Introduction Guide to the IBM Elastic Storage Server

Dino Quintero
Puneet Chaudhary
Brian Herr
Steven Normann
Marcelo Ramos
Richard Rosenthal
Robert Simon
Marian Tomescu
Richard Wale

Cloud

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

Notices ................................................................. v
Trademarks ......................................................... vi

IBM Redbooks promotions .......................................... vii

Preface ................................................................ ix
Authors ................................................................. ix
Now you can become a published author, too! ................... x
Comments welcome ................................................ xi
Stay connected to IBM Redbooks ................................ xi

Chapter 1. Introduction to the IBM Elastic Storage Server .................................... 1
  1.1 Elastic Storage Server ......................................... 2
  1.2 Software RAID .................................................. 2
    1.2.1 RAID codes ............................................... 2
    1.2.2 End-to-end checksum .................................. 3
    1.2.3 Declustered RAID .................................... 3
  1.3 Building blocks .................................................. 5
  1.4 Value added ..................................................... 5

Chapter 2. Architecture .................................................... 7
  2.1 Hardware components ....................................... 8
  2.2 Software components ....................................... 13
    2.2.1 Operating system ..................................... 14
    2.2.2 Extreme Cluster/Cloud Administration Toolkit (xCAT) .. 14
    2.2.3 IBM Spectrum Scale .................................. 14
    2.2.4 IBM Spectrum Scale RAID .......................... 14

Chapter 3. Planning and integration ......................................... 21
  3.1 Elastic Storage Server overview .................................. 22
  3.2 Elastic Storage Server installation and upgrading .................. 22
  3.3 Elastic Storage Server networks .................................. 22
  3.4 Elastic Storage Server: Storage parameters......................... 25
    3.4.1 Recovery group server parameters ..................... 25
    3.4.2 pdisks ................................................... 26
    3.4.3 Declustered arrays ................................... 31
    3.4.4 Typical configurations ................................ 35
  3.5 Best-practice recommendations for Spectrum Scale RAID ............ 35
  3.6 Elastic Storage Server file system configurations .................. 37
  3.7 Elastic Storage Server ordering .................................. 42
  3.8 Elastic Storage Server client integration ............................ 44
  3.9 Monitoring ................................................... 44

Related publications .................................................. 47
  IBM Redbook ...................................................... 47
  Online resources .................................................. 47
  Help from IBM .................................................... 47

© Copyright IBM Corp. 2016. All rights reserved.
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

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

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

- Redbooks (logo) ®
- AIX®
- AIX/ESA®
- GPFS™
- IBM®
- IBM Spectrum Scale™
- Power Systems™
- POWER7®
- POWER8®
- PowerHA®
- PowerVM®
- Redbooks®
- Redpaper™

The following terms are trademarks of other companies:

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

Other company, product, or service names may be trademarks or service marks of others.
Find and read thousands of IBM Redbooks publications

- Search, bookmark, save and organize favorites
- Get up-to-the-minute Redbooks news and announcements
- Link to the latest Redbooks blogs and videos

Get the latest version of the Redbooks Mobile App

Promote your business in an IBM Redbooks publication

Place a Sponsorship Promotion in an IBM® Redbooks® publication, featuring your business or solution with a link to your web site.

Qualified IBM Business Partners may place a full page promotion in the most popular Redbooks publications. Imagine the power of being seen by users who download millions of Redbooks publications each year!
Preface

This IBM® Redpaper™ publication introduces and describes the IBM Elastic Storage Server as a scalable, high-performance data and file management solution, which is built on proven IBM Spectrum Scale™ technology, formerly IBM General Parallel File System (GPFS™). Providing reliability, performance, and scalability, IBM Elastic Storage Servers can be implemented for a range of diverse requirements.

This publication helps you to understand the solution and its architecture and helps you to plan the installation and integration of the environment, which is created from the following combination of physical and logical components:

- Hardware
- Operating system
- Storage
- Network
- Applications

Knowledge of the Elastic Storage Server components is key for planning an environment.

This paper is targeted toward technical professionals (consultants, technical support staff, IT Architects, and IT Specialists) who are responsible for delivering cost effective cloud services and big data solutions. The content of this paper can help you to uncover insights among client's data so that you can take appropriate actions to optimize business results, product development, and scientific discoveries.

Authors

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

Dino Quintero is a Complex Solutions Project Leader and an IBM Level 3 Certified Senior IT Specialist with the ITSO in Poughkeepsie, New York. His areas of knowledge include enterprise continuous availability, enterprise systems management, system virtualization, technical computing, and clustering solutions. He is an Open Group Distinguished IT Specialist. Dino holds a Master of Computing Information Systems degree and a Bachelor of Science degree in Computer Science from Marist College.

Puneet Chaudhary is a Technical Solutions Architect working with the IBM Elastic Storage Server and IBM Spectrum Scale solutions. He has worked with IBM GPFS, now Spectrum Scale, for many years.

Brian Herr is a Software Engineer for IBM Spectrum Scale. He has been with IBM since 1983 working mostly in the area of High Performance Computing. He has been part of the IBM Spectrum Scale Redundant Array of Independent Disks (RAID) development team since 2008.

Steven Normann is a Senior Software Engineer in MTS working in Poughkeepsie, New York. He has worked with IBM since 1984. He is a Team Leader for IBM Spectrum Scale in the Software Technical Support Group, which supports High Performance Clustering software.
Marcelo Ramos is a Senior Software Support Specialist at IBM Brazil. He has been working with IBM AIX® and related products since 1998. His areas of expertise include implementation and advanced support for AIX, IBM PowerVM®, IBM PowerHA®, and Spectrum Scale. He also has SAN and storage-related skills.

Richard Rosenthal is a Senior Software Engineer for IBM Spectrum Scale. He has been with IBM since 1979, and in development since the mid-1990s, working mostly in the area of High Performance Computing. He has been part of the IBM Spectrum Scale RAID development team since 2008.

Robert Simon is a Senior Software Engineer for IBM Spectrum Scale. He has been with IBM since 1987 working in Software Support (Level 2) for VM, IBM AIX/ESA®, PSSP (HPC), Spectrum Scale, and Spectrum Scale RAID.

Marian Tomescu has 16 years of experience as an IT Specialist and currently works for IBM Global Technologies Services in Romania and has 10 years of experience in IBM Power Systems™. Marian has a Master’s degree in Electronics Images, Shapes, and Artificial Intelligence from Polytechnic University- Bucharest, Electronics and Telecommunications, in Romania.

Richard Wale is a Senior IT Specialist working at the IBM Hursley Lab, UK. He holds a B.Sc. (Hons) degree in Computer Science from Portsmouth University, England. He has over 14 years of experience supporting AIX. His areas of expertise include IBM Power Systems, PowerVM, AIX, and IBM i.

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

Rich Conway
International Technical Support Organization, Poughkeepsie Center

Wesley Jones
IBM Poughkeepsie, NY

Luis Bolinches
IBM Finland

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: 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:
  ibm.com/redbooks

» Send your comments in an email to:
  redbooks@us.ibm.com

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

Stay connected to IBM Redbooks

» Find us on Facebook:
  http://www.facebook.com/IBMRedbooks

» Follow us on Twitter:
  https://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:

» Stay current on recent Redbooks publications with RSS Feeds:
  http://www.redbooks.ibm.com/rss.html
Introduction to the IBM Elastic Storage Server

This chapter introduces the IBM Elastic Storage Server solution, its characteristics, and where it fits in the business environments.

This chapter also describes some of the software and hardware characteristics of the Elastic Storage Server, the software Redundant Array of Independent Disks (RAID), and the building block concepts of the solution.

The following sections are presented in this chapter:
- Elastic Storage Server
- Software RAID
- Building blocks
- Value added
1.1 Elastic Storage Server

The IBM Elastic Storage Server is software-defined storage that combines IBM Spectrum Scale (formerly GPFS) software, which provides the clustered file systems, and the CPU and I/O capability of the IBM POWER® architecture.

The building block-based solution of the Elastic Storage Server delivers high performance, high available, and scalable IBM Spectrum Scale functionalities to today's high performance and business analytics clusters.

Instead of hardware-based disk RAID controllers, the IBM declustered RAID technology based on the IBM Spectrum Scale RAID software maintain all RAID capabilities allowing superior data protection and rebuild times reduced to a fraction of the time needed with hardware-based RAID controllers.

IBM Spectrum Scale is required to access data on the Elastic Storage Server common repository infrastructure. Although, by using new capabilities that are available on IBM Spectrum Scale version 4.1.1, it is also possible to use protocol nodes to allow object access data using OpenStack Swift or file access through Server Message Block (SMB) or Network File System (NFS).

1.2 Software RAID

The IBM Spectrum Scale RAID software used in the Elastic Storage Server solution runs on standard serial-attached SCSI (SAS) disks in just a bunch of disks (JBOD) arrays, which allows cost reduction, with the option of solid-state drives (SSDs) when more performance is needed. The solution does not require or use any kind of external RAID controller or acceleration.

IBM Spectrum Scale RAID supports multiple RAID codes and distributes client data, redundancy information, and spare space across the disks in such a way that if there is a physical disk loss, or even a group of physical disk losses, does not affect data availability.

Also, instead of relying only on the disks to detect and report faults, read or write errors, and other integrity problems, Spectrum Scale RAID implements an end-to-end checksum.

1.2.1 RAID codes

Spectrum Scale RAID in the Elastic Storage Server supports different data protection algorithms and is capable of detecting and correcting up to two or three concurrent faults.

The options for RAID configuration are eight stripes of data plus two or three parity stripes using Reed-Solomon codes or one stripe of data plus two or three replica stripes. The data plus parity or replica stripes, which are called tracks, are illustrated in Figure 1-1 on page 3.
1.2.2 End-to-end checksum

The IBM Spectrum Scale software on the client used to access data on the Elastic Storage Server knows that the Spectrum Scale file system is based on Spectrum Scale RAID Network Shared Disks, and during a write operation an 8-bytes checksum is calculated, appended to the data, and sent over the network to the Spectrum Scale RAID server. The checksum is verified and then Spectrum Scale RAID writes the data along with its checksum on the disks and logs the version number on its metadata.

When a read operation is requested, Spectrum Scale RAID verifies checksum and version on its metadata. If it is OK, it sends the data to the client. If it is not OK, the data is rebuilt based on parity or replication and then sent to the client along with newly generated checksum.

The end-to-end checksum feature allows to prevent and correct silent disk errors or missing disk writes.

1.2.3 Declustered RAID

Spectrum Scale RAID implements its own data and spare disk layout scheme that reduces the overhead to clients when recovering from disk failures. To accomplish this, instead of leaving all spare space in a single disk, it spreads or declusters user data, redundancy information, and spare space across all the disks of the array. Figure 1-2 on page 4 compares conventional 1+1 RAID layout with a declustered array.

For example purposes, consider seven stripes of data on each disk. Figure 1-2 on page 4 shows the left three arrays of two disks in a replicated 1+1 configuration and a spare. On the left you can see the data stripes spreading all over the seven disks of the declustered array.
In Figure 1-2, note that on the Elastic Storage Server from v3.5 and even on GL6 models, only one declustered array is used. Figure 1-2 shows 348 HDD, which is a model reduced for simplification.

In case of failure of one disk, on the traditional 1+1, all data from the remaining disks of the array must be replicated to the spare disk. On the declustered array, the replication occurs on spare space of all the remaining disks, which can decrease the rebuild impact from three to four times. Figure 1-3 shows a model reduced for simplification.

On the Elastic Storage Server using RAID 8+2 or 2-way replication (1+2), if there is one disk loss, the rebuild operation starts with low priority with even lower impact for the clients. With this array configuration, only if there are two concurrent disk losses, the rebuild is considered critical and run on high priority. Using 8+3 RAID or 3-way replication (1+3), the rebuild operation only becomes critical in the case of three concurrent disk losses in the same declustered array.
1.3 Building blocks

The *building block* is the minimum configuration of an Elastic Storage Server and it is also the unit of expansion of the solution. If more space is needed, a new building block is added to the solution.

Each building block consists of two IBM POWER8 servers and a specific number of storage enclosures (1, 2, 4, or 6) depending on the model.

The POWER8 servers are model S822L and the storage enclosures can be DCS3700 (4U 60 drives) units for models GL2, GL4, and GL6 or EXP24S SFF Gen2-bay drawer (2U 24 drives) for models GS1, GS2, GS4, or GS6.

The GS models can use 2.5” 10 K rpm HDD (1.2 TB) or 2.5” SSD (400 GB or 800 GB). GL models can use 3.5” NL-SAS HDDs (2 TB, 4 TB, or 6 TB).

Models GL and GS can be mixed to achieve specific needs for client applications using the storage. For example, SSDs for metadata and NL-SAS for data storage.

1.4 Value added

Why Elastic Storage Server instead of any other hardware-controlled solution or in-house built solution?

IBM Spectrum Scale RAID and support from IBM for the solution contribute to the added value including a tune solution from the IBM services team.

Elastic Storage Server offers scalability from 40 TB to hundreds of petabytes, supporting 10-Gigabit Ethernet (10 GbE), 40-Gigabit Ethernet (40 GbE), and FDR InfiniBand.

Big data requires easy storage growth. The IBM Elastic Storage Server building block approach, where adding more storage servers adds to the overall capacity, bandwidth, and performance all with a single name space.

By leveraging JBODs and IBM Power Systems servers, the Elastic Storage Server provides price/performance storage for analytics, technical computing, and cloud computing environments.

The Elastic Storage Server modern declustered RAID technology is designed to recover from multiple disk failures in minutes, versus hours and days in older technology, giving the solution predictable performance and data protection. 8+2 and 8+3 RAID protection and platter-to-client data protection are included with the Elastic Storage Server.

IBM engineering and testing teams worked on designing the solution in a way that, from adapter placement on the servers, number of adapters, number of disks on the drawers, cabling, number of drawers to the software versions, and the way data is placed on disks, every part of the solution is meant to provide greater performance and data availability.

The complete solution is tested after manufactured, and tools and scripts are used by the services team during the delivery to ensure every piece of the solution integrates all together.

For more information about the Elastic Storage Server, refer to the following websites:

http://www.ibm.com/systems/storage/spectrum/ess
http://ibm.co/1kAzc3p
Architecture

This chapter covers the hardware of the IBM Elastic Storage Server including:

- I/O nodes
- Storage enclosures
- Hardware Management Console (HMC)
- Elastic Management Server (EMS)

**Note:** The Elastic Storage Server is like an appliance. Thus, the user should not install different kernel levels or drivers as added in the system. Moreover, the user should not run any application or workload on the I/O nodes or EMS.

Additional topics discussed in this chapter include the different building block models and software components and features, such as:

- Operating system
- Extreme Cluster/Cloud Administration Toolkit (xCAT)
- IBM Spectrum Scale
- IBM Spectrum Scale RAID
2.1 Hardware components

The Elastic Storage Server is a rack-mounted solution and the basic building block contains:

- 2 x I/O servers, which are IBM POWER8 servers model S822L (8247-22L) with 2x10 Core 3.4 GHz, 128 GB CDIMM memory is the default min, expandable to 256 GB
- 1, 2, 4, or 6 storage enclosures

Note: With the first release of the Elastic Storage Server generation 1 solution, the first building block came with an HMC for solely managing the Power Systems part of the Elastic Storage Server environment. Subsequent releases expanded on what is supported. IBM now allows the provided HMC to manage other IBM Power Systems servers outside of the Elastic Storage Server environment. Furthermore, customers can opt to use an existing HMC within their environment to manage a new Elastic Storage Server solution.

The ESS building block consist of a pair of two Power 822 servers (which are also known as gssIO server or head nodes), and at least one storage enclosure. In addition, SAS, NL-SAS and SSDs disk types are available and independent of various disk enclosure types. Different disk sizes also are available.

Although the following are not part of the basic building blocks, these are part of the solution and can be provided by the customer because each Elastic Storage Server requires one of each:

- HMC, which is used to manage the POWER8 servers and provide reliability, availability, and serviceability (RAS) features.
- EMS, which is another IBM POWER8 server (model S821L) used to manage and deploy the I/O nodes and also to run the software used by the graphical user interface (GUI).

Note: Specific versions of the HMC software are required for managing POWER8 servers. If the HMC is already managing IBM POWER7® servers or earlier servers, these servers might require firmware upgrades before a new HMC version is installed, and additional planning and maintenance on production servers might be necessary.

For additional information, check the POWER code matrix website:
http://www.ibm.com/support/customercare/sas/f/power5cm/home.html

Besides the high-speed network used to access the data using the General Parallel File System (GPFS) protocol, other dedicated networks are required by the Elastic Storage Server solution.

If IBM-ordered, a network switch with specific configuration with separated VLANs is provided for the HMC management network and for the environmental service module (ESM) management. Otherwise, the customer needs to provide two separated networks (dedicated switches or VLANs): One for the HMC and one for the EMS. Figure 2-1 on page 9 shows typical network configurations for the Elastic Storage Server solution.
Chapter 2. Architecture

Figure 2-1   Elastic Storage Server networks

An Elastic Storage Server network includes the following components:

► HMC Service Network: The HMC runs Dynamic Host Configuration Protocol (DHCP) on this isolated network or VLAN to provide IP addresses to the Flexible Service Processor (FSP) on the POWER8 servers.

► Management or Provisioning Network: The EMS runs DHCP on this isolated network or VLAN to provide IP addresses to the I/O servers. This network is used by the EMS to communicate with the HMC to manage the I/O servers and to deploy the operating system and Elastic Storage Server software to the I/O servers.

► Clustering Network: This is a high-speed network based on InfiniBand (IB), or 10-Gigabit Ethernet, or 40-Gigabit Ethernet used by the Spectrum Scale clients to access data on the Elastic Storage Server solution.

► External or Campus Management Network: This network is used to remotely access and manage the EMS and HMC.

Note: Although connecting EMS to the Clustering Network is optional, if there is only one building block, the EMS has an active role in helping to keep the quorum in an I/O server outage. Also, the high-speed network can be used to gather data used by GUI software.

The number of storage enclosures and disks for each Elastic Storage Server model (GL or GS) is shown in Table 2-1.

Note: Figure 2-1 on page 9 shows multiple networks to illustrate all possible networking implementation options. However, only one network is required.

This network topology is a high-level logical view of the management and provisioning network and the service network for an ESS building block on PPC64BE not PPC64LE. For the latest documentation, refer to the Elastic Storage Server, Version 5.2, Quick Deployment Guide at the following website:

https://www.ibm.com/support/knowledgecenter/SSYSP8_5.2.0/ess_qdg.pdf?view=kc
Every disk enclosure used on models GL or GS have high availability features, such as:

- Duplicated ESMs.
- Duplicated power supplies.
- Duplicated fan assemblies.

The storage enclosures are connected to the servers in a twin-tailed fashion in such a way that all disk drives are connected to two nodes.

Figure 2-2 shows the Elastic Storage Server models combinations GL2, GL4, and GL6 that use storage enclosures model DCS3700.

### Table 2-1  Building block models

<table>
<thead>
<tr>
<th>Model</th>
<th>Enclosure type</th>
<th>Enclosures</th>
<th>Number of drives&lt;sup&gt;a&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>GS1</td>
<td>EXP24S</td>
<td>1</td>
<td>24 SSD</td>
</tr>
<tr>
<td>GS2</td>
<td></td>
<td>2</td>
<td>46 SAS + 2 SSD or 48 SSD</td>
</tr>
<tr>
<td>GS4</td>
<td></td>
<td>4</td>
<td>94 SAS + 2 SSD or 96 SSD</td>
</tr>
<tr>
<td>GS6</td>
<td></td>
<td>6</td>
<td>142 SAS + 2 SSD</td>
</tr>
<tr>
<td>GL2</td>
<td>DCS3700</td>
<td>2</td>
<td>166 NL-SAS + 2 SSD</td>
</tr>
<tr>
<td>GL4</td>
<td></td>
<td>4</td>
<td>334 NL-SAS + 2 SSD</td>
</tr>
<tr>
<td>GL6</td>
<td></td>
<td>6</td>
<td>502 NL-SAS + 2 SSD</td>
</tr>
</tbody>
</table>

<sup>a</sup> On the mixed HDD x SDD configurations, SDD is used only for Spectrum Scale RAID metadata.
ESS models

- **GLxS = High Capacity**
  - Analytics, Cloud Serving, Technical, Media etc.
  - Drive Capacity
    - 4 TB, 8 TB or 10 TB Nearline-SAS HDDs
    - Up to 3.7 PB usable

- **GS = High IOPS**
  - Hot data and/or Metadata
  - Drive Capacity
    - 400 GB – 1.9 TB SSDs or 1.2 TB, 1.8 TB SAS HDDs
    - Up to 175 TB usable

Network: 10 GbE, 40 GbE, InfiniBand, or mixed

*Figure 2-2  Elastic Storage Server models GL2, GL4, and GL6*

**Note:** In addition to the models that are shown in Figure 2-2 on page 11, the hardware configuration of the head nodes is flexible in regards to selecting network adapters and the amount of memory. Three PCI slots are reserved for SAS adapters and one PCI slot is configured by default with a 4-Port 10/100/1000 Ethernet Card for deployment. Three other PCIe3 slots are available to configure, with any combination of Dual Port 10 GbE, Dual Port 40 GbE, or Dual Port InfiniBand PCI adapters.

Each storage enclosure has two ESMs. The standard DCS3700 ESMs have two 6 Gbps x 4 SAS host interface ports and supports up to 60 drives. Figure 2-3 shows the DCS3700 with and without the front bezel.

*Figure 2-3  DCS3700*
Each enclosure has five drive drawers, numbered from top to bottom, as shown in Figure 2-4 on page 12.

![Figure 2-4 DCS3700 drive drawers](image)

On each drive drawer, you can have up to 12 drives as shown in Figure 2-5.

![Figure 2-5 DCS3700 drives](image)

On the GL2 model, the first DCS3700 has SSD drives on Drawer1 position 3 and Drawer5 position 12. On the second DCS3700, these same positions are empty. Refer to Figure 2-6.

On the GL4 and GL6 models, the first two enclosures have the same drive placement as the GL2. The other enclosures have 60 NL-SAS drives each.

![Figure 2-6 SSD drive placement](image)
Figure 2-7 shows the Elastic Storage Server models GS1 through GS6, which use storage enclosures model EXP24S.

Each storage enclosure is a 2U rack-mountable storage enclosure that supports two ESMs. The standard EXP24S ESMs have two 6 Gbps x 4 SAS host interface ports and support up to 24 drives. Figure 2-8 shows the EXP24S storage enclosure.

The GS models have configurations where all drives are SSD drives (400 GB or 800 GB), as shown in Figure 2-9.

The GS2, GS4, and GS6 configurations with mixed SSDs and HDDs have SSDs on the left-most position (slot 1) and on the right-most position (slot 4) of the first enclosure. All other drives are 1.2 TB HDDs.

For more information about the hardware components of the solution, visit the IBM Knowledge Center for the Elastic Storage Server at the following website:

### 2.2 Software components

This section lists the software components that are used on the Elastic Storage Server solution.
2.2.1 Operating system

The I/O nodes and the EMS run Red Hat Enterprise Linux 7.1 for 64-bit IBM Power Systems architecture.

2.2.2 Extreme Cluster/Cloud Administration Toolkit (xCAT)

The Elastic Storage Server deploys scripts in the EMS node to configure the xCAT server, which helps to deploy the I/O nodes.

2.2.3 IBM Spectrum Scale

Formerly GPFS, IBM Spectrum Scale is the clustered file system that turns the virtual disks provided by the Spectrum Scale RAID into Network Shared Disks (NSDs) and creates and exports the file system to the Spectrum Scale clients.

2.2.4 IBM Spectrum Scale RAID

The IBM Spectrum Scale RAID integrates all high availability and functionality features usually of an advanced storage server into the NSD server. Sophisticated data placement and error correction algorithms deliver high levels of storage reliability, availability, serviceability, and performance. Using Spectrum Scale software, the clients transparently access the *virtual disks*, or *vdisks* that are created by the Spectrum Scale RAID.

The following list discusses the Spectrum Scale RAID main features:

- **Software RAID**
  
  Spectrum Scale RAID runs on standard serial-attached SCSI (SAS) disks in a twin-tailed JBOD array, and does not require external RAID storage controllers or other custom hardware RAID acceleration and supports multiple RAID codes.

- **RAID codes**
  
  Spectrum Scale RAID corrects disk failures and other storage faults automatically by reconstructing the unreadable data using the available data redundancy of a Reed-Solomon code or $N$-way replication.

  The reconstructed data is used to fulfill client operations, and to rebuild the data onto spare space in case of a physical disk failure. Spectrum Scale RAID supports 2 and 3-fault-tolerant Reed-Solomon codes and 3-way and 4-way replication, which respectively detect and correct up to two or three concurrent faults. Refer to Figure 1-3 on page 4.

  The layouts for these redundancy codes called *tracks* are shown in Figure 1-1 on page 3 and in Figure 2-10.
When using Reed-Solomon code, Spectrum Scale RAID divides a Spectrum Scale block of user data equally into eight data stripes and generates two or three redundant parity stripes. This results in a stripe or track width of 10 or 11 stripes and storage efficiency of 80% or 73%, respectively.

For N-way replication, the Spectrum Scale data block is replicated two or three times, with the stripe size equal to the Spectrum Scale block. Therefore, for every block and stripe that is written to the disks, N replicas of that block and stripe are also written, resulting in storage efficiency of 33% or 25%.

**Note:** The storage efficiency using Reed-Solomon or N-way replication does not consider the spare space, which is user configured.

- **Declustering**

  Spectrum Scale RAID distributes client data, redundancy information, and spare space across all disks of an enclosure. In this approach, there is a significant improvement on the application performance, and the rebuild time overhead is reduced (disk failure recovery process) if compared to conventional RAID.
Figure 2-11 compares a conventional RAID of three 1+1 arrays (+1 spare disk) versus a declustered array, both using seven disks. On the lower left, you can see the traditional 1+1 RAID with the spare disk on three arrays of two disks each with data plus replica stripes and the spare disk.

On the declustered array the disks are divided into seven tracks, two stripes per array, as shown in the upper left. The stripes from each group are then spread across all seven disk positions, for a total of 21 virtual tracks. The stripes of each disk position for every track are then arbitrarily allocated onto the disks of the declustered array of the lower right. The spare stripes are uniformly inserted, one per disk.

When comparing the rebuild operation in Figure 2-12, you can see an example of how Spectrum Scale RAID has significant less impact on client performance during rebuild operations by utilizing the throughput of all disks in the declustered array.

In a disk loss, on the 1+1 array, all the data from one disk has to be replicated to the spare disk. On the declustered array, all remaining disks, with much lower impact, will be working on the rebuild operation.
Pdisk-group fault tolerance

Spectrum Scale RAID divides the disks in this called *recovery groups*, which are physically connected to two servers (primary and backup). All the I/O operations of a recovery group flow through the active server for that recovery group, either the primary or the backup. The native Spectrum Scale NSD failover capabilities give the Spectrum Scale RAID server the possibility to seamless failover the disks in the recovery group to another server in the event of a hardware or software failure or normal server shutdown.

Due to characteristics of a disk enclosure and system, multiple disks could possibly fail together due to a common fault. In this context, starting with Spectrum Scale 4.1, Spectrum Scale RAID uses a revised placement algorithm, which is aware of the hardware groupings, and distributes the stripes in such a way that allows data survival in the event of multiple concurrent failures, even for a large group of disks.

Figure 2-13 shows an example of a vdisk using a 4-way replication algorithm. The replicas were placed across four drawers on the two enclosures available in the recovery group.

![Figure 2-13   pdisk-group fault tolerance](image)

Even in the case of the loss of a full enclosure and a drawer of the other enclosure, data still survived.

*Note:* In the Elastic Storage Server, the rebuild operations can run at different priority levels with different levels of impact to the clients.

For example, in a 7+2 RAID, the loss of the first disk on the RAID runs at low priority with very low impact on the clients. In a second disk loss on the same RAID, the rebuild process gets higher priority.

> **Checksum**

In the Elastic Storage Server, the rebuild operations can run at different priority levels with different levels of impact to the clients.

For example, in a 7+2 RAID, the loss of the first disk on the RAID runs at low priority with very low impact on the clients. In a second disk loss on the same RAID, the rebuild process gets higher priority.
An end-to-end data integrity check, using checksum and version numbers, is maintained between the disk surface and NSD clients.

Most implementations of RAID codes and storage controllers implicitly rely on the physical disks to detect and report faults, hard-read errors, and other integrity problems. However, sometimes some read faults and occasionally fail to write data, while actually claiming to have written the data can occur.

These errors are often referred to as silent errors, phantom-writes, dropped-writes, and off-track writes.

Spectrum Scale RAID implements an end-to-end checksum that can detect silent data corruption caused by either disks or other system components that transport or manipulate the data.

When an NSD client is writing data, a checksum of 8 bytes is calculated and appended to the data before it is transported over the network to the Spectrum Scale RAID server. On reception, Spectrum Scale RAID calculates and verifies the checksum. Then, Spectrum Scale RAID stores the data, a checksum, and version number to disk and logs the version number in its metadata for future verification during read.

When Spectrum Scale RAID reads disks to satisfy a client read operation, it compares the disk checksum against the disk data and the disk checksum version number against what is stored in its metadata. If the checksum and version numbers match, Spectrum Scale RAID sends the data along with a checksum to the NSD client. If the checksum or version numbers are invalid, Spectrum Scale RAID reconstructs the data using parity or replication and returns the reconstructed data and a newly generated checksum to the client. Thus, both silent disk read errors and lost or missing disk writes are detected and corrected.

- Large cache
  Provided by the POWER8 servers, a large cache improves read and write performance, particularly for small I/O operations.

- Arbitrarily-sized disk arrays
  The number of disks is not restricted to a multiple of the RAID redundancy code width, which allows flexibility in the number of disks in the RAID array.

- Disk hospital
  One of the key features of Spectrum Scale RAID is the disk hospital. It asynchronously diagnoses errors and faults in the storage subsystem.

  Spectrum Scale RAID times out an individual pdisk I/O operation after about 10 seconds, thereby limiting the impact from a faulty pdisk on a client I/O operation. If there is a pdisk I/O operation time-out, an I/O error, or a checksum mismatch, the suspect pdisk is immediately admitted into the disk hospital. When a pdisk is first admitted, the hospital determines whether the error was caused by the pdisk itself or by the paths to it. Meanwhile, Spectrum Scale RAID uses vdisk redundancy codes to reconstruct lost or erased stripes for I/O operations that would otherwise have used the suspect pdisk.

  Some health metrics are maintained by the disk hospital. If or when one of these metrics exceeds the threshold, the pdisk is marked for replacement according to the disk maintenance replacement policy for the declustered array.

  These metrics are as follows:
  - relativePerformance, which characterizes response times. Values are compared to the average speed. If the metric falls below a particular threshold, the hospital adds “slow” to the pdisk state, and the disk is prepared for replacement.
  - dataBadness, which characterizes media errors (hard errors) and checksum errors.
The disk hospital logs selected Self Monitoring Analysis and Reporting Technology (SMART) data, including the number of internal sector remapping events for each pdisk.

- **Journaling**

For improved performance and recovery after a node failure, internal configuration and small-write data are journaled to solid-state disks (SSDs) in the JBOD or to non-volatile random-access memory (NVRAM) that is internal to the Spectrum Scale RAID servers.

Later versions of the Elastic Storage Server typically use NVRAM, with a single SSD per recovery group that is only used as a backup for the NVRAM in case of failure.

For more references, go to *IBM Spectrum Scale RAID 4.1.0.8 Administration*, available at the following website:

Planning and integration

This chapter contains guidelines and considerations for proper planning, installation, and configuration of the IBM Elastic Storage Server.

This chapter presents configurations and integration considerations for a smooth Elastic Storage Server deployment into an existing or a new IT environment, covering elements as follows:

- Elastic Storage Server overview
- Elastic Storage Server installation and upgrading
- Elastic Storage Server networks
- Elastic Storage Server: Storage parameters
- Best-practice recommendations for Spectrum Scale RAID
- Elastic Storage Server file system configurations
- Elastic Storage Server ordering
- Elastic Storage Server client integration
- Monitoring
3.1 Elastic Storage Server overview

As shown in Chapter 1, “Introduction to the IBM Elastic Storage Server” on page 1, the Elastic Storage Server is a high-capacity and high-performance storage system that combines IBM Power Systems servers, storage enclosures and drives, software (including IBM Spectrum Scale Redundant Array of Independent Disks (RAID)), and networking components.

The Elastic Storage Server V3.5 is an IBM file-based storage that has as its core technology the IBM Spectrum Scale V4.1.1, which includes IBM Spectrum Scale RAID combined with specific IBM Power Systems servers.

Multiple recovery groups can be defined, and a Spectrum Scale cluster node can be the primary or backup server for more than one recovery group. The name of a recovery group must be unique within a Spectrum Scale cluster.

3.2 Elastic Storage Server installation and upgrading

An Elastic Storage Server building block could be installed by the IBM Lab Services team as an included service part of acquisition, or could be also performed by the customer’s IT team.

The Elastic Storage Server documentation is at the following web page:
http://ibm.co/1L3foiM

The following documents provide required information for proper deployment, installation, and upgrade procedures for an IBM Elastic Storage System:

- IBM Elastic Storage Server: Planning for the system, service maintenance packages, and service procedures
  http://ibm.co/1LYNkBa
- IBM Elastic Storage Server: FAQs
  http://ibm.co/1SzLtoC

3.3 Elastic Storage Server networks

Proper network planning and configuration are important steps for a rapid and successful Elastic Storage Server deployment. This section provides recommendations and details regarding the Elastic Storage Server network communications setup and node-naming conventions as shown in Table 3-1 on page 23.

Note: Network planning is an important part of the preparation for deploying an Elastic Storage Server.
## Table 3-1 Network descriptions

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
<th>Remarks</th>
</tr>
</thead>
</table>
| 1.     | **Support network:**  
  - This private network connects the HMC with the flexible service processor (FSP) of the EMS and the I/O server nodes.  
  - HMC uses this network to discover the EMS and the I/O server nodes and perform hardware management, such as create and manage logical partitions, allocate resources, power control, reboot, and so on.  
  - This is a private network between the HMC and FSPs, must not be seen by the operating system running in the node being managed (for example, EMS and I/O server node).  
  - Requires planning if the HMC is not part of the original order and supplied by the customer. | EMS (xCAT) and I/O server (FSP) are connected to the HMC via the support network. The HMC must be set to be the DHCP server for the support network. |
| 2.     | **Management or provisioning network:**  
  - This network connects the EMS with the HMC and other I/O servers.  
  - This network is visible by the operating system running on the nodes.  
  - The EMS uses this network to communicate with the HMC and to discover I/O server nodes.  
  - This network is also used for provisioning the node and therefore deploys and installs the operating system in the I/O server nodes.  
  - There cannot be any other DHCP server on this network.  
  - Requires additional planning if the EMS is not part of the order and building blocks are managed by an existing EMS. | The HMC, and I/O server (OS) nodes are connected to the EMS node via this network. The EMS will be the DHCP server in this network (although it will not serve any IP address). |
| 3.     | **Public or clustering network**  
  This network is for high performance data access. This network in most cases can also be part of the clustering network. | This network is typically composed of 10 Gb, 40 Gb Ethernet or InfiniBand. |
| 4.     | **Domain of the management network**  
  This is used by the EMS for proper resolution of short host names and must be in lowercase. | Example: gpfs.net |
| 5.     | **HMC node IP address on the management network, short and long host name:**  
  - This IP address must be configured and the attached link to the network interface must be up.  
  - The EMS must be able to reach the HMC using this address. The long name and short name must be in lowercase.  
  - The host name must never end in “-enx” for any x. | Example:  
  IP: 192.168.45.9  
  Shortname: hmc1  
  FQDN: hmc1.gpfs.net |
<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.</td>
<td><strong>EMS node IP address</strong>, short name and long host name (FQDN):</td>
<td>Example: &lt;br&gt;IP: 192.168.45.10 &lt;br&gt;Shortname: ems1 &lt;br&gt;FQDN: ems1.gpfs.net</td>
</tr>
<tr>
<td></td>
<td>▶ This IP address must be configured and the link to the interface must be up.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▶ The management network must be reachable from this IP address.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▶ There is a valid gateway for this IP address.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▶ The long and short host name must be in lowercase.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▶ The host name must never end in “-enx” for any x.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▶ This address will be assigned to the I/O server nodes during node deployment.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▶ The I/O server nodes must be able to reach the management network via this address.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▶ The names defined here must match the name of the partition created for this node via the HMC.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▶ The long and short host name must be in lowercase.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▶ The host name must never end in “-enx” for any x.</td>
<td></td>
</tr>
<tr>
<td>8.</td>
<td><strong>EMS node management network interface</strong>:</td>
<td>Example: enP7p128s0f0</td>
</tr>
<tr>
<td></td>
<td>This interface must have the IP address of item number 6 of this table. This interface must have only one IP address assigned. It can be obtained with the <code>ip addr</code> command.</td>
<td></td>
</tr>
<tr>
<td>9.</td>
<td><strong>Prefix or Suffix for public/clustering network</strong>:</td>
<td>Suffix example: -ib, -10G, -40G</td>
</tr>
<tr>
<td></td>
<td>Management network typically runs over 1 Gb. Public and clustering runs on a high-speed network implemented on 10 Gb Ethernet, 40 Gb Ethernet or InfiniBand network. It is customary to use host names for the high-speed network using prefix and suffix of the actual host name. Do not use “-enx” for any x as a suffix.</td>
<td></td>
</tr>
<tr>
<td>10.</td>
<td><strong>High-speed cluster network IP address of the I/O server and the EMS nodes</strong>:</td>
<td>Example: &lt;br&gt;172.10.0.10 ems1-ib.gpfs.net &lt;br&gt;ems1-ib &lt;br&gt;172.10.0.11 gssio1-ib.data.net gssio1-ib &lt;br&gt;172.10.0.12 gssio2-ib.data.net gssio2-ib</td>
</tr>
<tr>
<td></td>
<td>In the example, 172.10.0.11 is the IP address that the GPFS daemon uses for clustering. Corresponding short and long names are gssio1-ib and gssio1-ib.data.net, respectively.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NOTE: Do not make changes in the <code>/etc/hosts</code> file for the high-speed network until the deployment is over. Do not create or enable the high-speed network interface until the deployment is over.</td>
<td></td>
</tr>
</tbody>
</table>
It is highly recommended to use a redundant network communication either for private or public networks. Bonding interface can be created for the Ethernet or the InfiniBand network interfaces. Examples of bonding network configurations are provided in the Deploying the Elastic Storage Server documentation found at the following website:

http://ibm.co/1lFrLNS

### 3.4 Elastic Storage Server: Storage parameters

The Elastic Storage Server uses declustered arrays implementing as controller the IBM Spectrum Scale RAID, which helps decrease the rebuilding impact and client overhead of a conventional RAID because IBM Spectrum Scale stripes client data across all the storage nodes of a cluster, thus the file system performance becomes less dependent on the speed of any single rebuilding.

This section presents the elements and associated storage parameters required to be taken into account for a proper planning and solution sizing.

#### 3.4.1 Recovery group server parameters

To enable a Spectrum Scale cluster node as a recovery group server, it must have the `mmchconfig` configuration parameter `nsdRAIDTracks` set to a nonzero value, and the GPFS daemon must be restarted on the node.

The `nsdRAIDTracks` parameter defines the maximum number of vdisk (“vdisk” on page 33) track descriptors that the server can have in memory at a given time. The volume of actual vdisk data that the server can cache in memory is governed by the size of the Spectrum Scale page pool on the server and the value of the `nsdRAIDBufferPoolSizePct` configuration parameter. The `nsdRAIDBufferPoolSizePct` parameter defaults to 50% of the page pool on the server. A recovery group server should be configured with a substantial amount of page pool, on the order of tens of gigabytes. A recovery group server becomes an NSD server after NSDs are defined on the vdisks in the recovery group, so the `nsdBufSpace` parameter also applies.

The default for `nsdBufSpace` is 30% of the page pool, and it can be decreased to its minimum value of 10% because the vdisk data buffer pool is used directly to serve the vdisk NSDs.
The vdisk track descriptors, as governed by nsdRAIDTracks, include such information as the RAID code, track number, and status. The descriptors also contain pointers to vdisk data buffers in the Spectrum Scale page pool, as governed by nsdRAIDBufferPoolSizePct. It is these buffers that hold the actual vdisk data and redundancy information.

**Recovery group creation**

Recovery groups are created using the mmcrrecoverygroup command, which takes the following arguments:

- The name of the recovery group to create.
- The name of a stanza file describing the declustered arrays and pdisks within the recovery group.
- The names of the Spectrum Scale cluster nodes that will be the primary and, if specified, backup servers for the recovery group.

When a recovery group is created, the GPFS daemon must be running with the nsdRAIDTracks configuration parameter in effect on the specified servers.

**Recovery group server failover**

When, as is recommended, a recovery group is assigned two servers, one server is the preferred and primary server for the recovery group and the other server is the backup server.

Only one server can serve the recovery group at any given time; this server is known as the active recovery group server. The server that is not currently serving the recovery group is the standby server. If the active recovery group server is unable to serve a recovery group, it relinquishes control of the recovery group and passes it to the standby server, if available.

The failover from the active to the standby server should be transparent to any Spectrum Scale file system using the vdisk NSDs in the recovery group. There is a pause in access to the file system data in the vdisk NSDs of the recovery group while the recovery operation takes place on the new server. This server failover recovery operation involves the new server opening the component disks of the recovery group and playing back any logged RAID transactions.

The active server for a recovery group can be changed by the IBM Spectrum Scale RAID administrator by using the mmchrecoverygroup command. This command can also be used to change the primary and backup servers for a recovery group.

### 3.4.2 pdisks

The IBM Spectrum Scale RAID pdisk is an abstraction of a physical disk. A pdisk corresponds to exactly one physical disk, and belongs to exactly one declustered array within exactly one recovery group. Before discussing how declustered arrays collect pdisks into groups, it is useful to describe the characteristics of pdisks.

A recovery group can contain a maximum of 512 pdisks. A declustered array within a recovery group can contain a maximum of 256 pdisks. The name of a pdisk must be unique within a recovery group. That is, two recovery groups can each contain a pdisk named disk10, but a recovery group cannot contain two pdisks named disk10, even if they are in different declustered arrays.
A pdisk is usually created by using the `mmcrrecoverygroup` command, whereby it is assigned to a declustered array within a newly created recovery group. In unusual situations, pdisks can also be created and assigned to a declustered array of an existing recovery group by using the `mmaddpdisk` command.

To create a pdisk, a stanza must be supplied to the `mmcrrecoverygroup` or `mmaddpdisk` commands specifying the pdisk name, the declustered array name to which it is assigned, and a block device special file name for the entire physical disk as it is configured by the operating system on the active recovery group server. A sample pdisk creation stanza follows:

```
%pdisk: pdiskName=c073d1
device=/dev/hdisk192
da=DA1
nPathActive=2
nPathTotal=4
```

Other stanza parameters might be present.

The device name for a pdisk must refer to the entire single physical disk. pdisks should not be created using virtualized or software-based disks (for example, logical volumes, disk partitions, logical units from other RAID controllers, or network-attached disks). The exception to this rule is non-volatile RAM (NVRAM) volumes used for the log tip vdisk, which is described in “Log vdisks” on page 34. For a pdisk to be created successfully, the physical disk must be present and functional at the specified device name on the active server. The physical disk must also be present on the standby recovery group server, if one is configured.

The physical disk block device special name on the standby server will almost certainly be different, and will be discovered automatically by IBM Spectrum Scale.

The attributes of a pdisk include the physical disk’s unique worldwide name (WWN), its field replaceable unit (FRU) code, and its physical location code. pdisk attributes can be displayed by using the `mmlspdisk` command. Of particular interest here are the pdisk device paths and the pdisk states.

pdisks that have failed and have been marked for replacement by the disk hospital are replaced by using the `mmchcarrier` command. In unusual situations, pdisks can be added or deleted by using the `mmaddpdisk` or `mmdelpdisk` commands. When deleted, either through replacement or the `mmdelpdisk` command, the pdisk abstraction only ceases to exist when all of the data it contained has been rebuilt onto spare space (even though the physical disk might have been removed from the system).

pdisks are normally under the control of IBM Spectrum Scale RAID and the disk hospital. In some situations, however, the `mmchdisk` command can be used to manipulate pdisks directly. For example, if a pdisk has to be removed temporarily to allow for hardware maintenance on other parts of the system, you can use the `mmchdisk --begin-service-drain` command to drain the data before removing the pdisk.

After bringing the pdisk back online, you can use the `mmchdisk --end-service-drain` command to return the drained data to the pdisk.

**Note:** This process requires that there be sufficient spare space in the declustered array for the data that is to be drained. If the available spare space is insufficient, it can be increased with the `mmchrecoverygroup` command.
pdisk paths

To the operating system, physical disks are made visible as block devices with device special file names, such as /dev/sdb (on Linux) or /dev/hdisk32 (on AIX). Most pdisks that IBM Spectrum Scale RAID uses are in JBOD arrays, except for the NVRAM pdisk that is used for the log tip vdisk. To achieve high availability and throughput, the physical disks of a JBOD array are connected to each server by multiple (usually two) interfaces in a configuration known as multipath (or dualpath). When two operating system block devices are visible for each physical disk, IBM Spectrum Scale RAID refers to them as the paths to the pdisk.

In normal operation, the paths to individual pdisks are discovered by IBM Spectrum Scale RAID automatically. There are only two instances when a pdisk must be referred to by its explicit block device path name: During recovery group creation using the mmcrrecoverygroup command, and when adding new pdisks to an existing recovery group with the mmaddpdisk command. In both of these cases, only one of the block device path names as seen on the active server needs to be specified. Any other paths on the active and standby servers are discovered automatically.

For each pdisk, the nPathActive and nPathTotal stanza parameters can be used to specify the expected number of paths to that pdisk, from the active server and from all servers. This allows the disk hospital to verify that all expected paths are present and functioning.

The operating system could have the ability to internally merge multiple paths to a physical disk into a single block device. When IBM Spectrum Scale RAID is in use, the operating system multipath merge function must be disabled because IBM Spectrum Scale RAID itself manages the individual paths to the disk.

pdisk stanza format

pdisk stanzas have three mandatory parameters and five optional parameters, and look as follows:

```
%pdisk: pdiskName=PdiskName
device=BlockDeviceName
da=DeclusteredArrayName
[nPathActive=ExpectedNumberActivePaths]
[nPathTotal=ExpectedNumberTotalPaths]
[rotationRate=HardwareRotationRate]
[fruNumber=FieldReplaceableUnitNumber]
[location=PdiskLocation]
```

where:

- **pdiskName=PdiskName**
  - Specifies the name of a pdisk.

- **device=BlockDeviceName**
  - Specifies the name of a block device. The value provided for BlockDeviceName must refer to the block device as configured by the operating system on the primary recovery group server or have the node name prefixed to the device block name.

  Sample values for BlockDeviceName are /dev/sdb (on Linux), //nodename/dev/sdb (on Linux), and /dev/rdisk32 (on AIX).

  Only one BlockDeviceName needs to be used, even if the device uses multipath and has multiple device names.
da=DeclusteredArrayName

Specifies the DeclusteredArrayName in the pdisk stanza, which implicitly creates the declustered array with default parameters.

nPathActive=ExpectedNumberActivePaths

Specifies the expected number of paths for the connection from the active server to this pdisk. If this parameter is specified, the mmlsrecoverygroup and mmlspdisk commands display warnings if the number of paths does not match the expected number for a pdisk that should be functioning normally. If this parameter is not specified, the default is 0, which means “do not issue such warnings”.

Sample values are 2 for all pdisks that are in an Elastic Storage Server disk enclosure (or the IBM Power 775 Disk Enclosure) and 1 for the NVRAM pdisk that is used for the log tip vdisk.

nPathTotal=ExpectedNumberTotalPaths

Specifies the expected number of paths for the connection from all active and backup servers to this pdisk. If this parameter is specified, the mmlsrecoverygroup and mmlspdisk commands display warnings if the number of paths does not match the expected number for a pdisk that should be functioning normally. If this parameter is not specified, the default is 0, which means “do not issue such warnings”.

Sample values are 4 for all pdisks in an Elastic Storage Server disk enclosure (or the IBM Power 775 Disk Enclosure) and 1 for the NVRAM pdisk used for the log tip vdisk.

rotationRate=HardwareRotationRate

Specifies the hardware type of the pdisk: NVRAM, SSD, or a rotating HDD. The only valid values are the string NVRAM, the string SSD, or a number between 1025 and 65535 (inclusive) indicating the rotation rate in revolutions per minute for HDDs. For all pdisks that are used in an Elastic Storage Server disk enclosure (or the IBM Power 775 Disk Enclosure), there is no need to specify this parameter because the hardware type and rotation rate are determined from the hardware automatically. This parameter should only be specified for the NVRAM pdisk on the Elastic Storage Server. The default is to rely on the hardware to identify itself, or leave the hardware type and rotation rate unknown if the hardware does not have the ability to identify itself.

A sample value is the string NVRAM for the NVRAM pdisk used for the log tip vdisk.

fruNumber=FieldReplaceableUnitNumber

Specifies the unit number for the field-replaceable unit (FRU) that is needed to repair this pdisk if it fails. For all pdisks used in an Elastic Storage Server disk enclosure (or the IBM Power 775 Disk Enclosure), there is no need to specify this parameter because it is automatically determined from the hardware. For the NVRAM pdisk used in the log tip vdisk, the user can enter a string here, which is displayed to service personnel when replacement of that pdisk is performed. However, setting this value for the NVRAM pdisk is not required because the service replacement procedure for that pdisk is specific to that particular type of hardware. The default is to rely on the hardware to identify itself, or to leave the FRU number unknown if the hardware does not have the ability to identify itself.

location=PdiskLocation

Specifies the physical location of this pdisk. For all pdisks used in an Elastic Storage Server disk enclosure (or the IBM Power 775 Disk Enclosure), there is no need to specify this parameter because it is automatically determined from the hardware. For the NVRAM pdisk used in the log tip vdisk, the user can enter a string here, which is displayed in the output of mmlspdisk. The default is to rely on the location reported by the hardware, or leave the location unknown.
A sample value is SV21314035-5-1, which describes a pdisk in enclosure serial number SV21314035, drawer 5, slot 1.

**pdisk states**

IBM Spectrum Scale RAID maintains its view of a pdisk and its corresponding physical disk by using a pdisk state. The pdisk state consists of multiple keyword flags, which can be displayed by using the `mmlsrecoverygroup` or `mmlspdisk` commands. You can also use the Elastic Storage Server GUI to display pdisk states. The state of pdisks is displayed in these views: **Arrays** → **Physical**, **Monitoring** → **System**, and **Monitoring** → **System Details**. In addition, information about pdisks with a negative state (disks that should be replaced, for example) is displayed in the **Monitoring** → **Events** view.

The pdisk state flags indicate in detail how IBM Spectrum Scale RAID is currently using or managing a disk. The state of a pdisk is also summarized in its user condition, as described at the end of this section.

In normal circumstances, the state of the vast majority of pdisks is represented by the sole keyword *ok*. This means that IBM Spectrum Scale RAID considers the pdisk to be healthy: The recovery group server is able to communicate with the disk, the disk is functioning normally, and the disk can be used to store data. The diagnosing flag is present in the pdisk state when the IBM Spectrum Scale RAID disk hospital suspects, or attempts to correct, a problem.

If IBM Spectrum Scale RAID is unable to communicate with a disk, the pdisk state includes the keyword *missing*. If a missing disk becomes reconnected and functions properly, its state changes back to *ok*. The readonly flag means that a disk has indicated that it can no longer safely write data. A disk can also be marked by the disk hospital as failing, perhaps due to an excessive number of media or checksum errors. When the disk hospital concludes that a disk is no longer operating effectively, it declares the disk to be dead.

If the number of non-functioning (dead, missing, failing, or slow) pdisks reaches or exceeds the replacement threshold of their declustered array, the disk hospital adds the flag *replace* to the pdisk state, which indicates that physical disk replacement should be performed as soon as possible.

When the state of a pdisk indicates that it can no longer behave reliably, IBM Spectrum Scale RAID rebuilds the pdisk's data onto spare space on the other pdisks in the same declustered array. This is called *draining the pdisk*. That a pdisk is draining or has been drained will be indicated by a keyword in the pdisk state flags. The flag *systemDrain* means that IBM Spectrum Scale RAID has decided to rebuild the data from the pdisk. The flag *adminDrain* means that the IBM Spectrum Scale RAID administrator issued the `mmdelpdisk` command to delete the pdisk.

IBM Spectrum Scale RAID stores both user (Spectrum Scale file system) data and its own internal recovery group data and vdisk configuration data on pdisks. Additional pdisk state flags indicate when these data elements are not present on a pdisk. When a pdisk starts draining, IBM Spectrum Scale RAID first replicates the recovery group data and vdisk configuration data onto other pdisks. When this completes, the flags *noRGD* (no recovery group data) and *noVCD* (no vdisk configuration data) are added to the pdisk state flags. When the slower process of removing all user data completes, the *noData* flag is added to the pdisk state.
To summarize, the vast majority of pdisks are in the ok state during normal operation. The ok state indicates that the disk is reachable, functioning, not draining, and that the disk contains user data and IBM Spectrum Scale RAID recovery group and vdisk configuration information. A more complex example of a pdisk state is dead/systemDrain/noRGD/noVCD/noData for a single pdisk that has failed. This set of pdisk state flags indicates that the pdisk was declared dead by the system, was marked to be drained, and that all of its data (recovery group, vdisk configuration, and user) has been successfully rebuilt onto the spare space on other pdisks.

### 3.4.3 Declustered arrays

Declustered arrays are disjoint subsets of the pdisks in a recovery group. vdisks are created within declustered arrays, and vdisk tracks are declustered across all of an array’s pdisks. A recovery group can contain up to 16 declustered arrays. A declustered array can contain up to 256 pdisks (but the total number of pdisks in all declustered arrays within a recovery group cannot exceed 512).

A pdisk can belong to only one declustered array. The name of a declustered array must be unique within a recovery group. That is, two recovery groups can each contain a declustered array named DA3, but a recovery group cannot contain two declustered arrays named DA3. The pdisks within a declustered array must all be of the same size and should all have similar performance characteristics.

A declustered array is usually created together with its member pdisks and its containing recovery group by using the `mmchrecoverygroup` command. A declustered array can also be created by using the `mmadddisk` command to add pdisks to a declustered array that does not yet exist in a recovery group.

A declustered array can be deleted by deleting its last member pdisk, or by deleting the recovery group in which it resides. Any vdisk NSDs and vdisks within the declustered array must already have been deleted. There are no explicit commands to create or delete declustered arrays.

The main purpose of a declustered array is to segregate pdisks of similar performance characteristics and similar use. Because vdisks are contained within a single declustered array, mixing pdisks of varying performance within a declustered array would not use the disks optimally. In a typical IBM Spectrum Scale RAID system, the first declustered array contains SSD pdisks that are used for the log vdisk, or the log backup vdisk if configured. If the system is configured to use a log tip vdisk, another declustered array contains NVRAM pdisks for that vdisk. vdisks that are Spectrum Scale NSDs are then contained in one or more declustered arrays using high-capacity HDDs or SSDs.

A secondary purpose of declustered arrays is to partition disks that share a common point of failure or unavailability, such as removable carriers that hold multiple disks. This comes into play when one considers that removing a multi-disk carrier to perform disk replacement also temporarily removes some good disks, perhaps a number in excess of the fault tolerance of the vdisk NSDs. This would cause temporary suspension of file system activity until the disks are restored. To avoid this, each disk position in a removable carrier should be used to define a separate declustered array, such that disk position one defines DA1, disk position two defines DA2, and so on. Then when a disk carrier is removed, each declustered array suffers the loss of just one disk, which is within the fault tolerance of any IBM Spectrum Scale RAID vdisk NSD.
Data spare space and VCD spares

While operating with a failed pdisk in a declustered array, IBM Spectrum Scale RAID continues to serve file system I/O requests by using redundancy information about other pdisks to reconstruct data that cannot be read, and by marking data that cannot be written to the failed pdisk as stale. Meanwhile, to restore full redundancy and fault tolerance, the data on the failed pdisk is rebuilt onto data spare space, reserved unused portions of the declustered array that are declustered over all of the member pdisks. The failed disk is thereby drained of its data by copying it to the data spare space.

The amount of data spare space in a declustered array is set at creation time and can be changed later.

The data spare space is expressed in whole units equivalent to the capacity of a member pdisk of the declustered array, but is spread among all of the member pdisks. There are no dedicated spare pdisks.

This implies that a number of pdisks equal to the specified data spare space could fail, and the full redundancy of all of the data in the declustered array can be restored through a rebuild operation. If the user chooses to not fill the space in the declustered array with vdisks, and wants to use the deallocated space as extra data spare space, the user can increase the setting of the dataSpares parameter to the wanted level of resilience against pdisk failures.

At a minimum, each declustered array normally requires data spare space that is equivalent to the size of one member pdisk. The exceptions, which have zero data spares and zero VCD spares, are declustered arrays that consist of the following:

- Non-volatile RAM disks used for a log tip vdisk.
- SSDs used for a log tip backup vdisk.

Because large declustered arrays have a greater probability of disk failure, the default amount of data spare space depends on the size of the declustered array. A declustered array with nine or fewer pdisks defaults to having one disk of equivalent data spare space. A declustered array with 10 or more disks defaults to having two disks of equivalent data spare space. These defaults can be overridden, especially at declustered array creation. However, if at a later point too much of the declustered array is already allocated for use by vdisks, it might not be possible to increase the amount of data spare space.

IBM Spectrum Scale RAID vdisk configuration data (VCD) is stored more redundantly than vdisk content, typically 5-way replicated. When a pdisk fails, this configuration data is rebuilt at the highest priority, onto functioning pdisks. The redundancy of configuration data always has to be maintained, and IBM Spectrum Scale RAID will not serve a declustered array that does not have sufficient pdisks to store all configuration data at full redundancy. The declustered array parameter vcdSpares determines how many pdisks can fail and have full VCD redundancy restored, by reserving room on each pdisk for vdisk configuration data. When using pdisk-group fault tolerance, the value of vcdSpares should be set higher than the value of the dataSpares parameter to account for the expected failure of hardware failure domains.

Increasing VCD spares

When new recovery groups are created, the mkrginput script sets recommended values for VCD spares.

To increase the VCD spares for existing recovery groups, use the mmchrecoverygroup command.
**Declustered array free space**
The declustered array free space reported by the `mmlsrecoverygroup` command reflects the space available for creating vdisks. Spare space is not included in this value since it is not available for creating new vdisks.

**pdisk free space**
The pdisk free space reported by the `mmlsrecoverygroup` command reflects the actual number of unused data partitions on the disk. This includes spare space, so if a pdisk fails, these values decrease as data is moved to the spare space.

**vdisks**
vdisks are created across the pdisks within a declustered array. Each recovery group requires a special log home vdisk to function (along with other log-type vdisks, as appropriate for specific environments); see “Log vdisks” on page 34. All other vdisks are created for use as Spectrum Scale file system NSDs.

A recovery group can contain at most 64 vdisks. vdisks can be allocated arbitrarily among declustered arrays. vdisks are created with the `mmcrvdisk` command. The `mmdelvdisk` command destroys vdisks and all their contained data.

When creating a vdisk, you must specify the RAID code, block size, vdisk size, and a name that is unique within the recovery group and the Spectrum Scale cluster. There are no adjustable parameters available for vdisks.

**RAID code**
The type, performance, and space efficiency of the RAID codes used for vdisks, discussed in “RAID codes” on page 14, should be considered when choosing the RAID code for a particular set of user data.

Spectrum Scale storage pools and policy-based data placement can be used to ensure data is stored with appropriate RAID codes.

The vdisk block size must equal the Spectrum Scale file system block size of the storage pool where the vdisk is assigned. For replication codes, the supported vdisk block sizes are 256 KiB, 512 KiB, 1 MiB, and 2 MiB.

For Reed-Solomon codes, the supported vdisk block sizes are 512 KiB, 1 MiB, 2 MiB, 4 MiB, 8 MiB, and 16 MiB.

The `maxblocksize` configuration attribute of the IBM Spectrum Scale `mmchconfig` command must be set appropriately for all nodes. The value of `maxblocksize` must be greater than or equal to the maximum block size of the vdisks. For more information about this attribute, see the `mmchconfig` command description in *IBM Spectrum Scale: Administration and Programming Reference* at the following website:

http://ibm.co/20XE4oD

**vdisk size**
The maximum vdisk size is the total space available on the pdisks in the declustered array, taking into account the overhead of the RAID code, minus spare space, minus vdisk configuration data, and minus a small amount of space reserved as a buffer for write operations. IBM Spectrum Scale RAID will round up the requested vdisk size as required. When creating a vdisk, the user can specify to use all remaining space in the declustered array for that vdisk.
Log vdisks
IBM Spectrum Scale RAID uses log vdisks to store such internal information as event log entries, updates to vdisk configuration data, and certain data write operations quickly. There are four types of log vdisks, as follows. Among them, they can be created and destroyed in any order:

- Log home vdisk
  Every recovery group requires one log home vdisk to function. The log home vdisk must be created before any other non-log vdisks in the recovery group, and it can only be deleted after all other non-log vdisks in the recovery group have been deleted. The log home vdisk is divided into four sublogs: Long-term event log, short-term event log, metadata log, and fast-write log to log small write operations.

- Log tip vdisk
  The log tip vdisk (appropriate for certain environments, but not required for all) is a vdisk to which log records are initially written, then migrated to the log home vdisk. The intent is to use a small, high-performance NVRAM device for the log tip, and a larger vdisk on conventional spinning disks for the log home vdisk. The fast writes to the log tip hide the latency of the spinning disks used for the main body of the log.

- Log tip backup vdisk
  The log tip backup vdisk (appropriate for certain environments, but not required for all) is used as an additional replica of the log tip vdisk when the log tip vdisk is two-way replicated on nonvolatile RAM disks. Ideally, the log tip backup vdisk provides a level of performance between that of NVRAM disks and that of spinning disks.

- Log reserved vdisk
  Log reserved vdisks are optional vdisks that are used when the log home disk is not allocated in its own declustered array. Log reserved vdisks have the same size as the log home vdisk and are used to equalize the space consumption on the data declustered arrays, but they are otherwise unused.

Declustered array parameters
Declustered arrays have four parameters that can be set by using stanza parameters when creating a declustered array, and can be changed by using the `mmchrecoverygroup` command with the `--declustered-array` option. Following are the four parameters:

- dataSpares
  The number of disks worth of equivalent spare space used for rebuilding vdisk data if pdisks fail. This defaults to one for arrays with nine or fewer pdisks, and two for arrays with 10 or more pdisks.

- vcdSpares
  The number of disks that can be unavailable while the IBM Spectrum Scale RAID server continues to function with full replication of vdisk configuration data (VCD). This value defaults to the number of data spares. To enable pdisk-group fault tolerance, this parameter is typically set to a larger value during initial system configuration (half of the number of pdisks in the declustered array + 1, for example).

- replaceThreshold
  The number of disks that must fail before the declustered array is marked as needing to have disks replaced. The default is the number of data spares.

- scrubDuration
  The number of days over which all the vdisks in the declustered array is scrubbed for errors. The default is 14 days.
3.4.4 Typical configurations

The following are descriptions of typical vdisk configurations in various recovery group environments:

- Elastic Storage Server without NVRAM disks
  In this configuration, a three-way replicated log tip vdisk is allocated on a declustered array made up of three SSDs. A four-way replicated log home vdisk is allocated in the first declustered array of HDDs.

- Elastic Storage Server with NVRAM disks
  In this configuration, a two-way replicated log tip vdisk is allocated on NVRAM disks, one from each of the servers.
  A log tip backup vdisk is allocated on a declustered array of one or more SSDs. This provides an additional copy of the log tip data when one of the NVRAM disks is unavailable. If only one SSD is used, the log tip backup uses a RAID code of Unreplicated.
  A four-way replicated log home vdisk is allocated in the first declustered array of HDDs. A four-way replicated log reserved vdisk for each of the data declustered arrays that do not contain the log home vdisk.

- Elastic Storage Server with NVRAM disks, using SSDs for data
  In this configuration, a two-way replicated log tip vdisk is allocated on NVRAM disks, one from each of the servers.
  All SSDs for a recovery group form a single declustered array, containing the log home vdisk and user data vdisks. No log tip backup disk is used.

- Power 775 configuration
  In this configuration, the log home vdisk is allocated on a declustered array made up of four SSDs. Only three-way and four-way replication codes are supported for the log home vdisk. In the typical system with four SSDs and with spare space equal to the size of one disk, the three-way replication code would be used for the log home vdisk.

3.5 Best-practice recommendations for Spectrum Scale RAID

In the Elastic Storage Server, the IBM Spectrum Scale RAID implementation best practices are enforced as de facto standards allowing configuration parameters modifications for achieving the best storage performance. The Elastic Storage Server implementation uses JBOD arrays, the required redundancy protection and usable disk capacity, the required spare capacity and maintenance strategy, and also the ultimate Spectrum Scale file system configuration.

The following IBM Spectrum Scale recommendations are fulfilled or suitable for the Elastic Storage Server as well:

- A primary and backup server is assigned to each recovery group.
  Each JBOD array is connected to two servers to protect against server failure. Each server has two independent paths to each physical disk to protect against path failure and provide higher throughput to the individual disks.
  Recovery group server nodes are designated Spectrum Scale manager nodes, and they are dedicated to IBM Spectrum Scale RAID and not run application workload.
Configure recovery group servers with a large vdisk track cache and a large page pool.

The `nsdRAIDTracks` configuration parameter tells IBM Spectrum Scale RAID how many vdisk track descriptors, not including the actual track data, to cache in memory.

In general, many vdisk track descriptors should be cached. The `nsdRAIDTracks` value for the recovery group servers should be 10000 - 60000. If the expected vdisk NSD access pattern is random across all defined vdisks and within individual vdisks, a larger value for `nsdRAIDTracks` might be warranted. If the expected access pattern is sequential, a smaller value can be sufficient.

The amount of actual vdisk data (including user data, parity, and checksums) that can be cached depends on the size of the Spectrum Scale page pool on the recovery group servers and the percentage of page pool reserved for IBM Spectrum Scale RAID. The `nsdRAIDBufferPoolSizePct` parameter specifies what percentage of the page pool should be used for vdisk data. The default is 50%, but it can be set as high as 90% or as low as 10%. Because a recovery group server is also an NSD server and the vdisk buffer pool also acts as the NSD buffer pool, the configuration parameter `nsdBufSpace` should be reduced to its minimum value of 10%.

As an example, to have a recovery group server cache 20000 vdisk track descriptors (`nsdRAIDTracks`), where the data size of each track is 4 MiB, using 80% (`nsdRAIDBufferPoolSizePct`) of the page pool size of 20000 * 4 MiB * (100/80) ~ 100000 MiB ~ 98 GiB would be required. It is not necessary to configure the page pool to cache all the data for every cached vdisk track descriptor, but this example calculation can provide some guidance in determining appropriate values for `nsdRAIDTracks` and `nsdRAIDBufferPoolSizePct`.

Define each recovery group with at least one large declustered array.

A large declustered array contains enough pdisks to store the required redundancy of IBM Spectrum Scale RAID vdisk configuration data. This is defined as at least nine pdisks plus the effective spare capacity. A minimum spare capacity equivalent to two pdisks is strongly recommended in each large declustered array. The code width of the vdisks must also be considered. The effective number of non-spare pdisks must be at least as great as the largest vdisk code width. A declustered array with two effective spares where 11 is the largest code width (8 + 3p Reed-Solomon vdisks) must contain at least 13 pdisks. A declustered array with two effective spares in which 10 is the largest code width (8 + 2p Reed-Solomon vdisks) must contain at least 12 pdisks.

Define the log vdisks based on the type of configuration.

See 3.4.4, “Typical configurations” on page 35 and “Log vdisks” on page 34.

Determine the declustered array maintenance strategy.

Disks fail and need replacement, so a general strategy of deferred maintenance can be used. For example, failed pdisks in a declustered array are only replaced when the spare capacity of the declustered array is exhausted. This is implemented with the replacement threshold for the declustered array set equal to the effective spare capacity. This strategy is useful in installations with many recovery groups where disk replacement might be scheduled weekly. Smaller installations can have IBM Spectrum Scale RAID require disk replacement as disks fail, which means the declustered array replacement threshold can be set to 1.

Choose the vdisk RAID codes based on the Spectrum Scale file system usage.

The choice of vdisk RAID codes depends on the level of redundancy protection required versus the amount of actual space required for user data, and the ultimate intended use of the vdisk NSDs in a Spectrum Scale file system.
Reed-Solomon vdisks are more space efficient. An 8 + 3p vdisk uses approximately 27% of actual disk space for redundancy protection and 73% for user data. An 8 + 2p vdisk uses 20% for redundancy and 80% for user data. Reed-Solomon vdisks perform best when writing whole tracks (the Spectrum Scale block size) at once. When partial tracks of a Reed-Solomon vdisk are written, parity recalculation must occur.

Replicated vdisks are less space efficient. A vdisk with 3-way replication uses approximately 67% of actual disk space for redundancy protection and 33% for user data. A vdisk with 4-way replication uses 75% of actual disk space for redundancy and 25% for user data. The advantage of vdisks with N-way replication is that small or partial write operations can complete faster.

For file system applications where write performance must be optimized, the preceding considerations make replicated vdisks most suitable for use as Spectrum Scale file system metadataOnly NSDs, and Reed-Solomon vdisks most suitable for use as Spectrum Scale file system dataOnly NSDs. The volume of the Spectrum Scale file system metadata is usually small (1 - 3%) relative to file system data, so the impact of the space inefficiency of a replicated RAID code is minimized. The file system metadata is typically written in small chunks, which takes advantage of the faster small and partial write operations of the replicated RAID code. Applications are often tuned to write file system user data in whole multiples of the file system block size, which works to the strengths of the Reed-Solomon RAID codes both in terms of space efficiency and speed.

When segregating vdisk NSDs for file system metadataOnly and dataOnly disk usage, the metadataOnly replicated vdisks can be created with a smaller block size and assigned to the Spectrum Scale file system storage pool. The dataOnly Reed-Solomon vdisks can be created with a larger block size and assigned to the Spectrum Scale file system data storage pools. When using multiple storage pools, a Spectrum Scale placement policy must be installed to direct file system data to non-system storage pools.

When write performance optimization is not important, it is acceptable to use Reed-Solomon vdisks as dataAndMetadata NSDs for better space efficiency.

3.6 Elastic Storage Server file system configurations

This section describes the required steps for IBM Spectrum Scale file system creation using the Elastic Storage Server command-line interface (CLI) commands considering the Elastic Storage Server installed and all verification steps have been performed according to the Elastic Storage Server documentation.

Hardware verification steps assume that the software is installed in the I/O server nodes.

The system verification steps consist of information validation reported with the following commands:

1. `gssstoragequickcheck` checks the server, adapter, and storage configuration quickly.
2. `gssfindmissingdisks` checks the disk paths and connectivity.
3. `gsscheckdisks` checks for disk errors under various I/O operations.
The file system configuration is performed from the management server node as follows:

- Run the `gssgencluster` command from the management server node.

  Run the `gssgencluster` command on the management server to create the cluster. This command creates a Spectrum Scale cluster using all of the nodes in the node group if you specify the `-G` option. You can also provide a list of names using the `-N` option. The command assigns server licenses to each I/O server node, so it prompts for license acceptance (or use the `-accept-license` option). It applies the best-practice Spectrum Scale configuration parameters for a Spectrum Scale RAID-based NSD server. At the end of cluster creation, the SAS adapter firmware, storage enclosure firmware, and drive firmware are upgraded if needed. To bypass the firmware update, specify the `--no-fw-update` option.

  **Note:** This command could take some time to run.

- Verify if the cluster is active.

  Log in to one of the I/O server nodes and verify that the cluster is created correctly. Run: `mmlscluster`

- Create recovery groups.

  The `gssgenclusterrgs` command creates the recovery groups (RGs) and the associated log tip vdisk, log backup vdisk, and log home vdisk. This command can create NSDs and file systems for simple configurations that require one file system. More flexibility can be achieved using `gssgencluster` to create the recovery groups only and using `gssgendisks` (the preferred method) to create data vdisks, metadata vdisks, NSDs, and file systems. For compatibility with an earlier version, the `gssgenclusterrgs` command continues to support vdisk, NSD, and file system creation.

  The `gssgenclusterrgs` command creates and saves the stanza files for the data and metadata vdisks and NSD. The stanza files are in the `/tmp` directory of the first node of the first building block with names `node1_node2_vdisk.cfg.save` and `node1_node2_nsd.cfg.save`. These files can be edited for further customization.

  If a customized recovery stanza file is available, it can be used to create the recovery group. The files must be on the first node (in the node list) of each building block in `/tmp`. Their names must be in the format `xxxxL.stanza` and `yyyyR.stanza`, where `L` is for the left recovery group and `R` is for the right recovery group. The name of the recovery group is derived from the I/O server node's short name (with prefix and suffix) by adding a prefix of `rg_`. When the `--create-nsds` option is specified, by default, 1% of the space is left as reserved and the remaining space is used to create the NSDs. The amount of reserved space is user-selectable and the default is 1% of the total raw space. The percentage of reserved space is based on the total raw space (not on the available space) before any redundancy overhead is applied.

  If the system already contains recovery groups and log vdisks (created in the previous steps), their creation can be skipped using the appropriate options. This can be useful when NSDs are re-created (for a change in the number of NSDs or block size, for example).
Verify the recovery group configuration.

To view the details for one of the recovery groups, log on to one of the I/O server nodes and run: `mmlsrecoverygroup`.

Running `mmlsrecoverygroup` with no parameters lists all of the recovery groups in your Spectrum Scale cluster.

For each recovery group:
- NVR contains the NVRAM devices used for the log tip vdisk.
- SSD contains the SSD devices used for the log backup vdisk.
- DA1 contains the SSD or HDD devices used for the log home vdisk and file system data.
- DAn, where n > 1 (depending on the Elastic Storage Server model and with v3.5 by default allows only one DA), contains the SSD or HDD devices used for file system data.

**Note:** In an SSD-only configuration, there is no SSD declustered array (DA) for log backup because DA1 to DAn are composed of SSD devices.

Create vdisk stanza.

Use `gssgendisks` to create the vdisk stanza file. By default, the vdisk stanza is stored in `/tmp/vdisk1.cfg`. Optionally, `gssgendisks` can be used to create vdisks, NSDs, and the file system on existing recovery groups. If no recovery groups are specified, all available recovery groups are used. If the command is run on the management server node (or any other node) that is not part of the cluster, a contact node that is part of the cluster must be specified. The contact node must be reachable from the node (the management server node, for example) where the command is run.

You can use this command to add a suffix to vdisk names, which can be useful when creating multiple file systems. A unique suffix can be used with a vdisk name to associate it with a different file system (examples to follow). The default reserve capacity is set to 1%. If the vdisk data block size is less than 8 M, the reserved space should be increased with decreasing the data vdisk block size.

This command can be used to create a shared-root file system for IBM Spectrum Scale protocol nodes.

**Note:** NSDs that are in the same building block are given the same failure group by default. If file system replication is set to 2 (m=2 or r=2), there should be more than one building block or the failure group of the NSDs must be adjusted accordingly.
Reserved space configurations.

When all available space is allocated, the reserved space should be increased with decreasing data vdisk block size. A default reserved space of 1% works well for a block size of up to 4 MB. For a 2 MB block size, 2% should be reserved. For a 1 MB block size, reserved space should be increased to 3%.

Creating file systems by using the `gssgenvdisks` command.

Create two file systems, one with 20 TB (two vdisks, 10 TB each), and the other with 40 TB (two vdisks, 20 TB each) with a RAID code of 8+3p as shown in Example 3-1.

Example 3-1  Creating two file systems RAID code of 8+3p

```bash
[root@ems1 ~]# gssgenvdisks --contact-node gssio1 --create-vdisk --create-nsds --create-filesystem --vdisk-suffix=_fs1 --filesystem-name fs1 --data-vdisk-size 10240
2015-06-16T00:50:37.254906 Start creating vdisk stanza
vdisk stanza saved in gssio1:/tmp/vdisk1.cfg
2015-06-16T00:50:51.809024 Generating vdisks for nsd creation
2015-06-16T00:51:27.409034 Creating nsds
2015-06-16T00:51:35.266776 Creating filesystem
Filesystem successfully created. Verify failure group of nsds and change as needed.
2015-06-16T00:51:46.688937 Applying data placement policy
2015-06-16T00:51:51.637243 Task complete.
```

```bash
Filesystem Size Used Avail Use% Mounted on
/dev/sda3 246G 2.9G 244G 2% /
devtmpfs 60G 0 60G 0% /dev
tmpfs 60G 0 60G 0% /dev/shm
tmpfs 60G 43M 60G 1% /run
tmpfs 60G 0 60G 0% /sys/fs/cgroup
/dev/sda2 497M 161M 336M 33% /boot
/dev/fs1 21T 160M 21T 1% /gpfs/fs1
```

```bash
[root@ems1 ~]# gssgenvdisks --contact-node gssio1 --create-vdisk --create-nsds --create-filesystem --vdisk-suffix=_fs2 --filesystem-name fs2 --data-vdisk-size 20480 --raid-code 8+3p
2015-06-16T01:06:59.929580 Start creating vdisk stanza
vdisk stanza saved in gssio1:/tmp/vdisk1.cfg
2015-06-16T01:07:13.019100 Generating vdisks for nsd creation
2015-06-16T01:07:56.688530 Creating nsds
2015-06-16T01:08:04.516814 Creating filesystem
Filesystem successfully created. Verify failure group of nsds and change as needed.
2015-06-16T01:08:16.613198 Applying data placement policy
```

```bash
Filesystem Size Used Avail Use% Mounted on
/dev/sda3 246G 2.9G 244G 2% /
devtmpfs 60G 0 60G 0% /dev
tmpfs 60G 0 60G 0% /dev/shm
tmpfs 60G 43M 60G 1% /run
tmpfs 60G 0 60G 0% /sys/fs/cgroup
/dev/sda2 497M 161M 336M 33% /boot
/dev/fs1 21T 160M 21T 1% /gpfs/fs1
/dev/fs2 41T 160M 41T 1% /gpfs/fs2
```
To display the vdisk information, run the `mmlsvdisk` command as shown in Example 3-2.

**Example 3-2   vdisks configuration**

```
[root@gssio1 ~]# mmlsvdisk
declustered block size
disk name RAID code recovery group array in KiB remarks
------------------ --------------- ------------------ ----------- ----------
rg_gssio1_hs_Data_8M_2p_1_fs1 8+2p rg_gssio1-hs DA1 8192
rg_gssio1_hs_Data_8M_3p_1_fs2 8+3p rg_gssio1-hs DA1 8192
rg_gssio1_hs_MetaData_8M_2p_1_fs1 3WayReplication rg_gssio1-hs DA1 1024
rg_gssio1_hs_MetaData_8M_3p_1_fs2 4WayReplication rg_gssio1-hs DA1 1024
rg_gssio1_hs_loghome 4WayReplication rg_gssio1-hs DA1 2048 log
rg_gssio1_hs_logtp 2WayReplication rg_gssio1-hs NVR 2048 logTip
rg_gssio1_hs_logtipbackup Unreplicated rg_gssio1-hs SSD 2048 logTipBackup
rg_gssio2_hs_Data_8M_2p_1_fs1 8+2p rg_gssio2-hs DA1 8192
rg_gssio2_hs_Data_8M_3p_1_fs2 8+3p rg_gssio2-hs DA1 8192
rg_gssio2_hs_MetaData_8M_2p_1_fs1 3WayReplication rg_gssio2-hs DA1 1024
rg_gssio2_hs_MetaData_8M_3p_1_fs2 4WayReplication rg_gssio2-hs DA1 1024
rg_gssio2_hs_loghome 4WayReplication rg_gssio2-hs DA1 2048 log
rg_gssio2_hs_logtp 2WayReplication rg_gssio2-hs NVR 2048 logTip
rg_gssio2_hs_logtipbackup Unreplicated rg_gssio2-hs SSD 2048 logTipBackup
```

- **Check the file system configuration.**
  - Use the `mmlsfs` command to check the file system configuration. This command is run on one of the cluster nodes. If the management server node is not part of the cluster, Secure Shell (SSH) to one of the cluster nodes. Run: `mmlsfs all`.

- **Mounting the file system.**
  - Mount the file system by using the `mmmount` command (where `device` is the name of the file system): `mmmount device -a`.

- **Testing the file system.**
  - Use the `gpfsperf` script to run some basic I/O tests on the file system to measure the performance of the file system using a variety of I/O patterns. The `gpfsperf` script is included with Spectrum Scale. To run a basic I/O test by first sequentially creating a file, run this command:
    ```
    /usr/lpp/mmfs/samples/perf/gpfsperf create seq /gpfs/gpfs0/testfile1 -n 200G -r 16M -th 4
    ```

- **Adding nodes to the cluster.**
  - The management server node and additional I/O server nodes can be added to the Elastic Storage Server cluster by using the `gssaddnode` command. The management server node is updated with the required RPMs during deployment and prepared to join the cluster if needed. You can update the management server node by running the commands as shown in Example 3-3.

**Example 3-3   Adding nodes to the cluster commands**

```
updatenode ManagementServerNodeName -P gss_updatenode
reboot
updatenode ManagementServerNodeName -P gss_ofed
reboot
```
The I/O server nodes must be deployed properly and the high-speed network configured before `gssaddnode` can be used to add these nodes to the Elastic Storage Server cluster. The `gssaddnode` command adds the nodes to the cluster, runs the product license acceptance tool, configures the nodes (using `gssServerConfig.sh` or `gssClientConfig.sh`), and updates the host adapter, enclosure, and drive firmware. Do not use `gssaddnode` to add non-Elastic Storage Server (I/O server or management server) nodes to the cluster. Use `mmaddnode` instead.

On the `gssaddnode` command, the `-N ADD-NODE-LIST` option specifies the list of nodes that are being added. For the management server node, it is that node's host name. The `--nodetype` option specifies the type of node that is being added. For the management server node, the value is `ems`. This command must run on the management server node when that node is being added. This command can be also used to add I/O server nodes to an existing cluster.

### 3.7 Elastic Storage Server ordering

The Elastic Storage Server solution, whether it is evaluated or intended to be ordered, automatically engages a technical delivery assessment (TDA), which is performed by IBM. The assessment’s purpose is to meet the customer requirements, evaluating the customer’s current IT environment in order to help that the proposed solution meets IBM specifications and provides the client with the best possible implementation.

Depending on the required model, the Elastic Storage Server can be ordered in a minimum configuration with elastic storage nodes, EMS, drawers, and disks as long as the HMC and network devices exists in the current IT environment. This scenario is recommended when another Elastic Storage Server exists in the current IT environment.

The recommended setup is to have a dedicated HMC both for the Elastic Storage Server storage nodes and platform console and also a dedicated switch for private HMC network communication. Usually, the Elastic Storage Server systems are delivered in a T42 rack, with an HMC for the first Elastic Storage Server system, rack-mounted Flat Panel console kit, and network switches for public and internal communication either for HMC network and internal Elastic Storage Server system network communication. The networks could also be provided by the customer as dedicated switches or configured VLANs.

Figure 3-1 on page 43 shows two Elastic Storage Servers GS6 occupying 32U into one rack IBM 7014-T42 with network components and also an Elastic Storage Server GL6.
Figure 3-1  2 x Elastic Storage Servers GS6 and an Elastic Storage Server GL6 rack
3.8 Elastic Storage Server client integration

The Elastic Storage Server server access methods are similar to the ones for accessing an IBM Spectrum Scale cluster. Depending on the required configuration, the Elastic Storage Server can be accessed as an IBM Spectrum Scale Client, with a specific connector, or through dedicated protocol node servers.

When cNFS, Protocol Node services, the AFM gateway nodes, or the ISKLM key server nodes are required, one or more other servers should be configured for providing these services, running IBM Spectrum Scale at the necessary level for support, and must not run on the Elastic Storage Server server nodes.

3.9 Monitoring

This section presents some of the monitoring features based on the Elastic Storage Server GUI and command-line commands.

The Elastic Storage Server GUI is installed automatically during the Elastic Storage Server deployment process. The Elastic Storage Server GUI provides several monitoring functions.

The monitoring → System view
In this view, you can see the status of all servers and enclosures and the drives in an enclosure. The servers and enclosures are displayed by using images that represent the way they are physically placed within racks. Click a server or enclosure to view the details of that particular hardware component.

From this view, you run a procedure to replace one or more failed drives within an enclosure.

The monitoring → System details view
This view displays hardware-related information for all servers and enclosures and the drives in an enclosure. In addition, this view displays many more hardware-related details, such as the status of subhardware components: Processors and fans, for example. This view also provides a text search and filtering for unhealthy hardware.

The monitoring → Events view
This view displays events. From this view, you can monitor and troubleshoot errors in your system. The status icons can help you to quickly determine whether the event is informational, a warning, or an error.

Click an event and select the Properties action to see detailed information about the event. The event table displays the most recent events first. The status icons become gray if an error or warning has been fixed or if an informational message has been marked as read. The Mark as read action can be used on informational messages only.
The monitoring → Performance view
Use the Performance view to monitor the performance of file systems and block storage. The charting area allows panning (click and drag the chart or the timeline control at the bottom) and zooming (by using the mouse wheel or resizing the timeline control). Two charts can be displayed side-by-side in order to compare different objects and metrics or different time periods of the same chart. The timelines of the two charts can be linked together by using the Link button on the right side of the window. In addition, when the charts display two compatible metrics, you can synchronize the Y-axis as well by using the Settings fly out at the bottom of the toolbar.

The monitoring → Capacity view
You can monitor the capacity of the file system, file sets, users, and user groups.

For each file system, the capacity is divided into the total disk capacity available, used disk capacity, and free disk capacity. You can view the capacity trend for a selected file system over a 30-day or 12-month time period. You can also view all file systems at the same time. Select one or more of the listed file systems to view them in a graph.

You can also view the capacity of file sets in a file system, based on the nine largest file sets in the file system, by displaying the data by percentage. For this view, the fileset capacity as reported by the Spectrum Scale quota capability is displayed.

You can also use the `mmlsrecoverygroup`, `mmlspdisk`, and `mmpmon` commands for monitoring. The IBM Spectrum Scale RAID event log is visible by using the `mmlsrecoverygroupevents` command as shown in Example 3-4.

Example 3-4   Monitoring commands
[root@c55f02n01 ~]# mmlsrecoverygroup

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>recovery group</td>
<td>vdisks</td>
<td>vdisks</td>
<td>servers</td>
</tr>
<tr>
<td>---------------</td>
<td>--------</td>
<td>--------</td>
<td>------------</td>
</tr>
<tr>
<td>rgL</td>
<td>3</td>
<td>5</td>
<td>c55f02n01.gpfs.net,c55f02n02.gpfs.net</td>
</tr>
<tr>
<td>rgR</td>
<td>3</td>
<td>5</td>
<td>c55f02n02.gpfs.net,c55f02n01.gpfs.net</td>
</tr>
</tbody>
</table>

[root@c55f02n01 ~]# mmlsrecoverygroupevents
mmlsrecoverygroupevents: Missing arguments.
Usage:
  mmlsrecoverygroupevents RecoveryGroupName [-T] [--days days]
  [--long-term CEWID] [--short-term CEWID]
[root@c55f02n01 ~]# mmlsrecoverygroupevents rgL --days 1
Sat Oct 7 17:35:47.250 2015 c55f02n01 ST [I] Start scrubbing tracks of da_DA1_rgL.
Sat Oct 7 17:34:39.344 2015 c55f02n01 ST [I] End readmitting 1/3-degraded tracks of da_DA1_rgL.
Sat Oct 7 17:34:39.309 2015 c55f02n01 ST [I] Start readmitting 1/3-degraded tracks of loghome_rgL.
Sat Oct 7 17:34:39.233 2015 c55f02n01 ST [I] Finished repairing RGD/VCD in RG rgL.
Sat Oct 7 17:34:38.067 2015 c55f02n01 ST [I] Start repairing RGD/VCD in RG rgL.
Sat Oct 7 17:34:38.067 2015 c55f02n01 ST [I] Finished repairing RGD/VCD in RG rgL.

..........................<<snippet>>............................
[root@c55f02n01 ~]# mmlspdisk
mmlspdisk: Missing arguments.
Usage:
mmlspdisk {all | RecoveryGroupName
    [--declustered-array DeclusteredArrayName | --pdisk PdiskName]}
    [--not-in-use | --not-ok | --replace]
[root@c55f02n01 ~]# mmlspdisk rgL --pdisk "n2s01"
pdisk:
    replacementPriority = 1000
    name = "n2s01"
    device = "//c55f02n01/dev/sda10"
    recoveryGroup = "rgL"
    declusteredArray = "NVR"
    state = "ok"
    capacity = 2088763392
    freeSpace = 1904214016
    fru = ""
    location = ""
    WWN = ""
    server = "c55f02n01 gpfs.net"
    reads = 764
    writes = 114482633
    bytesReadInGiB = 1.381
    bytesWrittenInGiB = 678.420
    IOErrors = 18
    IOTimeouts = 1
    mediaErrors = 0
    checksumErrors = 0
    pathErrors = 0
    relativePerformance = 1.000
    dataBadness = 0.000
    rgIndex = 2
    userLocation = ""
    userCondition = "normal"
    hardware = "IBM IPR-10  68C08900  "
    hardwareType = NVRAM
Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this paper.

IBM Redbook

The following IBM Redbook publication provides additional information about the topic in this document.

*IBM Spectrum Scale (formerly GPFS), SG24-8254*

You can search for, view, download or order this document and other Redbooks, Redpapers, Web Docs, draft and additional materials, at the following website:

[ibm.com/redbooks](http://ibm.com/redbooks)

Online resources

These websites are also relevant as further information sources:

- IBM Spectrum Scale 4.1.1 FAQ
  [http://ibm.co/1JH0huE](http://ibm.co/1JH0huE)
- IBM Spectrum Scale Wiki
  [http://ibm.co/laKwTp5](http://ibm.co/laKwTp5)
- IBM Spectrum Scale Administration and Programming information
  [http://ibm.co/1hf1EtZ](http://ibm.co/1hf1EtZ)
- IBM Elastic Storage Server
- POWER code matrix
- IBM Spectrum Scale RAID 4.1.0.8 Administration

Help from IBM

IBM Support and downloads
[ibm.com/support](http://ibm.com/support)

IBM Global Services
[ibm.com/services](http://ibm.com/services)