate field options
default_days    = 365                   # how long to certify for
default_crl_days= 30                    # how long before next CRL
default_md      = md5                   # which md to use.
preserve        = no                    # keep passed DN ordering
copy_extensions = copy                  # Extension copying option
default_days    = 365                   # how long to certify for
default_crl_days= 30                    # how long before next CRL
default_md      = md5                   # which md to use.
preserve        = no                    # keep passed DN ordering
   .
   .
   .
[ req_distinguished_name ]
countryName                     = Country Name (2 letter code)
countryName_default             = US
countryName_min                 = 2
countryName_max                 = 2

stateOrProvinceName             = State or Province Name (full name)
stateOrProvinceName_default     = TX
localityName                    = Locality Name (eg, city)
localityName_default            = Tucson
O.organizationName              = Organization Name (eg, company)
O.organizationName_default      = xivstorage
organizationalUnitName          = Organizational Unit Name (eg, section)
commonName                      = xivstorage.org (eg, your server's hostname)
commonName_max                  = 64
emailAddress                    = ca@xivstorage.org
emailAddress_max                = 64
```

Also, the directories to store the certificates and keys must be created:

```
mkdir /root/xivstorage.orgCA /root/xivstorage.orgCA/certs
/root/xivstorage.orgCA/crl /root/xivstorage.orgCA/newcerts
/root/xivstorage.orgCA/private
```

OpenSSL is using a couple of files, which it uses to maintain the CA. These files must be created:

```
touch /root/xivstorage.orgCA/index.txt

echo "01" >> /root/xivstorage.orgCA/serial
```

The access rights on the directories and files should be reviewed to restrict access to the CA and, most importantly, to the private key as far as possible.

To create the CA certificate certified for 365 days, the OpenSSL command is issued directly, as shown in Example A-14.

Example: A-14 Generating the CA certificate

```
openssl req -new -x509 -days 365 -keyout /root/xivstorage.orgCA/private/cakey.pem
-out /root/xivstorage.orgCA/cacert.pem
Generating a 1024 bit RSA private key
....++++++
.....................++++++
writing new private key to '/root/xivstorage.orgCA/private/cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:
State or Province Name (full name) [Arizona]:
Locality Name (eg, city) [Tucson]:
Organization Name (eg, company) [xivstorage]:
Organizational Unit Name (eg, section) []:
Common Name (eg, YOUR name) []:xivstorage
Email Address []:ca@xivstorage.org
```

During the creation of the certificate missing information must be provided. Also, the information that has been defined as default in the openssl.cnf file must be confirmed. The password for the CA private key must be given during the creation process. This password is needed whenever the CA's private key is used. The following command can be used to view CA certificate: "`openssl x509 -in cacert.pem -text`"

Signing a certificate

The client or server that needs to get a certificate must create a certificate signing request and send this request to the CA.

Certificate request details can be viewed using the following command:

```
openssl req -in xivhost1_cert_req.pem -text
```

`xivhost1_cert_req.pem` is the certificate signing request, the file generated on xivhost1.xivhost1ldap.storage.tucson.ibm.com server

Signing a certificate for xivhost1 server

To sign the certificate, the **openssl** command is used with a specified policy, as shown in Example A-15.

Example: A-15 Signing certificate for xivhost1 server

```
# openssl ca -policy policy_anything -cert cacert.pem -keyfile private/cakey.pem
-out  xivhost1_cert.pem -in xivhost1_cert_req.pem
Using configuration from /usr/share/ssl/openssl.cnf
Enter pass phrase for private/cakey.pem:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 1 (0x1)
        Validity
            Not Before: Jun 29 21:35:33 2009 GMT
            Not After : Jun 29 21:35:33 2010 GMT
        Subject:
            commonName                =
xivhost1.xivhost1ldap.storage.tucson.ibm.com
        X509v3 extensions:
            X509v3 Basic Constraints:
            CA:FALSE
            Netscape Comment:
            OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
            C8:EB:8D:84:AB:86:BB:AF:5B:74:4D:35:34:0E:C5:84:30:A1:61:84
            X509v3 Authority Key Identifier:
            keyid:A8:0B:D1:B5:D6:BE:9E:61:62:E3:60:FF:3E:F2:BC:4D:79:FC:E3:5A

DirName:/C=US/ST=Arizona/L=Tucson/O=xivstorage/CN=xivstorage/emailAddress=ca@xivst
orage.org
            serial:00

            X509v3 Extended Key Usage:
            TLS Web Server Authentication
            X509v3 Key Usage:
            Digital Signature, Key Encipherment
Certificate is to be certified until Jun 29 21:35:33 2010 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

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

For information about ordering this publication, refer to “How to get IBM Redbooks publications” on page 386. This document might be available in softcopy only:

- *Introduction to Storage Area Networks*, SG24-5470
- *IBM Tivoli Storage Productivity Center V4.1 Release Guide*, SG24-7725

Other publications

These publications are also relevant as further information sources:

- *IBM XIV Storage System Installation and Service Manual*, GA32-0590
- *IBM XIV Storage System XCLI Utility User Manual 2.4*, GA32-0638-01
- *IBM XIV Storage System XCLI Reference Guide*, GC27-2213-02
- *IBM XIV Storage Theory of Operations*, GA32-0639-03
- *IBM XIV Storage System Installation and Planning Guide for Customer Configuration, GC52-1327-03*
- *IBM XIV Storage System Pre-Installation Network Planning Guide for Customer Configuration,* GC52-1328-01
- *Host System Attachment Guide for Windows- Installation Guide*:

  http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp
- *The iSCSI User Guide* :

  http://download.microsoft.com/download/a/e/9/ae91dea1-66d9-417c-ade4-92d824b871af/uguide.doc
- *AIX 5L System Management Concepts: Operating System and Devices*:

  http://publib16.boulder.ibm.com/pseries/en_US/aixbman/admnconc/hotplug_mgmt.htm#mpioconcepts
- *System Management Guide: Operating System and Devices for AIX 5L*:

  http://publib16.boulder.ibm.com/pseries/en_US/aixbman/baseadmn/manage_mpio.htm
- *Host System Attachment Guide for Linux*, which can be found at the XIV Storage System Information Center:

  http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp
- *Fibre Channel SAN Configuration Guide*:

  http://www.vmware.com/pdf/vi3_35/esx_3/r35u2/vi3_35_25_u2_san_cfg.pdf
- *Basic System Administration* (VMware Guide):

http://www.vmware.com/pdf/vi3_35/esx_3/r35u2/vi3_35_25_u2_admin_guide.pdf

- *Configuration of iSCSI initiators with VMware ESX 3.5 Update 2*:
  http://www.vmware.com/pdf/vi3_35/esx_3/r35u2/vi3_35_25_u2_iscsi_san_cfg.pdf
- *ESX Server 3 Configuration Guide*:
  http://www.vmware.com/pdf/vi3_35/esx_3/r35u2/vi3_35_25_u2_3_server_config.pdf

Online resources

These Web sites are also relevant as further information sources:

- IBM XIV Storage Web site:
  http://www.ibm.com/systems/storage/disk/xiv/index.html
- System Storage Interoperability Center (SSIC):
  http://www.ibm.com/systems/support/storage/config/ssic/index.jsp
- SNIA (Storage Networking Industry Association) Web site:
  http://www.snia.org/
- IBM Director Software Download Matrix page:
  http://www.ibm.com/systems/management/director/downloads.html
- IBM Director documentation:
  http://www.ibm.com/systems/management/director/

How to get IBM Redbooks publications

You can search for, view, or download IBM Redbooks publications, Redpapers, Technotes, draft publications and Additional materials, as well as order hardcopy IBM Redbooks publications, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z

IBM XIV Storage System: Architecture, Implementation, and Usage

(0.5” spine)
0.475”<->0.873”
250 <-> 459 pages

IBM XIV Storage System: Architecture, Implementation, and Usage

Non Disruptive Code load GUI and XCLI improvments

Support for LDAP authentication TPC Integration

Secure Remote Support

This IBM Redbooks publication describes the concepts, architecture, and implementation of the IBM XIV Storage System (2810-A14 and 2812-A14), which is designed to be a scalable enterprise storage system based upon a grid array of hardware components. It can attach to both Fibre Channel Protocol (FCP) and iSCSI capable hosts.

In the first few chapters of this book, we provide details about many of the unique and powerful concepts that form the basis of the XIV Storage System logical and physical architecture. We explain how the system was designed to eliminate direct dependencies between the hardware elements and the software that governs the system.

In subsequent chapters, we explain the planning and preparation tasks that are required to deploy the system in your environment. This explanation is followed by a step-by-step procedure of how to configure and administer the system. We provide illustrations of how to perform those tasks by using the intuitive, yet powerful XIV Storage Manager GUI or the Extended Command Line Interface (XCLI).

The book contains comprehensive information on how to integrate the XIV Storage System for authentication in an LDAP environment and outlines the requirements and summarizes the procedures for attaching the system to various host platforms.

We also discuss the performance characteristics of the XIV system and present options available for alerting and monitoring, including an enhanced secure remote support capability.

This book is intended for those individuals who want an understanding of the XIV Storage System, and also targets readers who need detailed advice about how to configure and use the system.

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

SG24-7659-01 ISBN 0738433373