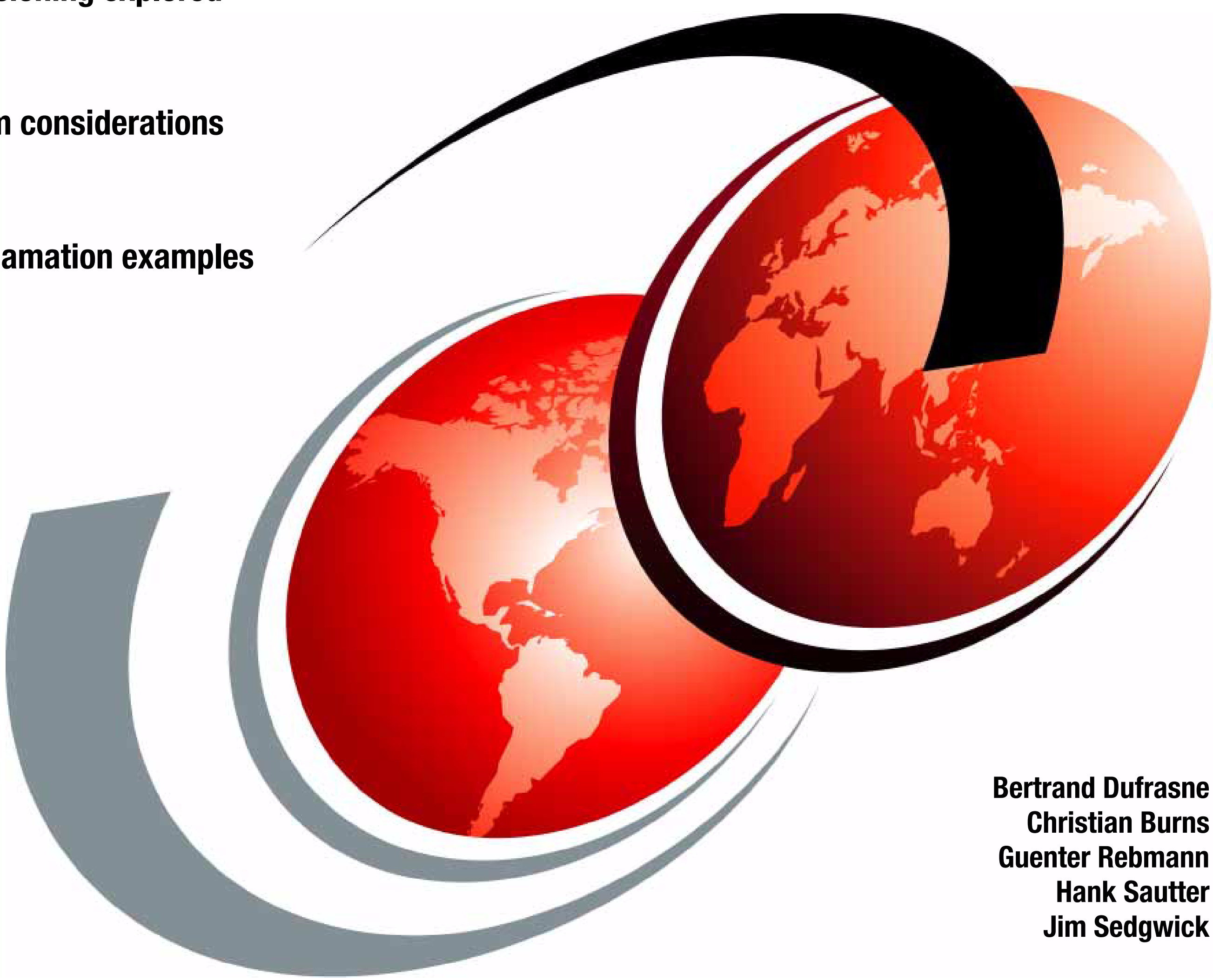
IBM XIV Storage System Thin Provisioning and Space Reclamation

Thin provisioning explored

File system considerations

Space reclamation examples



Bertrand Dufrasne
Christian Burns
Guenter Rebmann
Hank Sautter
Jim Sedgwick

Redpaper

ibm.com/redbooks

International Technical Support Organization

IBM XIV Storage System Thin Provisioning and Space Reclamation

June 2013

REDP-5001-00

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

First Edition (June 2013)

This edition applies to Version 3.2 of the IBM XIV Storage System with Storage Software Version 11.2.

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 grant you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

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

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

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

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

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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

Preface

Thin provisioning is the practice of passing logical unit number (LUN) sizes up to application servers without actually reserving the total physical capacity of those LUNs for data storage. Thin provisioning is a popular feature of IBM® XIV® Storage System.

Data *space reclamation* helps you enjoy the benefits of thin provisioning. Space reclamation is a storage system function to reclaim a specific amount of disk space for general-purpose use after being notified by the file system that the disk space was deleted at the host level.

Because thin provisioning and support for space reclamation are so tightly related in the XIV storage system, this IBM Redpaper™ publication explores both concepts in detail.

This publication is intended for system and storage administrators who want to take advantage of the XIV storage system functionality in thin-provisioned environments, coupled with the latest space reclamation enhancements.

Authors

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

Bertrand Dufrasne is an IBM Certified Consulting IT Specialist and Project Leader for IBM System Storage® disk products at the ITSO, San Jose Center. He has worked at IBM in various IT 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 Master's degree in Electrical Engineering.

Christian Burns is an IBM Storage Solution Architect based in New Jersey. As a member of the Storage Solutions Engineering team based in Littleton, MA, he works with clients, business partners, and IBM employees worldwide, designing and implementing storage solutions that include a variety of IBM products and technologies. Christian's areas of expertise include real-time compression, SAN Volume Controller (SVC), XIV, and IBM FlashSystem™. Prior to joining IBM, Christian was the Director of Sales Engineering at Storwize. He brings over a decade of industry experience in the areas of sales engineering, solution design, and software development. Christian holds a BA degree in Physics and Computer Science from Rutgers College.

Guenter Rebmann is a certified XIV Product Field Engineer in Germany. He joined IBM in 1983 as a Customer Engineer for large-system clients. After 10 years of experience with all large system products, he joined the DASD-EPSG (EMEA Product Support Group) in Mainz. In 2009 he became a member of the XIV PFE EMEA-Team. He has more than 20 years of experience providing technical support for past and present high-end DASD products.

Hank Sautter is a Consulting IT Specialist with Advanced Technical Support in the USA. He has worked at IBM for 34 years. He worked for 20 years in Tucson, Arizona with IBM S/390® and IBM disk storage hardware and Advanced Copy Services functions. His previous years of experience include IBM Processor microcode development and S/390 system testing while working in Poughkeepsie, NY. Henry'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 from Texas A&M.

Jim Sedgwick is an IT Specialist in the US. He has more than 20 years of experience in the storage industry. He worked for five years with IBM as a Printer Design Engineer after receiving his Mechanical Engineering degree from NCSU. Jim's current areas of expertise include enterprise storage performance and copy services. He writes and teaches on both subjects.

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

Eric Johnson, Betty Porat, Tedd Gregg, Hans-Paul Drumm, Moriel Lechtmann, and Shai Harony
IBM

Now you can become a published author, too!

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

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

http://**ibm.com**/redbooks/residencies.html

Comments welcome

Your comments are important to us!

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

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

  http://**ibm.com**/redbooks

- Send your comments in an email to:

  mailto:redbooks@us.ibm.com

- Mail your comments to:

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

Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/IBMRedbooks

- Follow us on Twitter:

  http://twitter.com/ibmredbooks

- Look for us on LinkedIn:

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

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

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

- Stay current on recent Redbooks publications with RSS Feeds:

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

1

Thin provisioning overview

For most businesses, managing a storage environment is an endless balancing act of providing enterprise-class functionality and services but still controlling costs. One area that is a considerable component in capital and operational costs is unused storage space that is allocated to applications but is not being used.

Thin provisioning is the practice of allocating storage to applications on a just-in-time and as-needed basis by defining a logical capacity that is larger than the physical capacity. This common practice is a means to reducing or postponing storage costs.

Tip: Thin provisioning is a no-charge feature of IBM XIV Storage System software.

XIV storage has a native virtual architecture. XIV thin provisioning is nothing more than a different way of displaying the XIV storage pools to the XIV management GUI. The underlying data structures do not change between XIV regular provisioning and thin provisioning. Any XIV storage pool can be changed from regular provisioning to thin provisioning. This reconfiguration is just a logical change, yet there are important considerations and best practices associated with XIV Storage System thin provisioning that you need to understand.

Following an introduction to thin provisioning, this chapter covers these topics:

- The business case for thin provisioning
- How the XIV system implements thin provisioning
- XIV thin provisioning considerations
- XIV thin provisioning planning, implementation, and useful procedures, including an implementation example

1.1 Thin provisioning introduction

Thin provisioning is becoming a widely accepted practice for data storage allocation because the cost and administrative benefits are easily realized. Thin provisioning is the practice of providing *virtual capacity* to a logical volume (logical unit number, or LUN) and passing that LUN to a host server. The host sees the full capacity of the LUN when, in fact, the LUN is only backed with partial capacity, or the capacity is dynamically allocated as data is written. In this way, thin provisioning provides significant improvements in data storage efficiency by overcoming the traditional problem of hosts only partially using the capacity of the LUNs assigned to them.

> **XIV:** The XIV system provides a highly flexible, cost-optimizing approach to thin provisioning.

This approach enables you to allocate capacity based on the total space actually used rather than just the space allocated. The result is improved storage utilization rates, leading to greatly reduced capital and operational costs.

Thin provisioning was a design requirement of the original XIV architecture. XIV is a virtual storage system with data awareness. Volumes, storage pools, and thin provisioning are all simply logical entities built around the data stored in the XIV system. An XIV storage pool is either defined as being thin provisioned, or fully provisioned. During normal operations, you can change an XIV storage pool from full provisioning to thin provisioning, or as is commonly said, from thin to full. At any time, you can move a simple volume from a thin pool to a full pool and back again.

You can implement these reconfigurations without any performance considerations. An XIV storage pool is purely a logical entity, just like the rules that define the behavior of thin and full storage pools. XIV storage pools are not associated with any physical entity, such as a disk drive, a RAID array, or an XIV module. All storage pools and XIV volumes are distributed evenly among all hardware components in the XIV system.

Thin provisioning is a standard, no-cost feature of XIV Storage System software. Thin provisioning can safely increase your storage utilization by up to 50% without requiring you to purchase more storage capacity or change application or user behavior. Thin provisioning is implemented on the storage level and is completely transparent to the host using the volume.

There are a couple of good similes to better understand thin provisioning. The first is that thin provisioning is like virtual memory in an operating system (getting something for nothing). The other is that thin provisioning is like when airlines overbook seats on a flight (which is generally a good thing, but can lead to problems if not managed properly).

1.2 The business case for thin provisioning

There are several business situations that can justify storage thin provisioning. This section reviews some of those situations and shows why XIV is usually a good fit.

1.2.1 IT without thin provisioning: costly and complex

As an IT manager or storage administrator, you must manage the cost of storage. You are likely faced with burgeoning organizational demands for storage capacity yet, expected to curb storage expenses.

In this section, we examine how traditional approaches to storage allocation and thin provisioning actually inflate costs.

Capacity gone to waste

One of the disturbing facts about storage costs is that 70% or more of the storage capacity in enterprise settings goes unused. Storage space is allocated to an application, but the application uses only a portion of it. This waste is caused by several factors:

- Pre-allocating for future requirements

  It is a common practice to pre-allocate storage space for the application based on future requirements. This is because resizing volumes is a difficult and time-consuming IT operational task, for both storage and applications.

- Over-estimating needs

  Storage space allocation is typically based on a prediction of future use. If this prediction is inaccurate and the allocated storage goes unused, there is no practical way to reclaim the allocated storage for other applications.

- Full copies of unwritten data

  Many application needs, such as testing and data mining, require copies of the original data, either at the application level (file copy and so on) or the storage level (snapshots and full copies). Traditional architectures copy both used and unused data, thus increasing the amount of unused space.

These reasons help explain why storage, one IT's most expensive and complex resources, is underutilized to a large extent. The cause is not inappropriate management practice, but the very nature of the pre-allocation of storage and the inability to easily reclaim this storage if it goes unused.

The real cost of unused storage

Although unused storage is itself costly, there are many additional hidden and indirect costs:

- Capital expense

  You can reduce the direct capital expense of storage by purchasing only the storage required for actual usage. For example, if you have a storage project with a 70% non-utilization rate and a budgeted capital expense of $1,000,000, you can implement the project using a thin-provisioned system for only $300,000.

- Environmental factors

  The unused capacity consumes electricity and generates heat, adding to power and cooling expenses. As with capital expenses, a storage project with a 70% non-utilization rate costs only 30% of the power and cooling if implemented with thin provisioning.

- Floor space

  The same analysis applies to savings in floor space, which is another expensive resource (involving the cost of building, raised floors, uninterrupted power supply systems, cooling systems, and so on) that is often in short supply.

- Declining storage costs over time

  Without thin provisioning, you must purchase storage at the time of allocation, sometimes years before it is used. This scenario involves doubling your costs: not only are power, cooling, and floor space wasted during the time of the storage not being used, but your organization is unable to take advantage of yearly declines in the cost of storage. Also, the power and space efficiency of storage systems improves over time; these advantages go untapped when you buy storage up front.

1.2.2 Advantages of thin provisioning in the XIV system

XIV Storage System software provides many capabilities to improve data storage efficiency, including usage monitoring, storage pools, reclaiming capacity, and snapshots.

Capacity usage monitoring

The XIV storage system has robust capabilities to monitor pool storage consumption constantly, and enables the storage administrator to configure the system to send notifications when space utilization exceeds a user-defined threshold. These notifications can be sent to various destinations as simple network management protocol (SNMP) traps, emails, or text messages. The severity and destination of the notification depends on the actual consumption threshold passed. The whole idea of thin provisioning monitoring is to never run out of physical capacity.

You can configure the XIV system so that, if a given event notification is not handled within a certain time period, another event notification is sent to the same addressees or to a broader distribution list, like a manager.

Thin provisioning limited to specific applications

Another important feature of the XIV storage system is that you can manage thin provisioning per *storage pool*. The concept of a storage pool is unique to the XIV system: a storage pool is a logical entity, defined in the system, which contains a group of volumes and their snapshots. Each storage pool limits the amount of space that the volumes and snapshots belonging to the pool can consume.

Within a production XIV environment, it is common to have a mix of fully provisioned storage pools and thin-provisioned storage pools.

You can define a thin provisioning policy per storage pool. Each pool has its own hard capacity (which limits the actual disk space used) and soft capacity (which limits the total size of volumes defined).

Separating thin provisioning per storage pool is essential in limiting the effect of running out of physical disk space. Thin provisioning management is performed per pool and running out of space in one pool does not impact other pools.

An example of this is when a data center has backup-to-disk and business-critical applications running in its environment. The backup-to-disk uses thin provisioning with 100 TB of logical volumes and only 50 TB of physical space. Application XYZ, a business-critical application, needs only 10 TB and thin provisioning is not used. With a poorly managed thin provisioning implementation, backup-to-disk storage potentially impacts the XYZ application. Such a scenario is unacceptable.

Some applications are simply too important to exist in thin-provisioned pools. With the XIV system, you can use two storage pools: one for the backup-to-disk and one for the XYZ application. The XYZ storage pool will have identical soft and hard capacities, and thus never be locked as a result of running out of physical space. Alternatively, you can configure the storage pool used for the backup-to-disk application with thin provisioning (that is, with soft capacity larger than hard capacity) and receive all the advantages of thin provisioning.

It can be easy for you to administer and manage storage pools, because pools are a purely logical entity. You can always resize a storage pool, move storage capacity between storage pools, or move volumes between storage pools. Pools are not associated with any physical entity, such as a disk drive or module. In fact, all storage pools are distributed equally in the XIV system among all hardware components.

The ability to reclaim logically unused capacity

The XIV system also has a sophisticated mechanism to reclaim unused areas of the volume, even after they have been defined as *used*. The effect of this feature is that you can decrease used capacity if used data is logically erased by writing all zeros. You can use this capability to reclaim space in several scenarios:

- When migrating volumes from existing storage equipment to XIV storage, parts of the volume that are all zeros are marked *unused*, and will not consume physical space
- When an application writes a long sequence of zeros, the relevant part is marked unused, even if previous information was not zero

As a background process, the XIV storage system scans volumes and searches for long areas that are all zeros, marking these areas unused.

This reclaiming process provides huge potential savings, especially when you migrate storage from existing systems. The operational cost reduction (power, cooling, and floor space per TB) associated with the XIV system is not the only advantage of thin provisioning. An additional advantage is your ability to take an existing volume on an existing system, migrate it to XIV storage, and reclaim unused space (requiring, for instance, only 30% of the original capacity).

XIV thin provisioning and snapshots

The XIV architecture seamlessly integrates thin provisioning with *snapshots*. The system automatically thin provisions snapshots, using physical space only when a delta exists between the snapshot and the master volume, or between subsequent snapshots of the same volume. For more information about the XIV system’s snapshot implementation and features, see the IBM Redbooks publication, *IBM XIV Storage System Gen3 Architecture, Implementation, and Usage*, SG24-7659.

1.2.3 Advantages and disadvantages of thin provisioning

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

The XIV system implementation of thin provisioning provides the following benefits beyond those enumerated in "The real cost of unused storage" on page 3:

- You can dynamically increase or decrease capacity associated with specific applications or departments 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 designed as a globally available pool, thinly provisioned resources share a "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 systems, 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.
- You can more effectively defer capacity acquisition and deployment until actual application and business needs demand additional space. Acquisition costs, operating costs, cooling costs, and administration costs all decrease with thin provisioning.
- Thin provisioning defers the time in which the actual physical space has to be allocated. At the same time, from the application owner's perspective, the current and future demands have already been met.
- The XIV storage system has no performance penalty associated with using thin provisioning.

Thin provisioning has inherent risks that you need to understand and mitigate with proper storage administrative practices and procedures:

- Thin provisioning involves the risk that the actual storage usage will grow unexpectedly and rapidly, and will reach the actual physical data storage capacity of the storage system with still further, unsatisfied demand.
- You need to identify and correct any situations where space utilization is approaching the hard data storage capacity of the system. This requires some administrative effort on a regular basis to monitor the system space utilization.
- If the space utilization actually reaches the hard data storage capacity of the storage system, migration off the storage system to another storage system with free space is required, involving significant administrative effort. The online migration capabilities of XIV make this task much easier.
- If the space utilization actually reaches the hard data storage capacity of the storage system, any running application will stop in a way that cannot be predicted. At a minimum, the application will stop and application downtime will be experienced unexpectedly. This is never a pleasant experience.

Summary

Thin provisioning is an important technological improvement that can solve many storage problems. The ability to separate the logical view of the system from the actual capacity provisioned and procured provides you a significant cost savings, and enhances ease of management.

However, traditional thin provisioning comes with several management challenges. Thin-provisioned systems need to be carefully planned and managed, because new risks for storage availability exist when thin provisioning is deployed.

The XIV system provides a thin provisioning architecture that is tightly coupled with advanced snapshot and snapshot management technologies. With the XIV storage system, you can achieve the following improvements:

- Use thin provisioning to limit storage expenses to the actual capacity used
- Reduce required management resources and the operational expense of resizing volumes
- Limit the effect of thin provisioning to specific applications, safe-guarding your most critical applications from running out of space
- Get notifications on thin provisioning usage
- Reclaim capacity that is no longer in use

With the XIV system, you can easily deploy thin provisioning and manage it effectively, resulting in significant storage cost savings in your enterprise IT environment.

2

Thin provisioning in XIV

This chapter provides information about how you can implement thin provisioning in the XIV system.

2.1 Thin provisioning implementation in XIV

To begin our discussion of XIV thin provisioning, it is a good idea to start with some definitions.

Thin provisioning is a method for optimizing storage utilization by allocating to a volume only the space required to hold its actual data, deferring additional space allocation to the time that it is needed. Thin provisioning is implemented on the storage level, and it is transparent to the host using the volume.

Over provisioning is the ability to provide storage pools (and volumes inside the pools) with an accumulated size larger than the physical capacity available on the system.

The XIV storage system supports both thin provisioning and over provisioning:

- Define volume sizes that are larger than the physical capacity of the of the storage pool
- Define pools (and volumes within them) with an accumulated size larger than the physical capacity available on the system

There is little benefit to thin provisioning without over provisioning. Thin provisioning is the mechanism for implementing thin-provisioned storage pools. This is also referred to as *basic provisioning*. Over provisioning can only be configured by IBM support. Over provisioning is referred to as *advanced provisioning*. The concept of thin provisioning commonly covers both basic and advanced provisioning.

2.1.1 Logical and actual volume sizes

In traditional volumes, the assigned physical capacity is equivalent to the logical capacity presented to hosts. This situation does not have to be the case with XIV thin provisioning. For a given XIV logical volume, there are effectively two associated sizes, the logical volume size and the actual volume size. 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 volume's storage pool is a thin pool or a regular pool. The volume size is specified in one of two ways, depending on units:

- In terms of gigabytes: The system allocates 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 allocates 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, although it effectively limits the hard capacity consumed collectively by snapshots as well.

Actual volume size

The actual volume size is 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. Volume size starts from zero at volume creation 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 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 in1 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.
- Application write access patterns determine the rate at which the allocated hard volume capacity is used, and therefore 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 is 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 grows 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.

2.1.2 Thinly provisioned storage pools

Although the system effectively thinly provisions volumes automatically, you can define storage pools (when using the XIV Storage Management GUI, as shown in Figure 2-1 on page 12) as either *regular* or *thinly provisioned.*

When you use the XIV 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 reflects the current usage by hosts, because the capacity is dynamically used as required. However, the remaining deallocated 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, you can realize benefits 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.

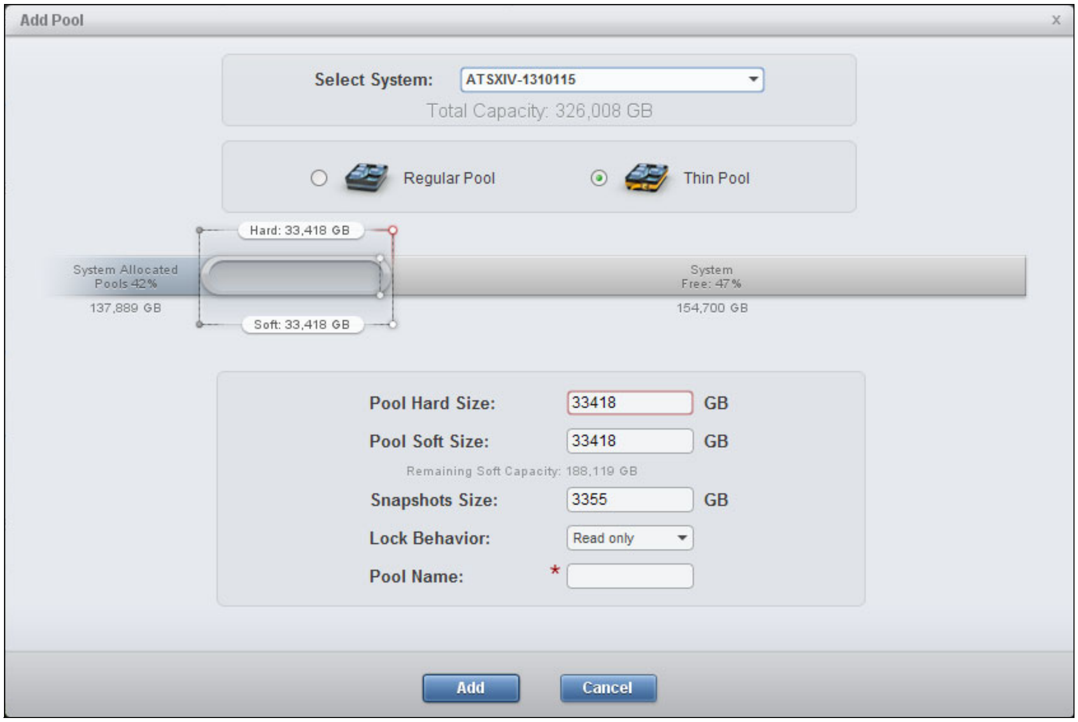


Figure 2-1 Define storage pool

The virtualized XIV data storage architecture provides you with wide flexibility in the use of thin provisioning:

- You can mirror volumes in thin pools. There are no restrictions related to which part of the mirroring relationship has thin volumes.
- You can move volumes from regular pools to thin pools, and back again.
- You can change pools from regular provisioning to thin provisioning, and back again.

Figure 2-2 on page 13 shows a regular storage pool and a thin pool:

- In the regular pool, the host system sees a 34 GB LUN (logical unit number), a 51 GB LUN, and a 68 GB LUN. The storage pool size is the total of all three LUNs, which is 153 GB, and about 40% of this storage is used.
- In the thin pool, the host system sees the same three LUN sizes, and the total storage pool size is also 153 GB. The difference is that the total space corresponding to unused portions of each LUN, 34 GB, 51 GB, and 68 GB in XIV, is not dedicated to those three LUNs, but remains available for other storage purposes.

When you create a storage pool 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).

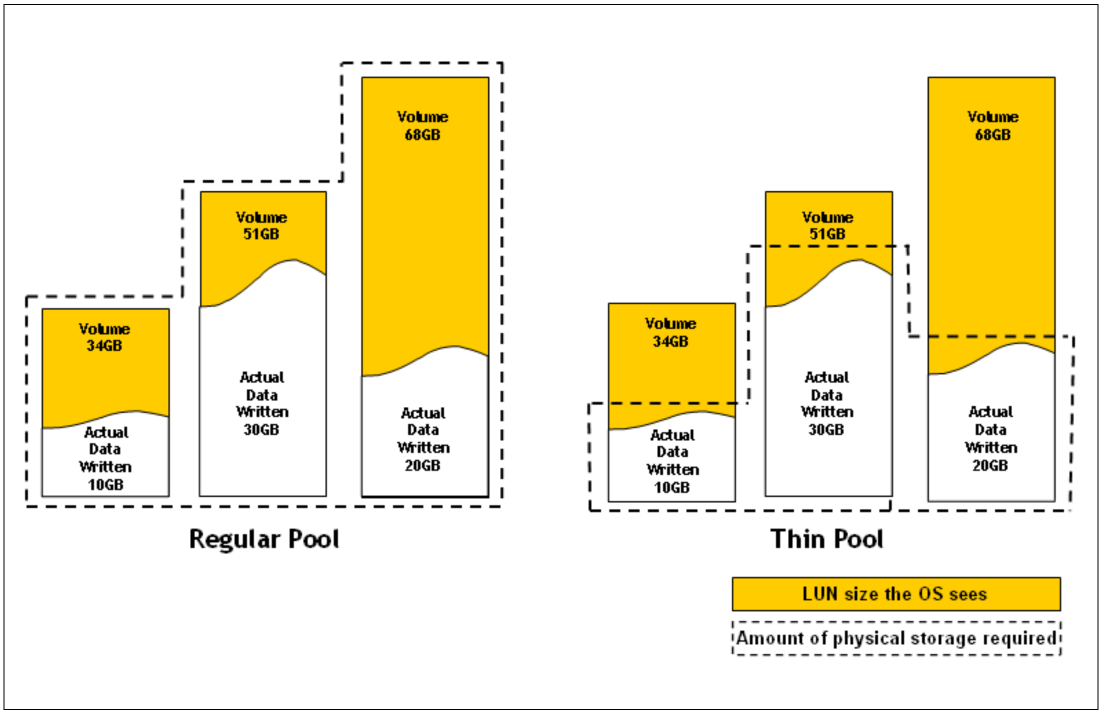


Figure 2-2 Regular pool versus thin pool

Hard pool size

Hard pool size is the maximum actual capacity that can be used by all of the volumes and snapshots in the pool. As long as there is hard system capacity available in the XIV system, you can increase the hard pool size dynamically. The hard pool size must always exceed the total amount of data written to all volumes in the pool.

Thin provisioning of the storage pool maximizes capacity utilization in the context of a group of volumes, where 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 hard pool 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. The aggregate space that is assignable to host operating systems is specified by the storage soft pool size.

The XIV System Management GUI will not allow the hard pool size to be decreased to a value less than the current amount of pool hard data.

Regular storage pools 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 allow 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 can be allowed to overwrite snapshots if necessary as hard space is used. The hard space that is allocated to the storage pool that is unused (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 hard pool size as needed in advance.

Storage pools and snapshots:

- 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 used only when changes occur to the master volumes or the snapshots themselves, not when snapshots are created.
- See *IBM XIV Storage System: Copy Services and Migration*, SG24-7759 for a more detailed explanation of snapshot deletion priority.

Soft pool size

Soft pool size is the maximum logical capacity that can be *assigned* to all the volumes and snapshots in the pool. The soft pool size must be equal to or larger than the total of the committed volume sizes in the storage pool.

Thin provisioning is managed for each storage pool independently of all other storage pools:

- Regardless of any unused capacity that might be located in other storage pools, snapshots within a given storage pool are deleted by the system according to the corresponding snapshot preset priority if the hard pool size contains insufficient space to create an additional volume or increase the size of an existing volume. (Snapshots are deleted only when a write occurs under those conditions, and not when allocating more space).
- As described in "Thinly provisioned storage pools" on page 11, the storage administrator defines both the soft size and the hard size of thinly provisioned storage pools and allocate resources to volumes within a given storage pool without any limitations imposed by other storage pools.

You can dynamically change the designation of a storage pool as a regular pool or a thinly provisioned pool.

- When you need to convert a regular pool to a thinly provisioned pool, you must explicitly set the soft pool size parameter in addition to the hard pool size, which remains unchanged unless updated.
- When you need to convert a thinly provisioned pool 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 hard pool size, the storage pool cannot be converted. This situation can be resolved if you selectively resize or delete individual volumes, or move them to another storage pool, to reduce the soft space used.

Hard system size

The *hard system size* represents the physical disk capacity that is available within the XIV system. This is the amount of data storage capacity that is purchased. Obviously, the system's hard capacity is the upper limit of the aggregate hard capacity of all the volumes and snapshots, and you can only increase it by installing new hardware components in the form of individual modules (and associated disks) or groups of modules.

There are situations (such as rebuilds and redistributions) that can *temporarily* reduce the system's hard limit.

Soft system size

The *soft system size* is the total, "global," and 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, realizing the benefits of thin provisioning of storage pools and volumes at the system level.

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

- Soft size is not related to any direct system attribute, and you can define it to be larger than the hard system size if thin provisioning is implemented. The storage administrator cannot set the soft system size.
- Ideally, in a partially populated XIV configuration using thin provisioning, when the limits of the XIV system's soft size is being challenged by the growth of the hard data storage requirements, your can purchase additional modules to alleviate the danger of running out of hard capacity.

> **Important:** Upgrading an XIV system beyond the full 15 modules in a single frame is not supported at the time of this writing.

- The soft system size is a purely logical limit, however, and you must exercise care when the soft system size is set to a value greater than the maximum potential hard system size. It is for this reason that defining the soft system size is not within the scope of the storage administrator role.
- If you want to increase the soft system size beyond the maximum hard system size for a particular XIV system model, there is a procedure that IBM performs that can accomplish this task. This change has no impact on the XIV operation. Contact your IBM Technical Advisor for further details.

2.1.3 System-level thin provisioning

The XIV system has the concept of *hard capacity* and *soft capacity*. Hard capacity, or hard system size, has to do with the maximum physical data storage capacity of the XIV system. Soft capacity, or virtual capacity, is the maximum limit of LUN capacity that you can configure on the XIV system.

The XIV architecture enables you to define a global system capacity in terms of both a *hard system size* and a *soft system size*. When over provisioning is not activated at the system level, these two sizes are equal to the system's physical capacity. The system capacity graphics from a regularly provisioned system are shown in Figure 2-3. Notice that the total system hard and soft sizes are both 131,418 GB. The hard size and the soft size can be toggled by selecting the up and down arrows on the right side of either graphic. Because these two values are equal, this XIV system has not activated thin provisioning at the system level.

Figure 2-3 Regular provisioning

When the system soft size is greater than the system hard size, the XIV system, by definition, is over-provisioned. In Figure 2-4, notice that the hard system limit is 159,726 GB and the soft system limit is set at 326,008 GB.

Figure 2-4 System level over provisioning

It is possible, and even common, for the system hard and soft capacity settings to be equal, and still use XIV thin provisioning. This is called *basic mode* thin provisioning. Indeed, when you first experiment with XIV thin provisioning, this is the usual case. You realize XIV thin provisioning benefits, but they will not be as significant as when the system soft capacity is extended beyond the hard capacity and storage capacity can be offered to host systems that exceeds the amount of purchased hard capacity. This is called *advanced mode* thin provisioning or system over provisioning.

Clients who use regular provisioning usually begin to think about over provisioning when the system soft capacity approaches the system hard capacity and space becomes a premium. In general, this is when the system hard capacity utilization is only around 50%. Before the XIV system, when you were no longer able to allocate more storage, it was time to buy more storage.

The XIV system so clearly shows the difference between soft system and hard system capacity that it almost forces you to consider thin provisioning, because it is almost irresistible to ignore the promise of using some of that storage that is allocated but is not storing any data.

Also note that 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.

As stated earlier, if you want to increase the soft system size beyond the maximum hard system size for a particular XIV system model, there is a procedure that IBM performs that can accomplish this task. This change has no impact on the XIV system operation. Contact your IBM Technical Advisor for further details.

System-level thin provisioning by definition allocates more space than is physically available. There is risk in this allocation decision. You can mitigate this risk by planning and monitoring, and by storage pool alerting mechanisms. However, the fact remains that a risk exists with thin provisioning that does not exist with regular pool provisioning. The decision to assume this risk is much easier if the XIV system has a partially populated configuration. With a partially populated XIV configuration, as hard space is depleted, it is simply a matter of purchasing more modules to add more hard capacity to the XIV system. With a fully populated XIV system configuration approaching hard space depletion, you cannot add more modules. The next section explores considerations for hard space depletion.

2.1.4 Out-of-space considerations

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. It is difficult to consider a situation where this condition can be tolerated in a production environment. Thin provisioning requires planning and monitoring.

It is important to determine the proper over-provisioning ratio for a given application environment and a realistic data growth average over time. Use the XIV storage pool threshold alerting mechanisms provided to receive warnings ahead of time. For this purpose, you need to set the thresholds according to your individual needs, which are driven by the workload and application characteristics and importance.

The thin provisioning implementation in the XIV system manages space allocation within each storage pool, so that hard capacity depletion in one storage pool never affects the hard capacity available to another storage pool.

Snapshot deletion

As mentioned previously, snapshots in regular storage pools can be automatically deleted by the system to provide space for newer snapshots. For thinly provisioned pools, snapshots can be deleted to free more physical space for volumes.

Volume locking

If you still require more hard capacity after all the snapshots in a thinly provisioned storage pool have been deleted, *all the volumes in the storage pool are locked, preventing any additional use of hard capacity.* There are two possible behaviors for a locked volume:

- Read-only (the default behavior)
- No I/O at all

In either case, your applications stop in a way that is not predictable.

> **Important:** Volume locking prevents writes to all volumes in the storage pool. Avoid this condition.

It is worth considering the differences between a regular pool running out of space and a thin pool running out of space. The size of a regular pool is determined by the application owner. If that application uses all of the space, the application normally knows how to handle the out-of-space condition. If the application has issues, it is certainly not the storage administrator's fault, because the application owner did not request adequate space.

Now consider the case where the application owner has asked for a certain amount of space and by all appearances that amount of space was delivered, but in fact, less real hard data storage capacity was delivered. This is thin provisioning. In this case, the application can run out of space unexpectedly. Although it is usually incumbent upon the application to know how to handle an out-of-space condition, it is not reasonable to expect an application to know what to do when it *unexpectedly* runs out of space when it thinks plenty of space remains. It is for this reason that XIV thin provisioning offers a choice of read-only or no I/O. It is also why the application might stop in a way that might be unpredictable. This is an uncomfortable situation for a storage administrator.

Attention: It is irresponsible for a storage administrator to allow a thin-provisioned pool to run out of real capacity.

If an application is so important that the risk of running out of space in a thin-provisioned storage pool configuration is too great, that application is simply not a candidate for thin provisioning. In the XIV system, it is common to have a mix of regular and thin provisioning.

Because storage pools are independent, thin provisioning volume locking on one storage pool never cascades into another storage pool. It is still possible for the storage administrator to intervene to increase pool hard capacity, or to redistribute hard capacity between pools.

3

Thin provisioning considerations

When you implement thin provisioning, there are variables that result in a range of the degree of success. This chapter contains information about these variables, which include:

- Performance
- Application considerations
- IBM XIV Storage System Generation 2 and Gen3 differences
- Data reclamation
- File system considerations
- User data differences

3.1 Performance

There are no performance-related considerations associated with XIV system thin provisioning. This is because the XIV architecture provides a certain performance level that has to do with the data distribution, the algorithms that handle the I/O, the performance capabilities of the XIV hardware, and the nature of the I/O characteristics, to name a few. All of the logical data constructs like a regular pool, a thin pool, or a LUN (logical unit number, or volume), are virtualized logical entities that do not have any affect on performance.

3.2 Application considerations

The most important enterprise applications are not typically the first choice for thin provisioning. Here are examples of applications, or parts of applications, that might make good candidates for thin provisioning:

- Test and development environments
  - Unpredictable data size
  - Dynamic use
- Crisis projects
  - Often over-request space due to no adequate planning time
- Backup application storage pool space

The efficiency of thin provisioning is dependent on the behavior of your applications in general, especially in the way that they allocate space for their data. Carefully select applications whose characteristics can actually benefit from thin provisioning. The application that performs this function is typically a file system, or a database in cases where a file system is not used.

Ideally, the space allocation of the applications should have the following characteristics to benefit from thin provisioning:

- The application tends to localize its data in contiguous regions, rather than scatter the data across the assigned volume.
- The application tends to reuse previously used space before using new space that was never used before. When the file system or database deletes data, the space used by this deleted data is not automatically released.
- The application storage needs tend to increase predictably over time.

The application (file system or database) might manage data in such a way that it ends up fully allocating the volume even when the stored data at any given time is actually much less than the full volume capacity.

In general, the use of a Logical Volume Manager (LVM) is not expected to have a negative impact on thin provisioning. If the LVM allows a logical volume to be subdivided into partitions, Ideally, create the partition boundaries aligned with an extent boundary of the logical volume to improve the likelihood that the extent remains unallocated. This might not be easy to determine in practice.

If a file system is used, this file system is expected to be the determining factor regarding the interaction with thin provisioning. Databases can usually operate with or without a file system. Without a file system, the database behavior is expected to be the determining factor regarding the interaction with thin provisioning. For instance, if the database has the characteristic of initializing all of its configured table spaces, any associated volumes end up fully allocated, making thin provisioning useless in this case.

> **Note:** It is beyond the scope of this Redpaper publication to identify specific software or applications with regards to their compatibility with the XIV thin provisioning feature.

3.3 XIV Storage System Generation 2 and Gen3 differences

XIV Storage System Gen3 software contains enhancements that benefit thin provisioning:

- Offsetting the beginning of where data is written within an XIV volume
- Using a prime number algorithm to ensure that no repeating data patterns can have a negative impact on data distribution within the XIV volume

In environments where there are many volumes significantly smaller than any multiple of 17 GB, the space over the 17 GB multiple is wasted. That is, given a 1 GB volume or an 18 GB volume, each wastes approximately 16 GB.

Defining pools with many such volumes as thin provisioned pools significantly reduces the amount of wasted capacity.

3.4 Data reclamation

Data reclamation is a major factor in enjoying the benefits of thin provisioning. Consider applications that write huge amounts of data, and then delete the data. Normally this space is not available for reuse, because the file system only removes the pointers to the data but all the actual data still resides on the storage system.

A few file systems support an API to a storage system notifying the storage of a space that is freed. These file systems enjoy an immediate space reclamation. VxFS, Microsoft Windows 2012, and Windows 8 are the only file systems with APIs that have been tested and certified with the XIV system. Refer to Chapter 5, “Space reclamation with XIV” on page 31.

The XIV system also has the concept of zero detection reclamation. During the XIV Storage System scrubbing process, when a partition is found that contains data composed of all zeros, the XIV system logically maps that data to metadata containing zeros, and reclaims that partition as free available hard space. In other words, in XIV storage, zeros are free and do not take up real data storage capacity.

To use the XIV storage system’s zero detection reclamation, zeros have to be written to host file system or raw volume areas before those areas are deleted. The XIV scrubbing process runs all the time, and reclaiming this zero space is automatic. These are some examples of how to “zero” space:

- On Windows: `SDelete -z`
- On UNIX or Linux: `dd if=/dev/zero of=file_name bs=2048`
- On Oracle ASM: ASRU utility

3.5 File system considerations

Some operating system file systems receive good space-saving benefits from thin provisioning, and others receive little benefit. For a file system to be *thin provision friendly*, the following general considerations apply:

- Formatting the file system (that is, including metedata) does not require allocation of a lot of space
- Writing data to the file system does not require much more space than the actual written data size
- The file system has a native reclamation method, or at least efficiently reuses free space

Table 3-1 summarizes file systems that can benefit the most from XIV thin provisioning.

Table 3-1 Systems compatible with XIV thin provisioning

File system	Gen3 thin provision compatible	Gen2 thin provision compatible	File system free space reuse	Reclamation method
NTFS	Yes	Yes	Substantial	`SDelete -z` (will be reclaimed by scrubbing)
VMFS	Yes	Yes	Not significant	Storage VMotion to new VMFS and delete old VMFS
JFS2	Yes	Yes	Significant	DD
EXT4	Yes	No	Not significant	DD
EXT3	Yes	No	Significant	DD
EXT2	Yes	No	Significant	DD
VxFS	Yes	Yes	Substantial	`vxdisk reclaim` (only for volumes without snapshots and mirroring)
XFS	Yes	No	Significant	DD
ZFS	Yes	Yes	Not significant	DD

There is no need to separate XIV volumes that can and cannot benefit from thin provisioning into different storage pools. If there are volumes in a mixed pool that can benefit from thin provisioning, make the pool a thin pool. It is normally best to group XIV volumes in pools according to application boundaries.

As for file systems, applications running on those file systems bear similar considerations in terms of being thin provisioning friendly or not. Considerations include how much space is allocated up front when initially creating files or databases, and is space freed when data is deleted or rows removed from tables.

Different systems offer the ability to reclaim space:

- IBM DB2® version 9.7 introduced a space reclamation feature that can reduce the physical space used by database files when rows are deleted.
- Oracle database with Automatic Storage Management (ASM) offers a possibility to reclaim unused space in the database by using the administrative **asru** command, which is a manual, three-phase script.

3.6 Thin provisioning for user data

As a simple example, consider user data in a business environment. Such data is similar for most users, and might consist of local mail replica, cached data from web browsers, and other application data commonly found in a business environment.

In addition, most users have similar capacity requirements for this type of data. However, some users can also have specific applications with different capacity requirements. For example, users who are working most of the time with documents, spreadsheets, and presentations require less capacity than those working with images, other multimedia data, or other binary and scientific data.

Assuming that no user is requesting more than 10 GB of data and that the average of effectively used data is less than 6 GB per user, the storage administrator might decide to allocate 10 GB of virtual capacity for each user (as thin-provisioned volumes), though in the back-end storage, the real capacity per user only amounts to 6 GB. If we assume 100 users, this means that the virtual capacity is 1 TB, but only 600 GB of real capacity is available in the storage subsystem. This is a valid approach only because the administrator knows from previous observations that not all users will request the whole amount of physical capacity at the same time. Some will only request 4 GB or less, some might request the whole 10 GB. But the assumption remains that on average they all request no more than 6 GB of real capacity.

It remains the responsibility of the storage administrator to monitor the allocation of real capacity to avoid any out-of-space condition.

4

Thin provisioning planning and usage

This chapter contains information about how you can determine an appropriate XIV system configuration with thin provisioning in a given environment, and uses an illustrated example to review implementation best practices.

4.1 Planning, implementation, and helpful processes

To determine an appropriate XIV system configuration with thin provisioning in a given environment, carefully gather the following required data. The order is not relevant at this point, but gather the information contained in these steps before you implement XIV thin provisioning.

- Determine application hard data storage and snapshot growth patterns as a part of the decision to deploy in thin pools. Monitor the storage pool used capacity size and snapshot used size until a monthly growth trend can be determined. As a rule, the used size has a general linear growth percentage. The snapshot used size likewise has a growth component. After you know the storage pool growth patterns, determine how far into the future to project growth. This can be months or years. This projection will be the hard size of your thin storage pool.
- Determine that the application is appropriate for thin provisioning. This is usually a consensus decision that includes the application owners. The most important enterprise applications are not the first choice for thin provisioning. Test and development environments are usually good thin provisioning candidates.
- Storage pools typically contain a single application. If the application is appropriate for thin provisioning, it is easy to change the single pool from regular to thin provisioning.
- Thin pools typically maintain a ratio of the pool soft size that is equal to 1.5 - 2.0 times the pool hard size.

In addition, implement these processes to improve your success:

- Keep snapshot sizes around 10% of the hard pool size unless observed growth patterns indicate larger snapshot areas are required.
- Monitor XIV system thin pools carefully. It is best to stay ahead of the alerting mechanisms.
- Ensure that system storage pool alerting is set up and working.
- Test storage pool alerts on a regular basis.
- Practice how to respond to storage pool alerts.
- Have escalation rules in place.
- Set pool alert thresholds no higher than the system defaults:
  - Warning 80%
  - Minor 90%
  - Major 95%
  - Critical at 95%. This is not a system default, but is recommended for thin provisioning.
- Follow these typical steps toward thin provisioning practice maturity:
  - Decide that thin provisioning is a storage provisioning policy option that you want.
  - Choose one or two XIV system storage pools to use to become proficient with thin provisioning. Ensure that these storage pools do not contain business-critical production applications.
  - Become familiar and comfortable with managing and monitoring thin pools.
  - Implement thin pools as a storage allocation option for new projects
  - Consider asking IBM to expand the total system soft capacity beyond the total system hard capacity when your XIV system-allocated soft capacity approaches the total

system hard and soft capacity. Typically this new system soft capacity is set at 50% over the hard system capacity.

- Have a contingency plan. As system hard capacity approaches 80%, you need more storage or high-priority intervention to alleviate this situation.
  - Add modules.
  - Delete snapshots.
  - Migrate or mirror data to another XIV system. IBM recently announced the near future ability to migrate volumes between XIV systems.
  - Host based migration capabilities.

4.1.1 Thin provisioning implementation example

XIV enables you to adapt thin provisioning technology easily. This section contains an example of how to plan and implement XIV thin provisioning. The following procedure assumes no prior experience with XIV thin provisioning.

1. Choose a non-production or non-business-critical application as the first candidate for thin provisioning.
2. Ensure that the candidate application volumes all reside in a single regular XIV storage pool. An an example, consider the regular XIV system storage pool named *Test* in Figure 4-1.

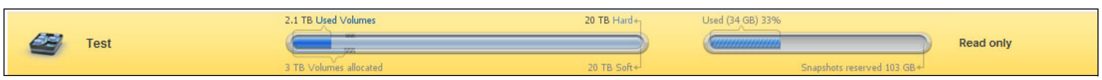


Figure 4-1 Regular storage pool: thin candidate

  - Three years ago, the Test group requested 20 TB of storage.
  - There is only one 3 TB XIV volume in the storage pool and it contains 2.1 TB of data.
  - The total volume sizes are 3 TB.
  - This data is basically used over and over again to test the product updates, and grows by less than 1% year to year.

  This storage pool is an ideal candidate for thin provisioning.

3. To change this pool from regular to thin provisioning, go to the storage pool view of the XIV Management GUI, right-click the Test storage pool, and choose **Resize**, as shown in Figure 4-2.

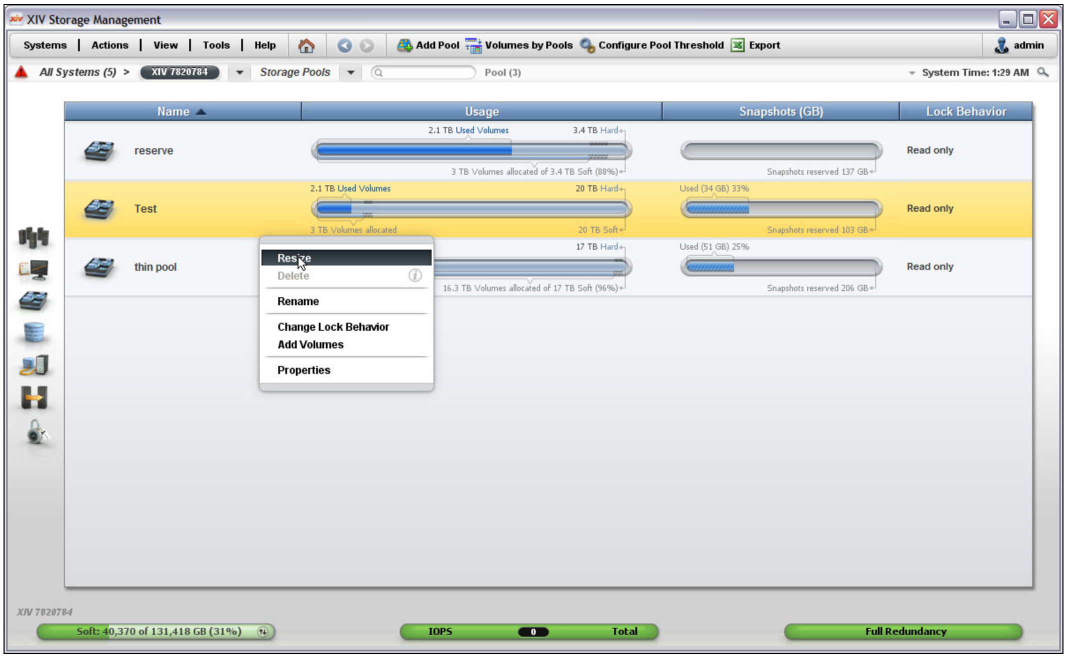


Figure 4-2 Resize storage pool

4. The Resize Pool window opens. Select **Thin Pool**, as shown in Figure 4-3.

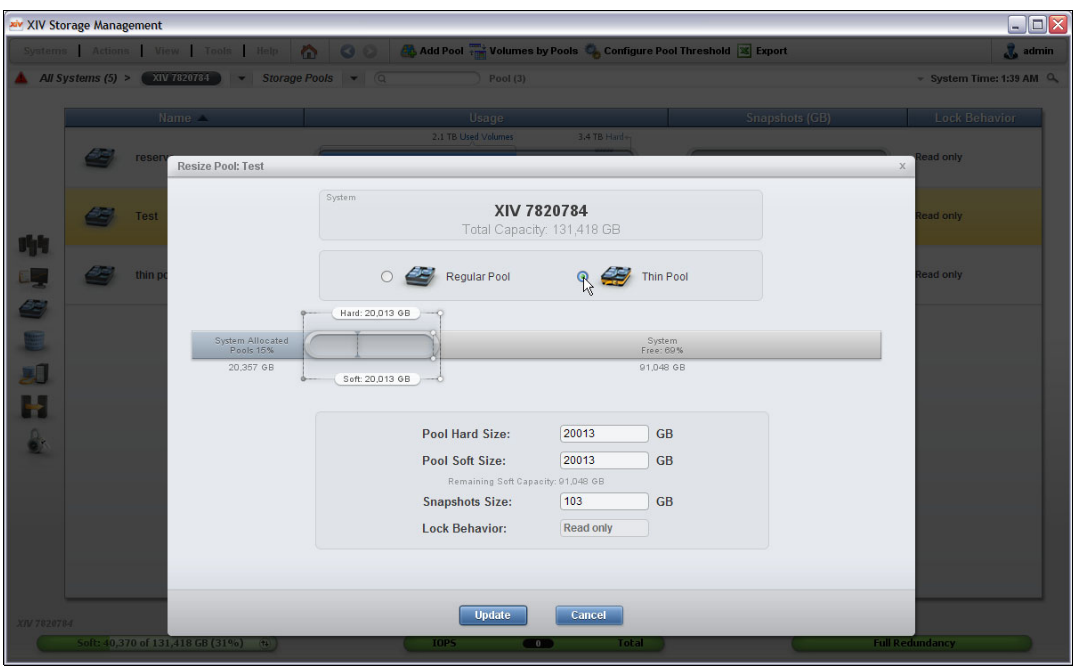


Figure 4-3 Select thin pool

The hard pool size in an XIV system thin pool always needs to contain enough data storage capacity to handle the hard data. At the current data growth rate of less than 1%, adding 20% to the current hard data can provide well over 10 years of growth. This calculation yields a thin pool hard size of 2529 GB.

5. Make the soft pool size 50% larger than the hard size. Add the volumes in the Test pool plus the snapshot space, and the new soft space will be 3803 GB. Also, increase the new snapshot space to 206 GB to implement the previously-mentioned guideline of 10% for snapshot space. Figure 4-4 shows the new Thin Pool specifications for the Test pool.

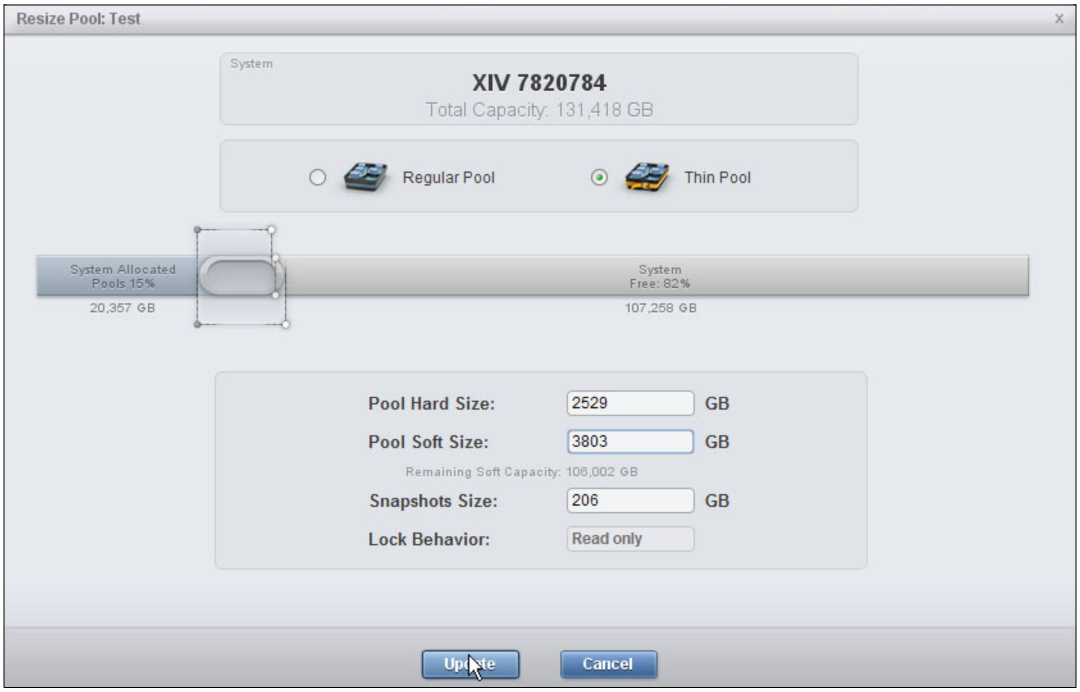


Figure 4-4 Thin pool configuration

When you convert this XIV storage pool to thin provisioning, 16.5 TB is returned to the total system capacity. Figure 4-5 shows the XIV Management GUI view of the new thin pool. Out of the total pool hard space, 2.1 TB of the 2.5 TB is used, which leaves 361 GB of pool hard space free.

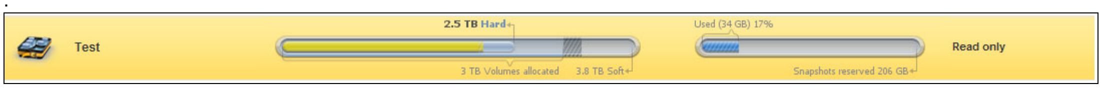


Figure 4-5 New thin pool

The amount of hard capacity used in the Test pool is shown in yellow because the volume usage is greater than 80% of the pool hard size. However, this pool's hard space was carefully sized with a knowledge of growth requirements.

4.2 Thin provisioning conclusions

There are several conclusions about XIV thin provisioning:

- XIV thin provisioning is a mature XIV function that is fundamentally part of the XIV architecture. XIV thin provisioning is easy to implement and use.
- Using thin provisioning allows for a more efficient storage allocation strategy.

- There are real cost savings benefits associated with thin provisioning. These cost savings include acquisition costs, power and cooling costs, and administration costs, among others.
- Because XIV thin provisioning is a logical entity, there is great flexibility involved in using thin provisioning. When the application is running, with no performance or downtime consideration, you can perform the following tasks:
  - Change an XIV storage pool from regular to thin and back again.
  - Change the pool capacity.
  - Move the actual LUN between different storage pools.
- Thin provisioning is not for every storage allocation situation. Initially, use this provisioning capability carefully, because it provides the appearance of more data storage capacity than is actually physically allocated. Thin provisioning requires caution and close monitoring.
- XIV provides sophisticated storage pool monitoring and alerting mechanisms to ensure that a storage pool never runs out of room to store data. Test these mechanisms to ensure that the email and text alerting functions work properly, especially when using thin provisioning.
- There are certain considerations, outside of XIV thin provisioning capabilities, that can help set expectations for the degree to which thin provisioning will effectively provide more efficient data storage capacity usage. These considerations include things like file system types and the way an application reuses data within the file system.

5

Space reclamation with XIV

Thin provisioning provides a volume size as specified, but uses physical storage only when needed. Although this sounds like a great way to use your available storage space efficiently, in real life situations this might not always be true. When files are deleted or moved, the now-unused data is usually still residing on the physical storage. The host system might indicate that the used capacity is now smaller, but the physical storage is still in use and remains allocated to that host system.

Space reclamation deals with this situation by providing a way for the host system to indicate that the physical storage assigned to it is no longer in use. Various host operating systems deal with this process differently, so the commands that you use, and the amount of space actually reclaimed, can vary.

5.1 Space reclamation architecture

After a host writes to a thin-provisioned volume, physical capacity is allocated to the host file system. Unfortunately, if the host deletes the file, only the host file system frees up that space. The physical capacity of the storage system remains unchanged. In other words, the storage system does not free up the capacity from the deleted host file. This is commonly referred to as *dead space*.

Obviously, this is not the most effective method for handling back-end, block-level storage. Ideally, when a host deletes files, not only the host file system but also the back-end storage system reclaim that space.

The current implementations of space reclamation in operating systems (OS) are typically achieved by invoking the Small Computer System Interface (SCSI) **WRITE SAME** or the SCSI **UNMAP** command. Which commands are used depends on the operating system involved.

These SCSI primitive commands have different characteristics and can vary in the amount of capacity that can be reclaimed.

5.1.1 UNMAP command

The T10 Technical Committee established the T10 SCSI Block Command 3 (SBC3) specification that defines the **UNMAP** command for a diverse spectrum of storage devices including hard disk drives (HDDs) and numerous other storage media. Using SCSI **UNMAP**, IT administrators can reclaim host file system space and back-end storage dead space, typically within 30 seconds of a host file deletion. However, not only does SCSI **UNMAP** require T10 SBC3-compliant SCSI hardware, it also requires necessary software application programming interfaces (APIs).

Note: The XIV Storage System Gen3 with Storage Software V11.2 or later is T10 SCSI **UNMAP** command-compliant.

The **UNMAP** command has the following characteristics:

- It deallocates a partition immediately, making the capacity available to other volumes.
- The operation will not increase the physical capacity used.
- The operation is not guaranteed to complete, and it can fail silently. This is dependent on conditions in the storage system involved.

The following operating systems support the **UNMAP** command with XIV storage:

- With XIV Storage System V11.2 or later, XIV Storage System Gen3 provides support for the SCSI **UNMAP** function, natively available with Microsoft Windows Server 2012.

  Note that from an OS perspective, Microsoft Windows uses file TRIM or FILE_LEVEL_TRIM code to trigger storage space reclamation using either **TRIM** or SCSI **UNMAP** requests, depending on the target device types. Consequently for Microsoft Windows solutions, file TRIM and space reclamation are often used interchangeably due to their obvious relationship.
- VMware began supporting SCSI **UNMAP** commands when it introduced the VMware vSphere 5.0 storage APIs for Array Integration (VAAI) primitives. However, VMware discovered issues that affected their Storage vMotion and VM Snapshot consolidation, which led them to alter their SCSI **UNMAP** support for their newest release (vSphere 5.1). Specifically, vSphere 5.1 does not provide proactive or automatic space reclamation for

SCSI **UNMAP** commands. Manual user intervention or scripts must be implemented to realize the SCSI **UNMAP** benefits (preferably outside of peak business hours).

5.1.2 WRITE SAME command

The **WRITE SAME** command with the UNMAP bit=1 can be used to reclaim unused capacity. This command has the following characteristics:

- It writes zeros to the partitions. XIV storage might later mark these partitions as unused as part of the normal scrubbing process.
- It might increase physical capacity usage during the process.
- The operation is guaranteed to complete, and cannot fail silently.

Symantec Storage Foundation employs SCSI **WRITE SAME** commands.

5.1.3 XIV space reclamation considerations

The XIV architecture is by design thinly provisioned. The effectiveness of space reclamation is really dependent on how the OS interacts with the XIV architecture. The smallest unit of capacity that can be freed is a partition (1 MB). Although the partitions might be freed, the change in physical capacity (hard space) might not be noticeable because the XIV GUI displays the allocation in multiples of the minimum XIV allocation size (17 GB).

Note: XIV pool space reclamation occurs after an entire 17 GB allocation is freed, because the minimum XIV allocation size for a LUN is 17 GB.

The use of XIV snapshots will affect the ability to reclaim space. Obviously data preserved by a snapshot cannot be reclaimed, because it is in use by the snapshot. Other considerations include these situations:

- **WRITE SAME** can increase snapshot space when writing zeros to data related to snapshots.
- **UNMAP** might result in snapshots being deleted if additional pool space is needed. Additional space is usually needed due to the possible delay in executing the **UNMAP** of unused data.

  Note: **UNMAP** can result in snapshots being deleted if additional pool space is needed.

- **UNMAP** is supported for volumes that use synchronous mirroring as long as all of the XIV systems are using V11.2 or later code.

  Restriction: **UNMAP** is *not* supported for volumes that use asynchronous mirroring.

5.2 Space reclamation and operating systems

Operating systems with the following characteristics are considered compatible with thin provisioning:

- Localized data placement (unused capacity is not pre-initialized).
- Reuse of freed space (write to previously used and deleted space before writing to never-used space).
- When file system space is deleted, that information is communicated to the storage system so that reclamation can take place. This is implemented with the **UNMAP** or **WRITE SAME** commands described in 5.1, "Space reclamation architecture" on page 32.

The operating systems listed in Table 5-1 currently support space reclamation.

Table 5-1 Operating systems that support space reclamation

Operating System	Command
Symantec	Using **WRITE SAME** (with UNMAP bit=1)
VMware vSphere 4.x	Using **WRITE SAME** (with UNMAP bit=1)
Microsoft 2012	Using the **UNMAP** command
Red Hat Linux	Usually uses **UNMAP** depending on the file system in use
VMware vSphere 5.x	Using **UNMAP** (currently disabled by default)

Restriction: At the time of this writing, the IBM XIV Storage System software does not support Red Hat Linux and VMware vSphere 5.x space reclamation.

5.2.1 Windows 2012 space reclamation

XIV Storage System V11.2 software supports the Windows 2012 **UNMAP** implementation.

Important: Upgrading to XIV Storage System V11.2 software by default enables **UNMAP**, which will start being used immediately.

Considerations for using thin provisioning

Considerations for using thin provisioning with Microsoft Windows 2012 are detailed in the *Plan and Deploy Thin Provisioning* article published by Microsoft at:

http://technet.microsoft.com/en-us/library/jj674351.aspx

The information in this section is taken directly from that article. The following list summarizes those considerations:

- There is predictable storage usage or a low-volatile usage pattern on a storage volume.
- The storage volume can tolerate a brief outage. For example, do not use thin provisioning if a mission-critical cluster disk cannot tolerate any downtime or delays that are caused by thin provisioning. That is, there can be delays that are associated with temporary resource exhaustion or with space reclamation that occurs after large files are deleted.

- There is an adequate storage monitoring process that detects when critical thresholds are crossed. Well-defined policies for monitoring and response are in place.
- You understand the time requirement to acquire new storage. If you do not understand the storage resource procurement process, thin-provisioned logical units (LUNs) might be exposed to the risk of permanent resource exhaustion.

If you do *not* want to use real-time space reclamation (which is enabled by default in Windows 2012), follow these steps:

1. Start Registry Editor.
2. Locate the following registry subkey:

   `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\FileSystem`
3. Double-click **DisableDeleteNotification**.
4. In the Value data box, enter a value of 1, and click **OK**.

Realize that if you configure this setting, this disables the file-delete notification for all LUNs that are assigned to the host.

If file-delete notification is disabled, you can still use the Defragment and Optimize Drives tool to perform space reclamation on-demand, or on a scheduled basis.

Pool thresholds notification

You can use the XIV Storage Management GUI to configure pool usage thresholds to trigger color-coded visual alerts at various severity levels. These alerts are system-based across all pools, and are viewable in the XIV Storage Management GUI by all users configured with the particular system. System events are also produced based on specified values. The default pool usage threshold values are for warning (80%), minor (90%), and major (95%), as shown in Figure 5-1. You can set rules to receive email alerts for these events.

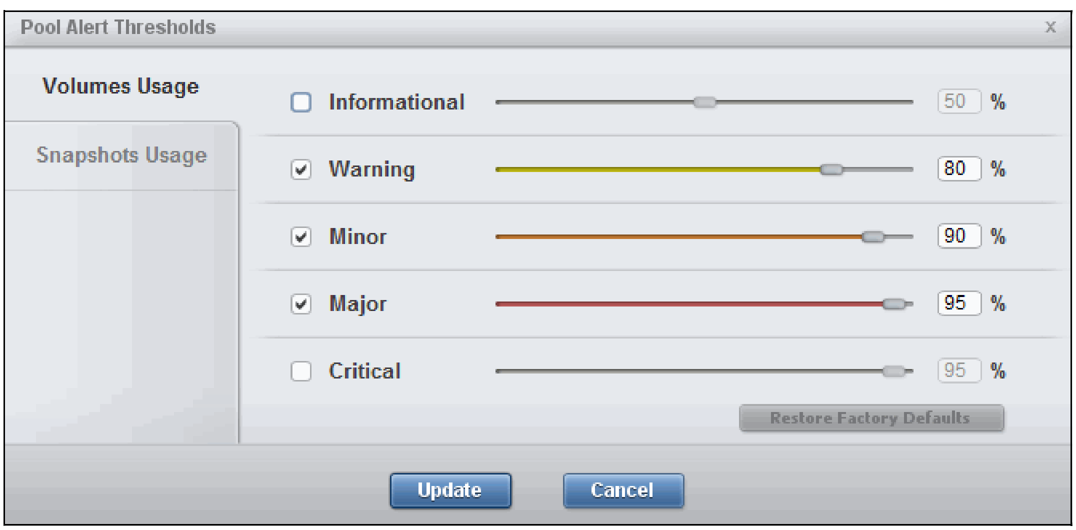


Figure 5-1 Pool thresholds in XIV GUI

Alert thresholds are now also available in Windows Server 2012. When a threshold is exceeded on the IBM XIV Storage System Gen3, the alerts are triggered as Windows System Event logs. There is no need to configure anything for the XIV system.

At the OS level, the Windows System Event log will report the system events shown in Table 5-2.

Table 5-2 Windows events for volume usage

Event ID	Information
144	Threshold notification without additional information
145	Threshold notification without specific information
146	Threshold notification with used LUN capacity and available LUN capacity information
147	Threshold notification with used LUN capacity and available pool capacity information
148	Threshold notification with used pool capacity and available LUN capacity information
149	Threshold notification with used pool capacity and available pool capacity information

If a permanent resource exhaustion threshold is reached, the following also occurs:

- A Windows System Event log ID 150 is generated
- The thin-provisioned XIV volume is taken offline and becomes unavailable until the storage administrator increases the volume capacity and pool capacity

Initial setup

Volumes are created in a thin pool on the XIV system and are defined to a Windows 2012 Server using the XIV Host Attachment Kit. Windows will recognize that these volumes are thin-provisioned drives.

The Optimize Drives menu, which is part of Windows Tools, displays the space efficiency and will analyze and optimize drives as needed, as illustrated in Figure 5-2 on page 37.

Space reclamation automatically occurs as files are deleted or moved, and the analyze option can verify the current state of the thin-provisioned volumes. Note that the space is not necessarily immediately reclaimed.

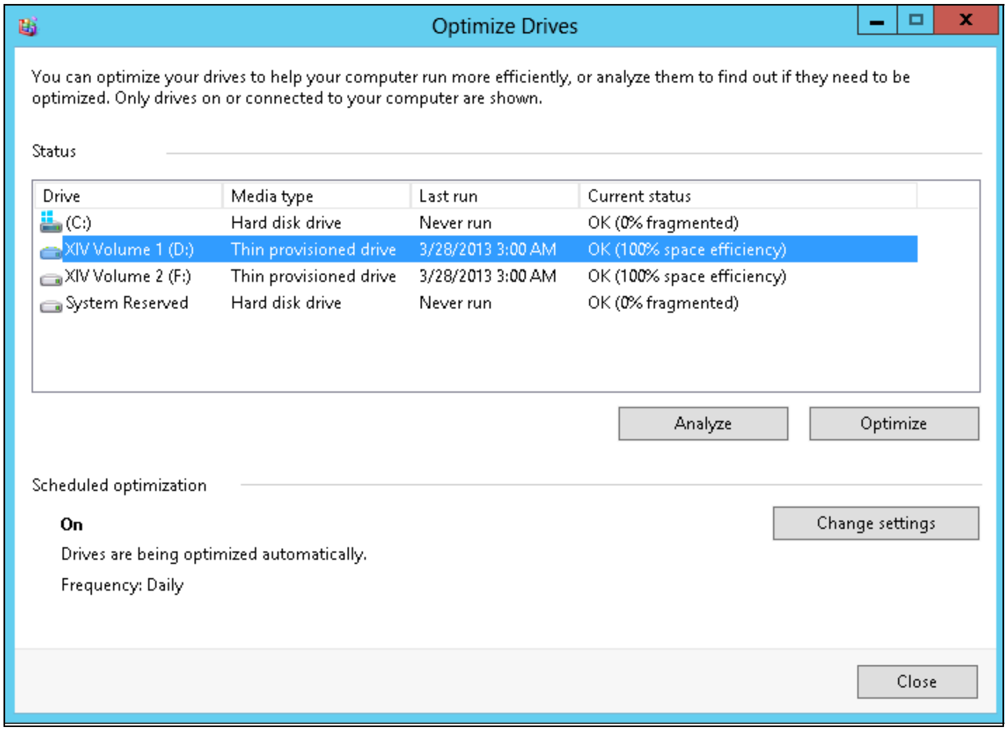


Figure 5-2 Optimize Drives menu

In this scenario, files are created on the drives and use 67 GB of physical space, as shown in Figure 5-3.

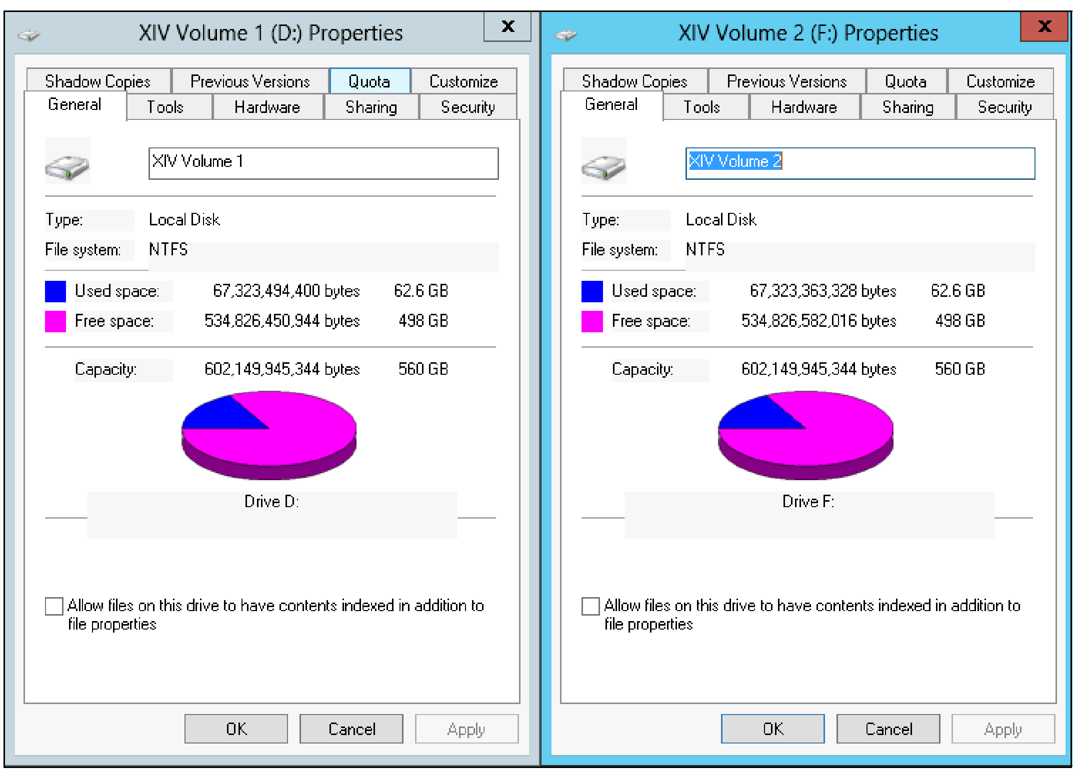


Figure 5-3 Initial volume size

The XIV GUI displays the volume capacity (also showing 67 GB capacity), as shown in Figure 5-4.

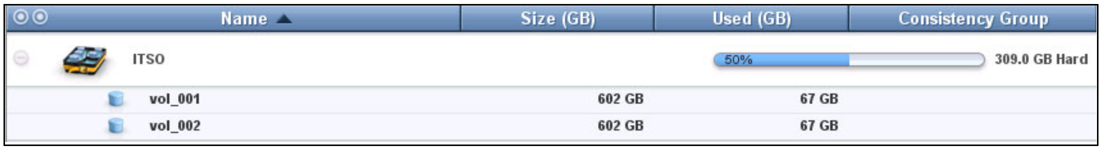


Figure 5-4 Initial volume capacity

Deleting files

When you normally delete files, the actual space allocated on the XIV system does not change. However with space reclamation active, the **UNMAP** command will identify the unused space and the XIV storage system will indicate that the volume allocated size has decreased.

After you delete files on Volume 2, the Windows Properties menu indicates that 33 GB is now allocated, as shown in Figure 5-5.

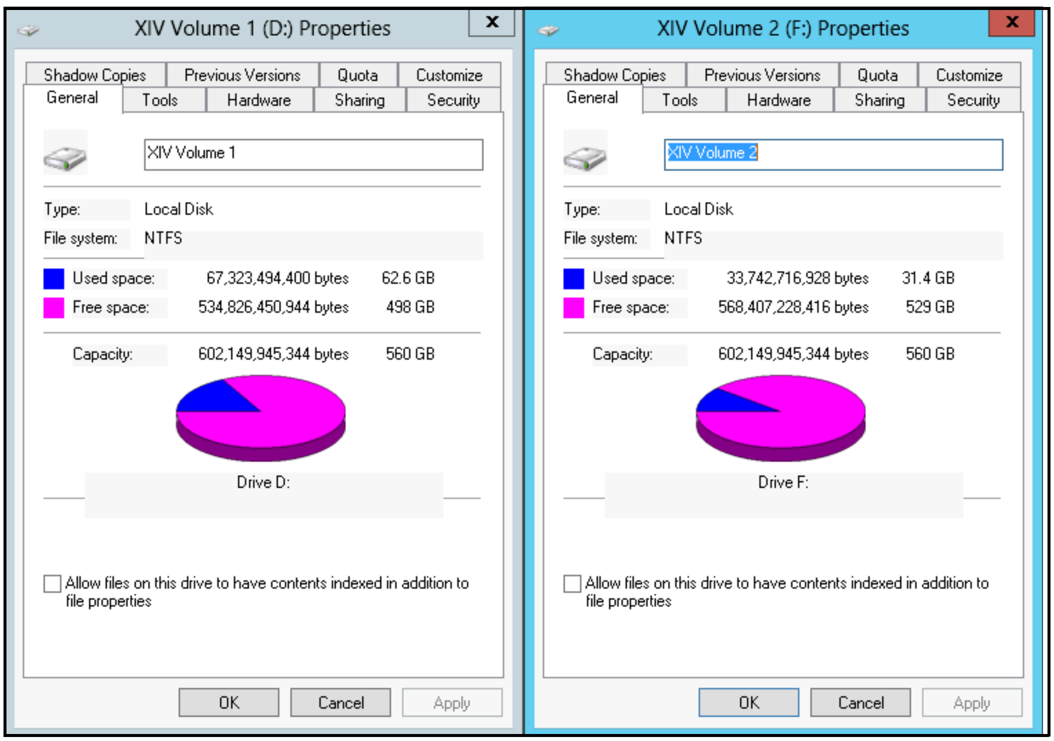


Figure 5-5 Volume size after deleting files

The XIV GUI Volumes by Pools display indicates that Volume 2 hard capacity has also been reduced to 33 GB, as shown in Figure 5-6.

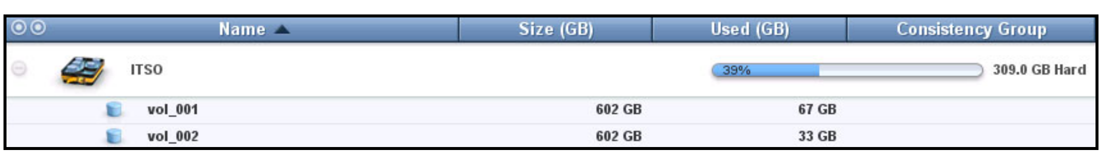


Figure 5-6 Volume capacity after deleting files

Note: The space reduction occurs after the actual file deletion, so there can be a delay of up to 30 seconds before the reduction in hard capacity is seen.

Moving files

Moving data between volumes follows a similar process. When the data is moved, the capacity of the source is reduced and the capacity allocated on the target volume increases.

If you move files from Volume 1 to Volume 2, the Volume Properties panel displays the new allocated size, as shown in Figure 5-7.

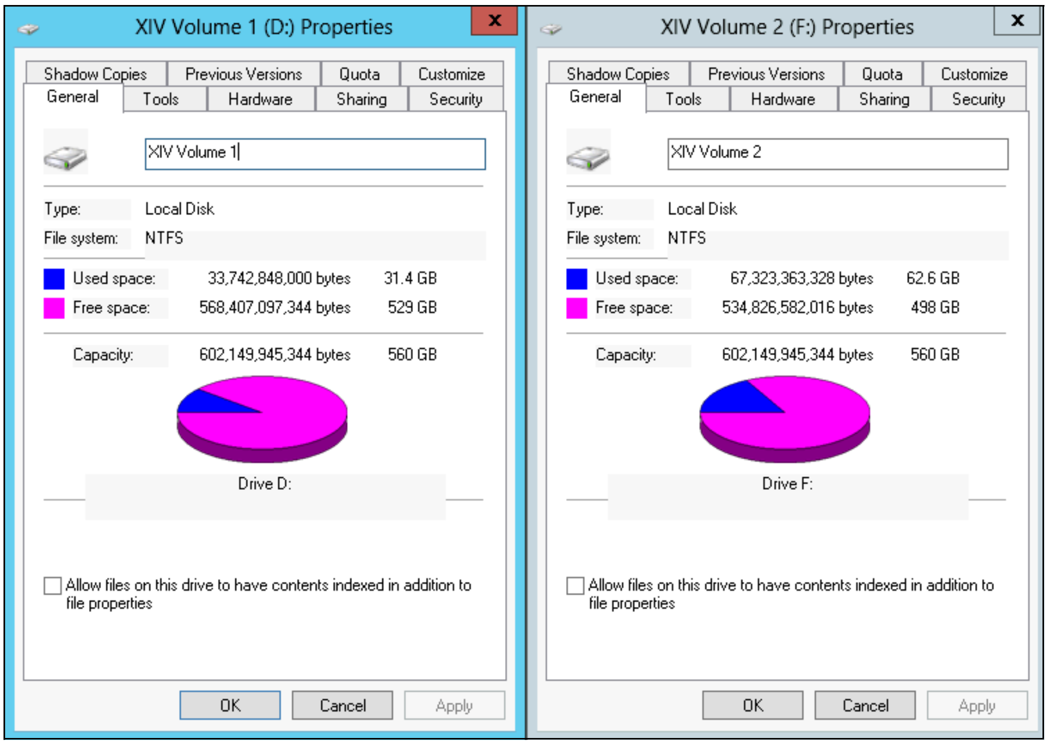


Figure 5-7 Volume size after moving files

The XIV GUI indicates the new volume capacities, as shown in Figure 5-8.

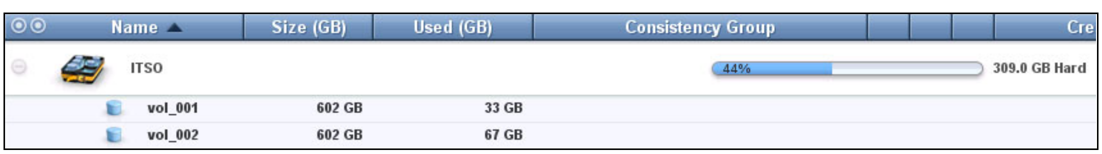


Figure 5-8 Volume capacity after moving files

Snapshot behavior

If you take a snapshot of Volume 2 and then move the files to Volume1, the same reduction in allocated space occurs, as shown in Figure 5-9.

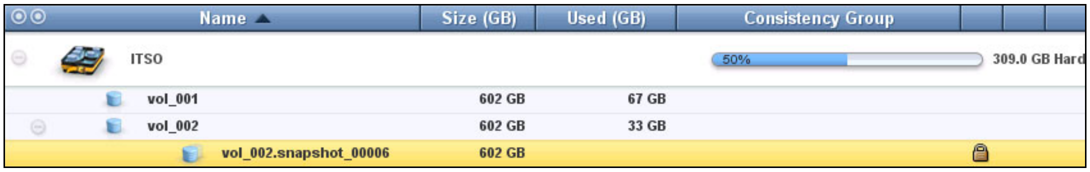


Figure 5-9 Volume capacity allocation

The Storage Pool does not show any Snapshot space allocated, because there have not been any updates to the files, as shown in Figure 5-10.

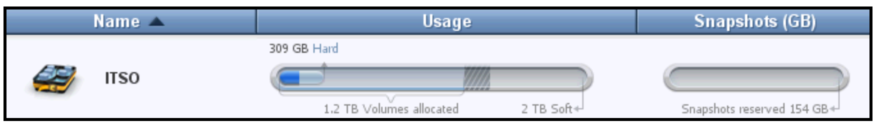


Figure 5-10 Pool capacity allocation

However, after the files are moved, the reclaimed space (34 GB) shows up as capacity allocated to the Snapshot. Therefore, the overall allocated capacity in the Pool has actually increased, as shown in Figure 5-11.

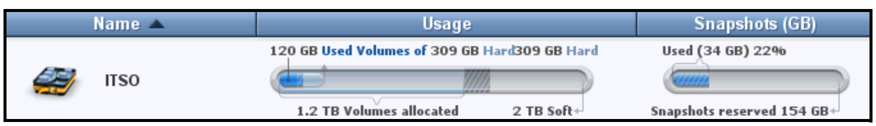


Figure 5-11 Snapshot capacity allocation

Tip: Be careful to allocate sufficient hard capacity in the Storage Pool and Snapshot area. There is a delay between the actual data updates and the space reclamation, which can require additional physical (hard capacity) in the pool. If the Storage Pool becomes full, snapshots might be deleted.

5.2.2 Veritas space reclamation

The Symantec Storage Foundation is available as a unified method of volume management at the OS level. It was formerly known as the Veritas Volume Manager (VxVM) and Veritas Dynamic Multipathing (DMP).

At the time of this writing, the XIV system supports the use of VxVM and DMP with the following operating systems:

- HP-UX
- IBM AIX®
- Red Hat Enterprise Linux
- SUSE Linux
- Linux on Power
- Solaris

Depending on the OS version and hardware, only specific versions and releases of Veritas Volume Manager are supported when connected to the XIV system. In general, IBM software supports VxVM versions 5.0 and 5.1.

For most of the OS and VxVM versions, IBM supports space reclamation on thin-provisioned volumes.

For more information about the operating systems and VxVM versions that are supported, see the System Storage Interoperability Center (SSIC) at:

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

For more information about attaching the IBM XIV Storage System to hosts with VxVM and DMP, see the Symantec website at:

https://sort.symantec.com/asl

The following tasks must be completed to successfully attach the XIV storage system to host systems by using VxVM with DMP:

- Verify Array Support Library (ASL) availability for XIV Storage System on your Symantec Storage Foundation installation and install the XIV Host Attachment Kit
- Place the XIV volumes under VxVM control
- Set up DMP multipathing with IBM XIV storage

Make sure that you install all the patches and updates available for your Symantec Storage Foundation installation. For more information, see your Symantec Storage Foundation documentation.

The following sections explain how you place the XIV volumes under VxVM control, and how to use space reclamation.

Placing the volumes under VxVM control

To place XIV LUNs under VxVM control, complete the following steps:

1. Label the disks with the **format** command.
2. Discover new devices on your hosts by using either the **vxdiskconfig** or **vxdisk -f scandisks** command.
3. Check for new devices that were discovered by using the **vxdisk list** command, as illustrated in Example 5-1.

   Example 5-1 Discovering and checking new disks on your host

```
# vxdisk -f scandisks
# vxdisk list
DEVICE       TYPE            DISK         GROUP        STATUS
disk_0       auto:ZFS        -            -            ZFS
disk_1       auto:ZFS        -            -            ZFS
disk_2       auto:ZFS        -            -            ZFS
disk_3       auto:ZFS        -            -            ZFS
xiv0_251a    auto:none       -            -            online invalid
xiv0_251b    auto:none       -            -            online invalid
```

4. After you discover the new disks on the host, you might need to format the disks. For more information, see your OS-specific Symantec Storage Foundation documentation. In this example, the disks must be formatted. Run the **vxdiskadm** command as shown in

Example 5-2. Select option `1` and then follow the instructions, accepting all defaults except for these questions:

a. `Encapsulate this device?` (answer `no`)

b. `Instead of encapsulating, initialize?` (answer `yes`).

Example 5-2 Configuring disks for VxVM

```
# vxdiskadm

Volume Manager Support Operations
Menu: VolumeManager/Disk

 1      Add or initialize one or more disks
 2      Encapsulate one or more disks
 3      Remove a disk
 4      Remove a disk for replacement
 5      Replace a failed or removed disk
 6      Mirror volumes on a disk
 7      Move volumes from a disk
 8      Enable access to (import) a disk group
 9      Remove access to (deport) a disk group
 10     Enable (online) a disk device
 11     Disable (offline) a disk device
 12     Mark a disk as a spare for a disk group
 13     Turn off the spare flag on a disk
 14     Unrelocate subdisks back to a disk
 15     Exclude a disk from hot-relocation use
 16     Make a disk available for hot-relocation use
 17     Prevent multipathing/Suppress devices from VxVM's view
 18     Allow multipathing/Unsuppress devices from VxVM's view
 19     List currently suppressed/non-multipathed devices
 20     Change the disk naming scheme
 21     Get the newly connected/zoned disks in VxVM view
 22     Change/Display the default disk layouts
 list   List disk information


 ?      Display help about menu
 ??     Display help about the menuing system
 q      Exit from menus

Select an operation to perform: 1

Add or initialize disks
Menu: VolumeManager/Disk/AddDisks
  Use this operation to add one or more disks to a disk group.  You can
  add the selected disks to an existing disk group or to a new disk group
  that will be created as a part of the operation. The selected disks may
  also be added to a disk group as spares. Or they may be added as
  nohotuses to be excluded from hot-relocation use. The selected
  disks may also be initialized without adding them to a disk group
  leaving the disks available for use as replacement disks.

  More than one disk or pattern may be entered at the prompt.  Here are
  some disk selection examples:
```

```
  all:          all disks
  c3 c4t2:      all disks on both controller 3 and controller 4, target 2
  c3t4d2:       a single disk (in the c#t#d# naming scheme)
  xyz_0 :       a single disk (in the enclosure based naming scheme)
  xyz_ :        all disks on the enclosure whose name is xyz

Select disk devices to add: [<pattern-list>,all,list,q,?] xiv0_251a xiv0_251b
  Here are the disks selected.  Output format: [Device_Name]

  xiv0_251a xiv0_251b

Continue operation? [y,n,q,?]  (default: y)
  You can choose to add these disks to an existing disk group, a
  new disk group, or you can leave these disks available for use
  by future add or replacement operations.  To create a new disk
  group, select a disk group name that does not yet exist.  To
  leave the disks available for future use, specify a disk group
  name of "none".

Which disk group [<group>,none,list,q,?]  (default: none) XIV_DG

Create a new group named XIV_DG? [y,n,q,?]  (default: y)

Create the disk group as a CDS disk group? [y,n,q,?]  (default: y)

Use default disk names for these disks? [y,n,q,?]  (default: y)

Add disks as spare disks for XIV_DG? [y,n,q,?]  (default: n)

Exclude disks from hot-relocation use? [y,n,q,?]  (default: n)

Add site tag to disks? [y,n,q,?]  (default: n)
  A new disk group will be created named XIV_DG and the selected disks
  will be added to the disk group with default disk names.

  xiv0_251a xiv0_251b

Continue with operation? [y,n,q,?]  (default: y)
  The following disk devices have a valid VTOC, but do not appear to have
  been initialized for the Volume Manager.  If there is data on the disks
  that should NOT be destroyed you should encapsulate the existing disk
  partitions as volumes instead of adding the disks as new disks.
  Output format: [Device_Name]
xiv0_251a xiv0_251b

Encapsulate these devices? [Y,N,S(elect),q,?]  (default: Y) N
xiv0_251a xiv0_251b

Instead of encapsulating, initialize?
[Y,N,S(elect),q,?]  (default: N) Y

Do you want to use the default layout for all disks being initialized?
[y,n,q,?]  (default: y)
  Initializing device xiv0_251a.
```

```
  Initializing device xiv0_251b.
  VxVM  NOTICE V-5-2-120
Creating a new disk group named XIV_DG containing the disk
  device xiv0_251a with the name XIV_DG01.
  VxVM  NOTICE V-5-2-88
Adding disk device xiv0_251b to disk group XIV_DG with disk
  name XIV_DG02.
Add or initialize other disks? [y,n,q,?]  (default: n)
```

> **Tip:** If the **vxdiskadm** initialization function warns that the disk is offline, you might need to initialize it using the default OS-specific utility. For example, use the **format** command in Solaris.

5. Check the results by using the **vxdisk list** command, as shown in Example 5-3.

Example 5-3 Showing the results of putting XI V LUNs under VxVM control

```
# vxdisk list
DEVICE       TYPE           DISK        GROUP       STATUS
disk_0       auto:ZFS       -           -           ZFS
disk_1       auto:ZFS       -           -           ZFS
disk_2       auto:ZFS       -           -           ZFS
disk_3       auto:ZFS       -           -           ZFS
xiv0_251a    auto:cdsdisk   XIV_DG01    XIV_DG      online thinrclm
xiv0_251b    auto:cdsdisk   XIV_DG02    XIV_DG      online thinrclm
```

The XIV LUNs that were added are now available for volume creation and data storage. The status `thinrclm` means that the volumes from the XIV system are thin-provisioned, and the XIV storage has the Veritas thin reclamation API implemented.

Using space reclamation

The **vxdisk reclaim <diskgroup> | <disk>** command to free up any space that can be reclaimed.

- In UNIX, you can reclain space as follows:
  - Reclaim by disk group, enclosure, or LUN, using these commands as appropriate:

    ```
    # vxdisk reclaim <dg>
    # vxdisk reclaim <enclosure>
    # vxdisk reclaim <da-name>
    ```

    By default, the reclamation does not affect unmarked space, which is the unused space between subdisks. If a LUN has a lot of physical space that was previously allocated, the space between the subdisks might be substantial. Use the `-o full` option to reclaim the unmarked space, as follows:

    ```
    # vxdisk reclaim –o full <dg>/<enclosure>/<da-name>
    ```

  - Reclaim by file system:

    ```
    # /opt/VRTS/bin/fsadm –R <mount point>
    # /opt/VRTS/bin/fsadm –R –A <mount point>
    # /opt/VRTS/bin/fsadm –R –o analyze <mount point>
    # /opt/VRTS/bin/fsadm –R –o auto <mount point>
    ```

- In Windows, you can reclaim space as follows:
  - Reclaim by volume, free space, or both (all):

    ```
    C:\vxdg -g <DiskGroup> reclaim option=volumes|freespace|all
    ```

  - Reclaim by disk:

    ```
    C:\vxdisk -g <DiskGroup> reclaim <DiskName>
    ```

> **Note:** Thin-reclaim operations are not supported on RAID-5 VxVM volumes on UNIX and Windows.

If your administrator wants to inhibit thin-reclaim operations, this can be achieved by populating a file (`/etc/vxdefault/vxdisk`) with a value **`reclaim=off`**.

5.2.3 Space reclamation with VMware

VMware began supporting SCSI **`UNMAP`** commands when it introduced the VMware vSphere 5.0 storage APIs for Array Integration (VAAI) primitives. However, VMware discovered issues which affected their Storage vMotion and VM Snapshot consolidation that led them to alter their SCSI **`UNMAP`** support for its newest release (vSphere 5.1). Specifically, vSphere 5.1 does not provide proactive or automatic space reclamation for SCSI **`UNMAP`** commands. Manual user intervention or scripts must be implemented to realize the SCSI **`UNMAP`** benefits (preferably outside of peak business hours). However, because it is enabled by default in V11.2, running VMware 5.0 means that space reclamation is enabled automatically.

Related publications

The publications listed in this section are considered particularly suitable for providing more detailed information about the topics covered in this paper.

IBM Redbooks

The following IBM Redbooks publications provide additional information about the topic in this document. Note that some publications referenced in this list might be available in softcopy only.

- *IBM XIV Storage System: Copy Services and Migration*, SG24-7759
- *IBM XIV Storage System: Host Attachment and Interoperability*, SG24-7904
- *Solid-State Drive Caching in the IBM XIV Storage System*, REDP-4842
- *Using the IBM XIV Storage System in OpenStack Cloud Environments*, REDP-4971
- *XIV Storage System in a VMware Environment*, REDP-4965

You can search for, view, download, or order these documents and other Redbooks, Redpapers, web docs, drafts, and additional materials at the following website:

http://**ibm.com**/redbooks

Other publications

These publications are also relevant as further information sources:

- *IBM XIV Remote Support Proxy Installation and User's Guide*, GA32-0795
- *IBM XIV Storage System Application Programming Interface,* GC27-3916
- *IBM XIV Storage System Planning Guide,* GC27-3913
- *IBM XIV Storage System Product Overview*, GC27-3912
- *IBM XIV Storage System Commands Reference*, GC27-3914
- *IBM XIV Storage System Management Tools version 4.0 User Guide* SC27-4230
- *IBM XIV Storage System XCLI Utility User Manual*, GC27-3915
- *The iSCSI User Guide*, found at:

  http://download.microsoft.com/download/a/e/9/ae91dea1-66d9-417c-ade4-92d824b871af/uguide.doc

Online resources

These websites are also relevant as further information sources:

- IBM XIV Storage System Information Center:

  http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp
- IBM XIV Storage website:

  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

Help from IBM

IBM Support and downloads:

http://**ibm.com**/support

IBM Global Services:

http://**ibm.com**/services

IBM XIV Storage System Thin Provisioning and Space Reclamation

Redpaper™

Thin provisioning explored

File system considerations

Space reclamation examples

Thin provisioning is the practice of passing logical unit number (LUN) sizes up to application servers without actually reserving the total physical capacity of those LUNs for data storage. Thin provisioning is a popular feature of IBM XIV Storage System.

Data *space reclamation* helps you enjoy the benefits of thin provisioning. Space reclamation is a storage system function to reclaim a specific amount of disk space for general-purpose use after being notified by the file system that the disk space was deleted at the host level.

Because thin provisioning and support for space reclamation are so tightly related in the XIV storage system, this IBM Redpaper publication explores both concepts in detail.

This publication is intended for system and storage administrators who want to take advantage of the XIV storage system functionality in thin-provisioned environments, coupled with the latest space reclamation enhancements.

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

REDP-5001-00