SAP HANA and ESS
A Winning Combination

Olaf Weiser
## Contents

<table>
<thead>
<tr>
<th>Notices</th>
<th>vi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trademarks</td>
<td>vi</td>
</tr>
<tr>
<td>Preface</td>
<td>vii</td>
</tr>
<tr>
<td>Authors</td>
<td>vii</td>
</tr>
<tr>
<td>Now you can become a published author, too!</td>
<td>vii</td>
</tr>
<tr>
<td>Comments welcome</td>
<td>ix</td>
</tr>
<tr>
<td>Stay connected to IBM Redbooks</td>
<td>ix</td>
</tr>
</tbody>
</table>

**Chapter 1. Introduction to SAP HANA and IBM Elastic Storage Server**  
1.1 Landscape overview  
1.2 Introduction to ESS  
1.3 Software components  

**Chapter 2. Cluster networking**  
2.1 Overview  
2.2 Summary  

**Chapter 3. Understanding IBM Spectrum Scale RAID components.**  
3.1 Recovery Group  
3.2 RAID Code, VDisk, and declustered array  
3.3 IBM Spectrum Scale RAID: Fast writes (NVR)  
3.3.1 Fundamental considerations  
3.4 IBM Spectrum Scale RAID: Hot spare and disk failures  
3.4.1 Hot spares in IBM Spectrum Scale RAID  
3.4.2 Disk failure  
3.4.3 Second disk failure  
3.4.4 Spare space  

**Chapter 4. IBM Spectrum Scale adjustments.**  
4.1 Overview  
4.1.1 Server-side settings  
4.1.2 Client-side settings  
4.2 IBM Spectrum Scale parameters  
4.2.1 DirectIO in IBM Spectrum Scale  
4.2.2 ignorePrefetchLUNCOUNT  
4.3 Performance numbers  
4.3.1 Single client performance  
4.3.2 SAP HANA HWCCT test.  

**Chapter 5. IBM Spectrum Scale customization for HANA.**  
5.1 Overview  
5.2 File system and VDisk layout  
5.2.1 Internal VDisks  
5.2.2 Log file system  
5.2.3 Data file system  
5.2.4 Shared file system  
5.2.5 Creating filesets  

**Chapter 6. Summary**  

© Copyright IBM Corp. 2017. All rights reserved.
Appendix A. Sample configuration for bonding on SUSE Linux Enterprise Server . 35
Appendix B. Loghome configuration of an ESS building block ......................... 37
Appendix C. Calculating maximum capacity of a DA ................................. 39
Appendix D. Example of file system setup and HANA installation ............... 41
Appendix E. Side aware configuration examples ................................. 45
Related publications ................................................................. 49
IBM Redbooks ........................................................................... 49
Other publications ....................................................................... 49
Online resources ........................................................................... 49
Help from IBM .............................................................................. 50
Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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, MD-NC119, Armonk, NY 10504-1785, US

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 jurisdictions 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 provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

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.

Statements regarding IBM’s future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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 actual people or business enterprises 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. The sample programs are provided “AS IS”, without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.
Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

<table>
<thead>
<tr>
<th>Redbooks (logo) ®</th>
<th>IBM Spectrum™</th>
<th>POWER8®</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPFS™</td>
<td>IBM Spectrum Scale™</td>
<td>PowerVM®</td>
</tr>
<tr>
<td>IBM®</td>
<td>POWER®</td>
<td>Redbooks®</td>
</tr>
<tr>
<td>IBM Elastic Storage™</td>
<td>Power Systems™</td>
<td>Redpaper™</td>
</tr>
</tbody>
</table>

The following terms are trademarks of other companies:

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

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

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

SAP HANA on IBM® POWER® is an established HANA solution with which customers can run HANA-based analytic and business applications on a flexible IBM Power® based infrastructure. IT assets, such as servers, storage, and skills and operation procedures, can easily be used and reused instead of enforcing more investment into dedicated SAP HANA only appliances.

In this scenario, IBM Spectrum Scale™ as the underlying block storage and files system adds further benefits to this solution stack to take advantage of scale effects, higher availability, simplification, and performance.

With the IBM Elastic Storage™ Server (ESS) based on IBM Spectrum Scale, RAID capabilities are added to the file system. By using the intelligent internal logic of the IBM Spectrum™ Scale RAID code, reasonable performance and significant disk failure recovery improvements are achieved.

This IBM Redpaper™ publication focuses on the benefits and advantages of implementing a HANA solution on top of IBM Spectrum Scale storage file system.

This paper is intended to help experienced administrators and IT specialists to plan and set up an IBM Spectrum Scale cluster and configure an ESS for SAP HANA workloads. It provides important tips and best practices about how to manage IBM Spectrum Scale's availability and performance.

If you are familiar with ESS, IBM Spectrum Scale, and IBM Spectrum Scale RAID, and you need only the pertinent documentation about how to configure a IBM Spectrum Scale cluster with an ESS for SAP HANA, see Chapter 5, “IBM Spectrum Scale customization for HANA” on page 27.

Before reading this IBM Redpaper publication, you should be familiar with the basic concepts of IBM Spectrum Scale and IBM Spectrum Scale RAID. For more information, see the following resources:

- Elastic Storage Server (ESS) topic at the IBM Knowledge Center
- IBM Spectrum Scale 4.2 topic at the IBM Knowledge Center

This IBM Redpaper publication can be helpful for architects and specialists who are planning a SAP HANA on POWER deployment with the IBM Spectrum Scale file system. For more information about planning considerations for Power, see the SAP HANA on Power Planning Guide.
Authors

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

**Olaf Weiser** joined IBM as a seasoned professional over 8 years ago and has worked in the DACH TSS team since then delivering Power-based solutions to enterprise and HPC customers. He has developed deep skills in IBM Spectrum Scale (previously IBM General Parallel File System [IBM GPFS™]) and has a worldwide reputation as the performance optimization specialist for IBM GPFS™ outside development and research. At the IBM European Storage Competence Center (ESCC), Olaf is working on Advanced Technical Support (ATS) and Lab Services and Skill Enablement tasks as required to grow our IBM Spectrum Scale business in EMEA.

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

Sven Oehme
Frank Schmuck
Ralph Becker

**IBM Research and Development, San Jose**

Alexander Saupp
Nils Haustein
Achim Christ
Markus Fehling
Indulis Bernstein
Michael Murtagh
Sandeep Naik
Sandeep R. Patil

**IBM Systems**

Larry Coyne

**IBM Digital Business Group**

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](http://ibm.com/redbooks/residencies.html)
Comments welcome

Your comments are important to us!

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

- Use the online Contact us review Redbooks form found at:
  ibm.com/redbooks
- Send your comments in an email to:
  redbooks@us.ibm.com
- Mail your comments to:
  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- Find us on Facebook:
  http://www.facebook.com/IBMRedbooks
- Follow us on Twitter:
  http://twitter.com/ibmredbooks
- Look for us on LinkedIn:
  http://www.linkedin.com/groups?home=&gid=2130806
- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:
- Stay current on recent Redbooks publications with RSS Feeds:
  http://www.redbooks.ibm.com/rss.html
Introduction to SAP HANA and IBM Elastic Storage Server

In this chapter, we present an overview of SAP HANA, IBM Elastic Storage Server, and the recommended software levels. Although this IBM Redpaper publication describes SAP HANA with ESS, IBM Spectrum Scale supports other deployment modes for SAP HANA, which are based on storage rich server (leveraging IBM Spectrum Scale File Placement Optimizer [FPO] based deployment) or over traditional SAN based Block Storage.

This chapter includes the following topics:

- 1.1, “Landscape overview” on page 2
- 1.2, “Introduction to ESS” on page 3
- 1.3, “Software components” on page 5
1.1 Landscape overview

A typical SAP HANA environment consists of the HANA nodes, which are mostly machines with a large amount of memory, at least one network for TCP/IP communication, and optionally (but common) a storage area network (SAN) to the storage servers or disk back ends.

With IBM Spectrum Scale, IBM Elastic Storage Server (ESS) as the disk back end, and IBM POWER8® server technology, the design landscape can be made to be flexible. With IBM PowerVM® virtualization on the server side and the advantages of the IBM Spectrum Scale file system on the storage side, nearly endless scaling can be achieved.

IBM Spectrum Scale introduces various features and scaling options for IOPS and bandwidth without requiring architectural changes to the environment, as are required in other conventional storage designs.

At the time of this writing, SAP and IBM supports running up to eight HANA database instances for production purposes on a physical power machine in parallel with any other combination of further “non production” SAP DB virtual machines. You also can add and operate any number of SAP application servers on the same physical hardware.

IBM Spectrum Scale as a file system adds high bandwidth, data replication, and parallel access without the need for a complex SAN infrastructure.

The flexible combination of these components allows side awareness and high availability to the landscape. A high-level overview of the IBM SAP HANA solution stack is shown in Figure 1-1.

![IBM SAP HANA solution stack](image)

*Figure 1-1 IBM SAP HANA solution stack*
1.2 Introduction to ESS

The IBM ESS is a modern implementation of software-defined storage that is built on the IBM Spectrum Scale. This technology combines the CPU and I/O capability of the IBM POWER8® architecture and matches it with 2U and 4U storage enclosures. This architecture permits the IBM Spectrum Scale RAID software capability to actively manage all RAID functionality that was accomplished by a hardware disk controller.

Newly developed RAID techniques from IBM use this CPU and I/O power to help overcome the limitations of current disk drive technology. They also simplify your transition to a multi-tier storage architecture by employing solid-state flash technology and robotic tape libraries.

ESS is designed for performance. Storing PetaBytes of data is meaningless unless it can be accessed and analyzed quickly. Sustained streaming performance of data can reach 20 GBps and more in each building block, growing as more blocks are added to a configuration.

By combining the superior data movement capability of IBM Power Systems™ servers with the enhanced I/O subsystem that was introduced in the POWER8 processor and adding the disk management capability of the Power server driven Native RAID technology, a complete storage solution can be deployed without traditional storage controllers acting as a bottleneck to overall system performance.

With support for multiple 10 GbE, 40 GbE, and 100 GbE and multiple InfiniBand ports speeds of up to 100 Gb per second (EDR speed), Elastic Storage Servers provides the architecture to deliver improved data throughput.

An 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.
At the time of this writing, all available model combinations are shown in Figure 1-2. Other hardware vendors and disk enclosures, flash technologies, and disk models will be added in the future.

**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

![ESS models overview](image)

In addition to the models that are shown in Figure 1-2, 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.

For more information about updates to the 100 GbE or EDR IB adapter that are based on Mellanox ConnectX-4 cards, see the Elastic Storage Server (ESS) topic of the IBM Knowledge Center website.
1.3 Software components

Power Servers include IBM PowerVM®, which provides a secure and scalable server virtualization environment for Linux applications that are built on the advanced RAS features and leading performance of the Power Systems platform. It is maintained as part of the power server’s firmware.

Although the operating system has flexibility of running virtual machines (VMs), which are sometimes called LPARs, we highly recommend SUSE Linux Enterprise Server for our SAP HANA scenario. In early function, performance, and verification tests, SUSE Linux Enterprise Server 11 SP4 BE was used. Meanwhile, SUSE Linux Enterprise Server 12 for running SAP applications on power was released.

For more information about the latest software releases, see the SUSE website.

For more information about deploying IBM Spectrum Scale in the VMs (IBM provides the self-extracting software package), see the IBM Spectrum Scale Frequently Asked Questions and Answers topic at the IBM Knowledge Center website.

As delivered, the ESS storage server nodes include Linux (RHEL), IBM Spectrum Scale, and IBM Spectrum Scale RAID installed and ready for final configuration (network, file system parameters, and so on).

The software minimum requirements are listed in Table 1-1.

Table 1-1 Software requirements

<table>
<thead>
<tr>
<th>Component</th>
<th>Minimum release level</th>
<th>More information</th>
</tr>
</thead>
<tbody>
<tr>
<td>powerVM</td>
<td>85x</td>
<td>IBM Support Fix Central</td>
</tr>
<tr>
<td>SUSE Linux Enterprise Server</td>
<td>SUSE Linux Enterprise Server11 SP4 or SUSE Linux Enterprise Server 12 SP1</td>
<td>SUSE website</td>
</tr>
<tr>
<td>IBM Spectrum Scale</td>
<td>4.2.0.4</td>
<td>IBM Spectrum Scale Frequently Asked Questions and Answers topic at the IBM Knowledge Center website</td>
</tr>
<tr>
<td>Elastic Storage Server</td>
<td>4.5.0</td>
<td>IBM Spectrum Scale RAID Overview topic at the IBM Knowledge Center website</td>
</tr>
</tbody>
</table>
Cluster networking

In this chapter, we describe the cluster networking considerations to take and which network topology is required to meet the minimum requirements in terms of throughput and latency to and from an IBM Elastic Storage Server (ESS).

This chapter includes the following topics:

- 2.1, “Overview” on page 8
- 2.2, “Summary” on page 9
2.1 Overview

IBM Spectrum Scale supports various network scenarios. A typical high-level architecture overview is shown in Figure 2-1.

Because POWER8 and ESS storage servers can easily use 10 GbE network capabilities, the infrastructure can be enhanced by adding networks or InfiniBand connectivity. An SAP HANA setup is shown in Figure 1-1 on page 2.

Although the minimum configuration requires at least one TCP-based network, you can add an InfiniBand network to the environment and connect all or a subset of nodes directly by InfiniBand host channel adapters (HCAs).

For planning purposes, if you do not configure Remote Direct Memory Access (RDMA) over InfiniBand, IBM Spectrum Scale relies on Ethernet only. Therefore, all data is transferred by a TCP/IP socket-based communication between HANA node (client) and storage server. To enhance the bandwidth of your network connectivity, you might consider bonding multiple devices. However, the theoretical possible bandwidth a node can reach is still limited by the number of sockets a client opens to the storage servers.

In IBM Spectrum Scale, a so-called NSD client has one open socket for daemon communication to each corresponding NSD server, which provides access to the block storage. Therefore, the number of open network sockets scales by the number of NSD servers.
For an IBM Spectrum Scale environment, it does not make sense to configure more than two network ports into a bonding device on the client side when your IBM Spectrum Scale Cluster runs with one ESS building block that consists of a pair of NSD servers. The Linux TCP bonding layer cannot scale higher than two ports.

2.2 Summary

In a 10 GbE network topology with a single building block (ESS), the maximum theoretical bandwidth per client cannot exceed the bandwidth of two sockets, which provides a throughput of approximately 2 GBps. In comparison within a 40 GbE network, you can scale up to 8 GBps.

For all GL4 and GL6 models, we recommend that you consider RDMA/InfiniBand or a 40 GbE or 100 GbE topology. Otherwise, the performance benefits from an ESS building block are limited by the connectivity between NSD server and clients.

However, the minimum requirements by SAP for operating an ESS solution for SAP HANA still can be met with a standard 10 GbE network infrastructure.

For more information about a sample configuration for configuring a bond device on SUSE Linux Enterprise Server, see Appendix A, “Sample configuration for bonding on SUSE Linux Enterprise Server” on page 35.
In this chapter, we describe some basic components of IBM Spectrum Scale RAID for a better understanding of how IBM's software RAID is implemented in IBM Spectrum Scale. IBM Spectrum Scale RAID is a software implementation of storage RAID technologies within IBM Spectrum Scale.

By using conventional dual-ported disk or solid-state drives in a JBOD configuration, IBM Spectrum Scale RAID implements sophisticated data placement and error-correction algorithms to deliver high levels of storage reliability, availability, and performance.

In this chapter, we focus on an essential subset of IBM Spectrum Scale RAID components only.

For more information about IBM Spectrum Scale RAID and its components, see IBM Spectrum Scale RAID administration guide.

This chapter includes the following topics:

- 3.1, “Recovery Group” on page 12
- 3.2, “RAID Code, VDisk, and declustered array” on page 12
- 3.3, “IBM Spectrum Scale RAID: Fast writes (NVR)” on page 14
- 3.4, “IBM Spectrum Scale RAID: Hot spare and disk failures” on page 16
3.1 Recovery Group

A Recovery Group (RG) is a set of nodes that can access the same set of disks. Within an IBM Elastic Storage Server (ESS), two RGs are configured by default, of which half of all physical disk drives are assigned to it. Each ESS head node is responsible for one RG as primary server and as backup for the other RG and vice versa.

All drives are SAS twin tailed, which are connected to both head nodes. The control of an RG can be failed or taken over by the other node because of maintenance or failure situations.

3.2 RAID Code, VDisk, and declustered array

IBM Spectrum Scale RAID supports 2- and 3-fault-tolerant Reed-Solomon codes and 2-, 3-, and 4-way replication. These configurations detect and correct up to one, two, or three concurrent faults, depending on the chosen RAID level. The redundancy code layouts that IBM Spectrum Scale RAID supports is also known as tracks and maps to one block that is inside the IBM Spectrum Scale file system (see Figure 3-1).

The IBM Spectrum Scale RAID code allocates the needed space for the RAID tracks from specific sets of disk. This set of disk is called declustered array (DA). The number of physical disk drives (PDisk) that belong to the same DA is configurable in IBM Spectrum Scale RAID.
However, with models G(S,L)(1, 2, 4, 6), a fixed number of PDisks is available. Therefore, the IBM Spectrum Scale RAID deployment procedure reflects the possible choices according to your hardware model. A VDisk consists of many, wildly distributed, RAID tracks. Therefore, IBM Spectrum Scale RAID allocates the RAID tracks for a VDisk within one DA.

An important positive effect of having a higher distribution (you get with a single DA) reduces the likeliness of being critically affected from multiple physical disk failures. The configuration scenario for a single DA is shown in Figure 3-2.

![Figure 3-2](image)

**Single DA per RG**

- All vdisk tracks are allocated within 174 drives
- Total capacity of this vdisk is limited to 174 drives
  (reduced by RAID factor and spare space)
- Better performance
- Critical RAID tracks(8+2p), in case of two disk failures:
  \[
  \frac{10}{174^4 \times 9} \times 173 \, \% = 0.298 \, \% \quad \text{(critical)}
  \]

A VDisk is a logical construct of allocated space with a certain RAID level. The amount of space, which is allocated for each full RAID track, must be specified and reflect the block size for which this VDisk is used.

For example, for a targeted block size of 16 MB and a RAID level of 8+2p, a VDisk is created with a RAID segment size of approximately 2 MB. Then, a full RAID track allocates approximately 10 x 2 MB.

IBM Spectrum Scale RAID code adds a check sum trailer and a version number to each write to protect against lost writes and silent data corruption. A regular NSD device is created on top of the VDisk, which contains the characteristics, such as disk usage (dataOnly and MetaDataOnly), failure group, and storage pool.
An overview is shown in Figure 3-3.

Figure 3-3  Overview IBM Spectrum Scale RAID configuration layers

3.3 IBM Spectrum Scale RAID: Fast writes (NVR)

The fast-write IO path is one of the major recent enhancements in IBM Spectrum Scale/ESS. It is essential for the performance improvement of small I/O writes. The clipping level of the I/Os are considered small and eligible for fast writes, so they are configurable.

3.3.1 Fundamental considerations

The number of IOPS in an ESS is limited by the number of physical drives. The highest bandwidth is achieved only if large block sizes (8 MB or 16 MB) are used, which leads to I/O sizes down to the PDisk of 1 MB or 2 MB, according to the chosen RAID level \(8+2\).[2, 3]p.

I/O sizes up to 2 MB can be handled by all NL-SAS disks types without breaking data into smaller fractions. The ESS default deployment procedure pre-configures the appropriate OS (RHEL) settings that adjust the needed values for the kernel and devices.

**Note:** It is not recommended to configure VDisk with n-WayReplication for block sizes that are larger than 2 MB.

For VDisks with \(8+[2, 3]p\) RAID level, a minimum block size of 512 Kb is required.
The use of a large block size for good throughput performance and high bandwidth on the one side can generate much overhead for workloads with small random I/Os on the other side. Even worse is when I/O is done with DIRECT_IO /O_SYNC.

Writing smaller fractions of data than the block size to disk generates the need to read the corresponding VDisk track from disk into memory to modify the data and write back the VDisk track, which is known as Read-Modify-Write (RMW).

With the introduction of so-called “fast writes” in IBM Spectrum Scale RAID, RMW can be avoided completely or at least significantly reduced to gain overall performance in environments with small I/O workloads or mixed workloads.

How fast writes (IBM Spectrum Scale RAID) work is shown in Figure 3-4.

As shown on the right side of Figure 3-4, small writes are written to the so-called logtip, which is a VDisk in a special DA that is named NVR. Thee logtip is a two-way replicated VDisk that is configured by default up on disk partitions from each node from its internal hard disk drives. The internal drives do not take the I/O. The key is that these devices are connected to RAID controller with a reasonable amount of NVRAM so that all the I/Os can be satisfied by the cache. IBM Spectrum Scale RAID then mirrors the inflight I/O across the two head nodes.
The I/Os from the clients can immediately be acknowledged, as they are written successfully to the logtip. The information that is written by logtip is all kept in the cache (NVR of the SAS RAID adapter) of both head nodes. The IBM Spectrum Scale RAID codes then take the content from the logtip and writes it into loghome, which is a four-way replicated VDisk in the DA1 to free the space in the logtip without the need to immediately write down to the final VDisk tracks. By having the data safely staged down to disk, IBM Spectrum Scale RAID can hold them for a longer period.

By using this configuration, IBM Spectrum Scale RAID can collect and coalesce many small I/Os into bigger portions of data, intentionally trying to reach a full track write to physical disk. During normal operation, IBM Spectrum Scale RAID never reads from loghome (all data is still kept in the pagepool).

If the reserved space runs out of buffer, IBM Spectrum Scale RAID must flush down data to the final VDisk track and it is that enough data is available to do full track writes or IBM Spectrum Scale RAID must fetch in data first and do RMW to write down changes to disk. Therefore, the greater the loghome and pagepool are to store lots of small writes, the more likely RMW can be avoided completely. Also, a greater loghome means more allocated space among all the physical disk drives in the DA1, which improves performance.

**Tip:** A loghome VDisk with a greater size than the default of 20 GB can improve performance for small writes. Loghome size must be fixed during the initial installation and cannot be changed after other VDisks are created in the RG. Therefore, this configuration must be done during the initial deployment of the ESS. For more information about checking loghome size, see Appendix B, “Loghome configuration of an ESS building block” on page 37.

During normal operation, IBM Spectrum Scale RAID never reads from loghome, so all data must be kept in the pagepool. Therefore, having sufficient amount of pagepool depends on the total amount of memory, which is assembled in the I/O head nodes. Configuring the ESS models with 256 GB further improve the small write performance by having the chance to use greater pagepool sizes.

**Tip:** Configure your ESS models with 256 GB internal memory for better performance with small I/O workloads.

### 3.4 IBM Spectrum Scale RAID: Hot spare and disk failures

In this section, we describe hot spare and disk failures in IBM Spectrum Scale RAID.

#### 3.4.1 Hot spares in IBM Spectrum Scale RAID

In general, the IBM Spectrum Scale RAID distributes the logical capacity of the hot spare widely among all drives. Therefore, no physical reserved disks are used; only its capacity is reserved. By default, the deployment procedure configures two theoretical hot spares every 58 disks. Depending on the overall configuration that is used (single or multiple DA) a specific amount of spare space is configured. The number of spares can be customized according to your needs.
To verify the number of hot spares, use the `mmlsrecoverygroup` command and review the spares column that is shown in Example 3-1. The first value is the number of spares. In our example, six spares are defined by using the rule of two spares every 58 drives. The second value is 89 and represents how many drives out of the 174 drives that would need to be lost before affecting the VDisk information. If you use every available capacity for VDisks, IBM Spectrum Scale RAID still reserves a capacity of six drives for spare, which means this space cannot be used for creating VDisks.

Example 3-1 Use the mmlsrecoverygroup command to review the spares settings

```
[root@p8n06 ~]# mmlsrecoverygroup ess02_L -L
```

<table>
<thead>
<tr>
<th>recovery group</th>
<th>declustered arrays</th>
<th>VDisks</th>
<th>pdisks</th>
<th>current format version</th>
<th>allowable format version</th>
</tr>
</thead>
<tbody>
<tr>
<td>ess02_L</td>
<td>3</td>
<td>7</td>
<td>179</td>
<td>4.2.0.1</td>
<td>4.2.2.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>declustered array</th>
<th>needs</th>
<th>VDisks</th>
<th>pdisks</th>
<th>spares</th>
<th>replace threshold</th>
<th>free space</th>
<th>scrub duration</th>
<th>task</th>
<th>background activity</th>
<th>activity</th>
<th>priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSD</td>
<td>no</td>
<td>0</td>
<td>3</td>
<td>0,0</td>
<td>1</td>
<td>558 GiB</td>
<td>14 days</td>
<td>inactive</td>
<td>0%</td>
<td>low</td>
<td></td>
</tr>
<tr>
<td>NVR</td>
<td>no</td>
<td>1</td>
<td>2</td>
<td>0,0</td>
<td>1</td>
<td>3632 MiB</td>
<td>14 days</td>
<td>scrub</td>
<td>91%</td>
<td>low</td>
<td></td>
</tr>
<tr>
<td>DA1</td>
<td>no</td>
<td>6</td>
<td>174</td>
<td>6,89</td>
<td>2</td>
<td>62 TiB</td>
<td>14 days</td>
<td>scrub</td>
<td>71%</td>
<td>low</td>
<td></td>
</tr>
</tbody>
</table>

### 3.4.2 Disk failure

In this section, we use an example that includes an 8+2p configured VDisk (a two-fault tolerant configuration scenario). A disk becomes less than perfectly usable for the following reasons:

- The administrator might remove it by using the `mmdelpdisk` command (unlikely).
- The IBM Spectrum Scale RAID disk hospital might find that the disk is not functioning and sets the systemDraining state flag.

In this scenario, one PDisk is becoming generically draining. The state of that DA changes from scrub to rebuild-1r because we are rebuilding something that has only one redundancy missing. Also, the state of the data VDisks changes from OK to 1/2-deg because they are degraded by missing one disk out of a fault tolerance of two disks.

The rebuild process moves slowly and often finishes in a day, but the process can take as much as a few days, depending on how full the DA is, whether the data on the VDisks was ever written, how fast the CPU and the disks are, and the intensity of the foreground workload.

IBM Spectrum Scale RAID rebuilds the data onto spare space, which is distributed on all other disks. How much data depends on the VDisk (capacity utilization) and the RAID level. For our example (8+2P data VDisks), 174 disks are used (single DA in a GL6), of which one PDisk is draining.

Each track of the VDisks is spread over 10 disks (8+2p); therefore, each track has a 10/174 chance of having a single fault. The amount of affected data per VDisk is 10/174 x VDisk size. According to the RAID level, only 8/10 really is data (the rest is parity) and only one of these segments must be rebuilt. (For more information about this calculation, see Appendix C, “Calculating maximum capacity of a DA” on page 39.) The total amount of data to rebuild in the DA is the sum over all VDisks, individually according to its RAID level (fault tolerance).
3.4.3 Second disk failure

The first disk failure that was described in this chapter is easy to explain and handle. It is rebuilt, faster or slower, and then spare space is available and, in a fully utilized (space) DA, one fewer PDiskss. Therefore, a second disk fails. We are using RAID 8+2p only (that is, codes that tolerate at least two failures), so we are not yet in a scenario where data loss is ensured.

The second disk failed soon after the first failure while the rebuilding process of the first failure was still in progress. Therefore, some tracks still had a single fault. Now, some tracks likely feature a double fault, while many more new affected tracks have a single fault.

Because we are using a two-fault tolerant code in our example, we are now “critical”, with some tracks having two faults or having no redundancy at all. After we become critical, the rebuild accomplishes two things: It rebuilds only those tracks that are critical, and it runs at much higher speeds.

The DA state in mmlsrecoverygroup shows as “rebuild-critical”, with a high priority and the state of the VDisks most likely is at first critical.

As all the critical tracks are rebuilt, the state of the VDisks changes back to 1/2-deg, which indicates that they still have many single faults. Most likely, the critical part of rebuild takes several minutes only.

The amount of affected data can be estimated as shown in Figure 3-2 on page 13 and described in Appendix C, “Calculating maximum capacity of a DA” on page 39. The rest of rebuild happens as usual and in the same way as a single disk fault. Accordingly, the same rules apply for an 8+3p or n-Way replication, only dependently from the specific fault tolerance when the highest number of faults by disk failures is reached.

3.4.4 Spare space

If enough spare space is available, IBM Spectrum Scale RAID always rebuilds all VDisk tracks back to its intended fault toleration (RAID level) in case of physical disk failures. Also, it is possible that the user configured the system so that the available space in the DA is not used by VDisks. In that case, rebuild can use deallocated space to rebuild; therefore, deallocated DA data space can temporarily act as spare space.

However, you might be faced with running out of spare space because poor administration or delayed disk replacement can occur.

IBM Spectrum Scale RAID never reduces a healthy, perfect VDisk track and therefore lower its fault tolerance to repair a critical track. So, the rebuilding process stops working until a new, usable disk is available.

When the replacement disk is inserted and activated (for example, by using the mmchcarrier or mmaddpdisk --replace commands), a disk's worth of spare space is available and the rebuild process proceeds.
**Note:** A PDisk can be removed for replacement only. If all data was removed (including metadata), verify the status of the PDisk by using the `mmlspdisk` command and see that `drained` is set in the PDisk state. That is, we must wait until all the data from a `draining` disk is rebuilt elsewhere because any data that is still on the disk might be useful for finishing the rebuild.

Also, any PDisk in the dead state can be replaced immediately (even if data is allocated on them but is unreadable) because there is no expectation that dead disks can be readable again.
IBM Spectrum Scale adjustments

In addition to VDisk and file system settings, the SAP workload requires some specific tuning parameters in the cluster configuration. In this chapter, we describe some of the tuning parameters.

This chapter includes the following topics:

- 4.1, “Overview” on page 22
- 4.2, “IBM Spectrum Scale parameters” on page 23
- 4.3, “Performance numbers” on page 24
4.1 Overview

In this section, we describe several server and client settings to consider.

4.1.1 Server-side settings

Most parameters on the server side (the ESS I/O nodes) include the default deployment procedure. However, by adding memory to the machine and increasing the loghome capabilities, some of those parameters must be adjusted, as shown in Example 4-1.

Example 4-1 Configuration changes

```
mmchconfig nsdRAIDFlusherFWLogLimitMB=60k,-N gss_ppc64
mmchconfig nsdRAIDFlusherFWLogHighWatermarkMB=30k -N gss_ppc64
mmchconfig nsdRAIDFastWriteFSMetadataLimit=1m -N gss_ppc64
mmchconfig nsdRAIDFastWriteFSDataLimit=2m -N gss_ppc64
```

4.1.2 Client-side settings

A similar procedure applies for the client nodes. In addition to the ESS head nodes, you must check that the appropriate gssclientconfig script was applied. Because client nodes can be dynamically added and removed from a cluster, there is no guarantee that the correct clients settings are implemented by the default deployment procedure.

To ease the process of adding and removing clients, it is recommended to create node classes and configure the client settings (which are deployed by the sample script) on this node classes. New clients then receive their settings by ordering them into the correct node class. For more information, see the IBM Spectrum Scale documentation for node classes.

A sample script for the minimum recommend ESS clients tuning is shown in Example 4-2.

Example 4-2 Script for minimum recommended ESS clients tuning

```
[root@gssio1 gss]# cd /usr/lpp/mmfs/samples/gss/
[root@gssio1 gss]# ll
 total 24
-rwxr-xr-x 1 root root 7817 Jul 26 15:20 gssClientConfig.sh
```

Because HANA nodes feature an unusually large amount of memory, adjust the pagepool after the client configuration is applied. This adjustment is necessary because the clientCinfig script is using an internal heuristic to calculate the pagepool from the available memory.

In addition to these default settings, you must adjust other settings, such as the setting that are shown in Example 4-3 (the commands are split into single lines for better text formatting). The settings can be deployed all the same time.

Example 4-3 Adjusting default settings

```
mmchconfig maxMBpS=2000,maxGeneralThreads=2048 -N hananode
mmchconfig numaMemoryInterleave=yes,verbsRdmaMinBytes=8k -N hananode
mmchconfig verbsRdmaSend=yes,verbsRdmasPerConnection=128 -N hananode
mmchconfig verbsSendBufferMemoryMB=1024,nsdInlineWriteMax=4k -N hananode
mmchconfig aioWorkerThreads=256 -N hananode
mmchconfig disableDIO=yes,aioSyncDelay=10 -N hananode
```
4.2 IBM Spectrum Scale parameters

This IBM Redbooks publication is not intended to describe all of the various IBM Spectrum Scale parameters. However, some parameters are described in this section.

4.2.1 DirectIO in IBM Spectrum Scale

Even if DirectIO (DIO) is indicated, the file system is always allowed to ignore the DIO option and execute a read/write as a normal, buffered I/O. Instances can occur in which we must use buffered I/O instead of DIO, regardless of which configuration parameters are set; for example, if a read/write is not aligned on sector boundaries (although a correctly written application should always read/write on sector boundaries). Another example is when DIO is used to write a new file (rather than an update-in-place of an existing file) or when writing to a sparse file. In this case, the normal DIO path cannot be used because disk space must be allocated before anything can be written.

According to the Portable Operating System Interface (POSIX) definition, there is no requirement that data is written through to disk unless the application specifies O_SYNC. However, some UNIX systems traditionally interpreted O_DIRECT to imply O_SYNC and there are applications that rely on this behavior.

Therefore, IBM Spectrum Scale implements the same semantics. This implementation is done by implicitly performing a fsync at the end of each DIO write if the write was executed as buffered I/O instead of DIO, regardless of why it was done (as though the application specified O_SYNC in addition to O_DIO).

Therefore, if DIO is disabled by using the disableDIO option, data is still written through to disk, and the application receives the same semantics as without this option.

The HANA workload frequently forced DIO operation. However, IBM Spectrum Scale needs occasionally to switch to buffered mode+sync, depending on the conditions.

Some non-trivial overhead exists for switching between DIO and buffered mode; therefore it is better in many cases to stay in buffered mode for some specific types of workload.

With the disableDIO=yes,aioSyncDelay=10 setting on the client, we adjust IBM Spectrum Scale to stay in buffered mode and fsync the data for any operation, which is called with DIO.

4.2.2 ignorePrefetchLUNCOUNT

This client parameter controls how many threads the IBM Spectrum Scale deamon awakes for write behind or pre-fetching. An old internal heuristic is used for calculating and starting threads, depending on the number of network shared disks (NSDs). With IBM Spectrum Scale RAID, the number of NSDs is small, so we must advise IBM Spectrum Scale to use all available threads that are derived by cluster configuration.

The ignorePrefetchLUNCOUNT tells the NSD client to not limit the number of requests that are based on the number of visible LUNs (as they can have many physical disks behind them). Rather, it indicates that the maximum number of buffers and pre-fetch threads is limited.
The default of this parameter is no(0); however, the default is set automatically after the gssclient config script is started.

You can check that the parameter is set correctly on each NSD client by using the command that is shown in Example 4-4.

Example 4-4  Checking parameter setting

```
[root@ems1 ~]# mmlsconfig | grep -i ignorepre
ignorePre fetchLUNCount yes
[root@ems1 ~]#
```

### 4.3 Performance numbers

A performance test and verification environment is shown in Figure 4-1. The numbers are achieved from a model GL6 that was deployed with ESS 4.5.1 code level. It is recommended that gpfsperf is used to verify your setup.

As shown in Figure 4-1, the ESS nodes are connected by 4 x InfiniBand FDR, the clients by 2 x FDR, and IBM Spectrum Scale code level was used on the client side 4.2.0.4. The numbers that are shown in Example 4-5 are real measured numbers that were achieved in a customer setup.

The NSD client machines are all virtual machines (VMs or LPARs) on a power8 E880 model with at least four cores each and 32 GB memory for IBM Spectrum Scale pagepool.

Example 4-5  Multiple clients, write

```
root@ems1 # mmdsh -N beer0200g,beer0201g,beer0202g,beer0203g,beer0205g,beer0206g,beer0207g,beer0204g,beer0208g "gpfsperf create seq /gpfs/test/data/$(hostname)/100Gfile -n 100g -r 16m -th 12 -fsync" | grep "Data rate"
beer0206g:   Data rate was 2925860.09 Kbytes/sec, thread utilization 0.771, bytesTransferred 107374182400
beer0201g:   Data rate was 2889809.46 Kbytes/sec, thread utilization 0.749, bytesTransferred 107374182400
beer0202g:   Data rate was 2888886.65 Kbytes/sec, thread utilization 0.770, bytesTransferred 107374182400
beer0203g:   Data rate was 2863675.27 Kbytes/sec, thread utilization 0.766, bytesTransferred 107374182400
beer0205g:   Data rate was 2859437.49 Kbytes/sec, thread utilization 0.771, bytesTransferred 107374182400
beer0200g:   Data rate was 2767664.24 Kbytes/sec, thread utilization 0.766, bytesTransferred 107374182400
beer0207g:   Data rate was 2738951.66 Kbytes/sec, thread utilization 0.867, bytesTransferred 107374182400
beer0204g:   Data rate was 2340173.58 Kbytes/sec, thread utilization 0.917, bytesTransferred 107374182400
```
As you can see in the read performance that is shown in Example 4-6, we are approaching the theoretical overall SAS bandwidth of the building block, which is 3 SAS adapters x 4 ports (12 Gbps) ~ 36 GBps.

Example 4-6  Multiple clients, read

```
[root@rb3i0001 hwcct]# mmdsh -N beer0200g,beer0201g,beer0202g,beer0203g,beer0205g,beer0206g,beer0207g "gpfsperf read seq /gpfs/test/data/$(hostname)/100Gfile -n 100g -r 16m -th 12 -fsync" | grep "Data rate"
beer0200g:      Data rate was 4779483.20 Kbytes/sec, thread utilization 0.968, bytesTransferred 107374182400
beer0203g:      Data rate was 4428156.11 Kbytes/sec, thread utilization 0.973, bytesTransferred 107374182400
beer0206g:      Data rate was 4419566.91 Kbytes/sec, thread utilization 0.980, bytesTransferred 107374182400
beer0205g:      Data rate was 4413607.93 Kbytes/sec, thread utilization 0.972, bytesTransferred 107374182400
beer0202g:      Data rate was 4409906.75 Kbytes/sec, thread utilization 0.985, bytesTransferred 107374182400
beer0201g:      Data rate was 4408141.93 Kbytes/sec, thread utilization 0.982, bytesTransferred 107374182400
beer0207g:      Data rate was 4408088.04 Kbytes/sec, thread utilization 0.984, bytesTransferred 107374182400
```

~ 31,2 Gbytes/s

4.3.1 Single client performance

For a HANA environment, the single client performance is essential for recovery or the time it takes to load data from disk into HANADB.

A rough test scenario is shown in Example 4-7, which demonstrates IBM Spectrum Scale's single client performance of about 10 GBps read performance. For more information about the hardware setup, see Figure 4-1 on page 24.

Example 4-7  Test scenario

```
beer0201 [data] # gpfsperf read seq /gpfs/test/data/tmp1/file100g -n 100g -r 8m -th 8 -fsync
gpfsperf read seq /gpfs/test/data/tmp1/file100g
recSize 8M nBytes 100G fileSize 100G
nProcesses 1 nThreadsPerProcess 8
file cache flushed before test
not using direct I/O
offsets accessed will cycle through the same file segment
not using shared memory buffer
not releasing byte-range token after open
fsync at end of test
Data rate was 10318827.72 Kbytes/sec, thread utilization 0.806,
bytesTransferred 107374182400
```
4.3.2 SAP HANA HWCCT test

Although the ESS model was certified with eight productive HANA DB instances, an ESS can outperform this certified value by more than 50%. If all of the customized settings are configured correctly, you can achieve high numbers with the SAP test tool `hwcct`, which is included with the HANA distribution.

For more information about HWCCT, see the SAP HANA Tailored Data Center Integration - Frequently Asked Questions page of the SAP website.

A summary of the results is shown in Figure 4-2.

The results show a test with 12 HANA nodes on a power8 E880 machine in parallel to one ESS GL6 building block, which is connected by InfiniBand FDR. In the summary chart, the first column describes the workload regarding log (sequential) or random (data), the second column references the various I/O sizes from the HWCCT, and the third column lists the expected minimum level.

The measured performance numbers by SAP's HWCCT for each client is listed in the rest of the table.
IBM Spectrum Scale customization for HANA

In this chapter, we describe how to customize your IBM Spectrum Scale for HANA, including the file system and VDisk layout.

This chapter includes the following topics:

- 5.1, “Overview” on page 28
- 5.2, “File system and VDisk layout” on page 28
5.1 Overview

The following levels of changes are needed to configure an optimized environment for SAP HANA workloads:

- Adjust the ESS’s default VDisk configuration by creating at least three file systems.
  
  Because of the different IO workload behaviors of SAP HANA, we create one file system for the log workload and one file system for data workload.
  
  During the tests (which are eligible for any customer environment), you can share data and log file system among many Security Identifiers (SIDs).
  
  Optionally, you can consider creating filesets to separate the SIDs from each other.
  
  A third file system for the SAP named “shared” includes no special recommendations and can be created without any special recommendations.
  
  For more information about creating a shared file system, see Appendix E, “Side aware configuration examples” on page 45.

- After setting up the file system and ESS adjusted VDisk layout, you must set some specific parameters in your cluster configuration.

The minimum normally required settings are not described further here because they are set by the default deployment procedure. This document focuses only on the other parameters to change for an optimized SAP HANA environment.

5.2 File system and VDisk layout

During the installation and set up of your HANA environment, you can customize your file system and IBM Spectrum Scale device names to your own naming conventions (see Figure 5-1 on page 29); for example, how to set up and configure the IBM Spectrum Scale file systems and as recommend by SAP¹.

¹ For more information, see the Recommended File System Layout page of the SAP website.
5.2.1 Internal VDisks

Now we must adjust a pre-configured ESS (default deployment) for an SAP HANA workload. Depending on the chosen model, you can proceed with the default settings, which are deployed by using the default deployment procedure. With the default settings, you do not scale enough to exceed more than four HANA databases in parallel. If you want to exceed this limit, you must increase the size of the VDisk loghome to 100 GB (the default is 20 GB).

Increasing the size of VDisk loghome is not possible and the loghome VDisk cannot be re-created if there are any other customer VDisks available. A prepared sample file is found in the common /usr/lpp/mmfs/samples/vdisk directory.

Rename the sample and edit the file according to the example that is shown in Example 5-1.

```
Example 5-1 Edited file
[root@is38san1a vdisk]# cat vdisk.stanza.ini
# Recovery group hanaL
#   NVR
%vdisk: vdiskName=hanal_ltip rg=hanal da=NVR size=48m blocksize=2m raidCode=2WayReplication diskUsage=vdiskLogTip
#   SSD
#     create log tip backup vdisk on a single SSD
%vdisk: vdiskName=hanal_ltbakup rg=hanal da=SSD size=48m blocksize=2m raidCode=Unreplicated
diskUsage=vdiskLogTipBackup
#   DA1
%vdisk: vdiskName=hanal_lhome rg=hanal da=DA1 size=100g blocksize=2m raidCode=4WayReplication
diskUsage=vdiskLog
longTermEventLogSize=4m shortTermEventLogSize=4m fastWriteLogPct=90 diskUsage=vdiskLog
# Recovery group hanaR
%vdisk: vdiskName=hanar_ltip rg=hanar da=NVR size=48m blocksize=2m raidCode=2WayReplication
diskUsage=vdiskLogTip
%vdisk: vdiskName=hanar_ltbakup rg=hanar da=SSD size=48m blocksize=2m raidCode=Unreplicated
diskUsage=vdiskLogTipBackup
%vdisk: vdiskName=hanar_lhome rg=hanar da=DA1 size=100g blocksize=2m raidCode=4WayReplication
diskUsage=vdiskLog
```
A larger loghome is always good to use for better performance. The more loghome that is available, the more pagepool is need. If your loghome is configured to “large” in terms of existing pagepool and buffer sizes, it is not fully used.

**Tip:** Because increasing loghome results in wiping out all VDisks, you should reconfigure your ESS for HANA workloads before the first file systems gets created.

Create your VDisks for the ESS Building block by using the `mmcrvdisk -F yourfilename` command. The Spectrum Scale daemons must be running to create the VDisk.

### 5.2.2 Log file system

As described in 5.2.1, “Internal VDisks” on page 29, create a VDisk stanza file and edit it according to your naming conventions (see Example 5-2). If you are creating a highly available or site aware configuration, add a `failureGroup` identifier for each VDisk in the file. The `fg` must be the same within a building block when you want to use IBM Spectrum Scale replication later on and mirror your data between two ESS building blocks. If you have more than one building block per site, the failure group identifier must be the same over all of the ESSs in the same site. For more information about a highly available scenario configuration example, see Appendix E, “Side aware configuration examples” on page 45.

**Example 5-2   Create and edit a VDisk stanza file**

```
[root@is38san1a vdisk]# cat vdisk.stanza.logfs
%vdisk: vdiskName=hanaLM1 rg=hanaL da=DA1 blocksize=1m size=50g raidCode=4WayReplication diskUsage=metadataOnly
%vdisk: vdiskName=hanaLD1 rg=hanaL da=DA1 blocksize=1m size=200g raidCode=8+2p diskUsage=dataOnly pool=datapool

%vdisk: vdiskName=hanaRM1 rg=hanaR da=DA1 blocksize=1m size=50g raidCode=4WayReplication diskUsage=metadataOnly
%vdisk: vdiskName=hanaRD1 rg=hanaR da=DA1 blocksize=1m size=200g raidCode=8+2p diskUsage=dataOnly pool=datapool
```

Create the VDisk by using the `mmcrvdisk` command. Create the NSDs by using the `mmcrnsd` command.

Next, create the file system with the parameters that are shown in Example 5-3 (in this example, without IBM Spectrum Scale replication enabled by default).

**Example 5-3   Creating the file system**

```
mmcrfs hanalog -F vdisk.stanza.logfs -B 1M --metadata-block-size 1M -M 2 -R 2 -m 1 -r 1 -L 256M -T /hana/log -E no -j scatter -S relatime
```

For more information about a replicated scenario, see Appendix E, “Side aware configuration examples” on page 45.
5.2.3 Data file system

As described in 5.2.2, “Log file system” on page 30, create the data file system (see Example 5-4).

**Example 5-4 Creating the data file system**

```bash
[root@is38san1a vdisk]# cat vdisk.stanza.datafs
%vdisk: vdiskName=hanaLDFT2M1 rg=hanaL da=DA1 blocksize=1m size=200g raidCode=4WayReplication diskUsage=metadataOnly
%vdisk: vdiskName=hanaLDFT2D1 rg=hanaL da=DA1 blocksize=16m size=500g raidCode=8+2p diskUsage=dataOnly pool=datapool
%vdisk: vdiskName=hanaRDFT2M1 rg=hanaR da=DA1 blocksize=1m size=200g raidCode=4WayReplication diskUsage=metadataOnly
%vdisk: vdiskName=hanaRDFT2D1 rg=hanaR da=DA1 blocksize=16m size=500g raidCode=8+2p diskUsage=dataOnly pool=datapool

Create the VDisk by using the `mmcrvdisk` command. Create the databases by using the `mmcrnsd` command.

Then, create the file system with the parameters that are shown in Example 5-5 (in this example, without IBM Spectrum Scale replication enabled by default).

**Example 5-5 Creating the file system**

```bash
mmcrfs hanadata -F vdisk.stanza.datafs -B 16M --metadata-block-size 1M -M 2 -R 2 -m 1 -r 1 -L 256M -T /hana/data -E no -j scatter -S relatime

For more information about a replicated scenario, see Appendix E, “Side aware configuration examples” on page 45.

5.2.4 Shared file system

As described in 5.2.3, “Data file system” on page 31, create a VDisk stanza file (see Example 5-6).

**Example 5-6 Creating a VDisk stanza**

```bash
[root@ems1 vdisk]# cat vdisk.stanza.sharedfs
%vdisk: vdiskName=rg_gssioS1M1 rg=rg_gssio1 da=DA1 blocksize=1m size=5g raidCode=4WayReplication diskUsage=metadataOnly
%vdisk: vdiskName=rg_gssioS1D1 rg=rg_gssio1 da=DA1 blocksize=4m size=100g raidCode=8+2p diskUsage=dataOnly pool=datapool
%vdisk: vdiskName=rg_gssioS2M1 rg=rg_gssio2 da=DA1 blocksize=1m size=5g raidCode=4WayReplication diskUsage=metadataOnly
%vdisk: vdiskName=rg_gssioS2D1 rg=rg_gssio2 da=DA1 blocksize=4m size=100g raidCode=8+2p diskUsage=dataOnly pool=datapool
[root@ems1 vdisk]#

Next, create the file system with the parameters that are shown in Example 5-7 (without IBM Spectrum Scale replication enabled by default).

**Example 5-7 Creating the file system**

```bash
[root@ems1 vdisk]# mmcrfs hanashared -F vdisk.stanza.sharedfs -B 4M --metadata-block-size 1M -M 2 -R 2 -m 1 -r 1 -L 256M -T /hana/shared -E no -j scatter -S relatime

For more information about a replicated scenario, see Appendix E, “Side aware configuration examples” on page 45.
5.2.5 Creating filesets

You can create an own set of file systems (log, home, and shared) for each SAP instance (SID). You also can operate in an environment with multiple SIDs sharing systems. If you plan to use multiple SIDs, we recommend creating filesets for each SID.

During the SAP verification and certification, 12 productive SIDs within the same IBM Spectrum Scale file system in parallel were tested successfully in our example.
Summary

This IBM Redbooks publication describes the powerful combination of a IBM Spectrum Scale RAID storage back end (ESS) and POWER8 servers for building an SAP environment. This combination provides the highest performance specifications, while being less complex and more flexible.

The most important IBM Spectrum Scale RAID fundamentals and components to get started with an ESS environment for SAP HANA DB workloads also were described.

If you follow the suggestions that are described in Chapter 5, “IBM Spectrum Scale customization for HANA” on page 27 and configure your environment, you should get similar performance numbers out of your configuration, as described in Chapter 4, “IBM Spectrum Scale adjustments” on page 21.
Sample configuration for bonding on SUSE Linux Enterprise Server

In this appendix, we provide a sample configuration for bonding on SUSE Linux Enterprise Server.

Remember that you must check with your network administrator that LACP settings are configured correctly.

Alternatively, you can configure your bonding device as an active-passive bond (exchange the line BONDING_MODULE). However, you cannot exceed the bandwidth of this single active port.

A so-called LACP or trunk configured bond is shown in Example A-1 on page 36.
Example A-1  Sample configuration

```
root@pils:> cat ifcfg-bond0
DEVICE='bond0'
NAME='bond0'
BONDING_MASTER='yes'
BONDING_SLAVE_0='eth3'
BONDING_SLAVE_1='eth2'
IPADDR='10.0.0.114/24'
NETMASK='255.255.255.0'
ONBOOT='yes'
BOOTPROTO='static'
BONDING_MODULE_OPTS='mode=802.3ad miimon=100 xmit_hash_policy=
layer3+4'
primary_reselect=0 fail_over_mac=2'
BONDING_SLAVE0='eth2'
BONDING_SLAVE1='eth3'
BROADCAST=''
ETHTOOL_OPTIONS=''
MTU=''
NETWORK=''
REMOTE_IPADDR=''
STARTMODE='hotplug'
USERCONTROL='no'
#
# The minimum configuration for the bonding slaves looks as
follows:
root@pils [network] # cat ifcfg-eth2
STARTMODE='hotplug'
BOOTPROTO='none'
root@pils [network] # cat ifcfg-eth3
STARTMODE='hotplug'
BOOTPROTO='none'
#```
Loghome configuration of an ESS building block

In this appendix, we provide an example of a loghome configuration of an ESS building block.

Use the following `mmlsrecoverygroup` command to verify the loghome configuration of an ESS building block:

```
[root@ems1 ~]# mmlsrecoverygroup rg_gssio1 -L
```

The result of the use of the command is shown in Example B-1.

**Example B-1  Loghome configuration verification using mmlsrecoverygroup command**

<table>
<thead>
<tr>
<th>recovery group</th>
<th>declustered arrays</th>
<th>VDisks</th>
<th>pdisks</th>
<th>format version</th>
</tr>
</thead>
<tbody>
<tr>
<td>rg_gssio1</td>
<td>3</td>
<td>11</td>
<td>61</td>
<td>4.1.0.1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>declustered array</th>
<th>needs</th>
<th>VDisks</th>
<th>pdisks</th>
<th>spares</th>
<th>replace threshold</th>
<th>free space</th>
<th>scrub</th>
<th>duration</th>
<th>task</th>
<th>progress</th>
<th>priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSD</td>
<td>no</td>
<td>1</td>
<td>1</td>
<td>0,0</td>
<td>1</td>
<td>372 GiB</td>
<td>scrub</td>
<td>14 days</td>
<td></td>
<td>20%</td>
<td>low</td>
</tr>
<tr>
<td>NVR</td>
<td>no</td>
<td>1</td>
<td>2</td>
<td>0,0</td>
<td>1</td>
<td>3648 MiB</td>
<td>scrub</td>
<td>14 days</td>
<td></td>
<td>66%</td>
<td>low</td>
</tr>
<tr>
<td>DA1</td>
<td>no</td>
<td>9</td>
<td>58</td>
<td>2,31</td>
<td>2</td>
<td>263 GiB</td>
<td>scrub</td>
<td>14 days</td>
<td></td>
<td>7%</td>
<td>low</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VDisk</th>
<th>RAID code</th>
<th>declustered array</th>
<th>VDisk size</th>
<th>block size</th>
<th>granularity</th>
<th>state</th>
<th>remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>rg_gssio1_logtip</td>
<td>2WayReplication</td>
<td>NVR</td>
<td>48 MiB</td>
<td>2 MiB</td>
<td>4096</td>
<td>ok</td>
<td>logTip</td>
</tr>
<tr>
<td>rg_gssio1_logtipbackup</td>
<td>Unreplicated</td>
<td>SSD</td>
<td>48 MiB</td>
<td>2 MiB</td>
<td>4096</td>
<td>ok</td>
<td>logTipBackup</td>
</tr>
<tr>
<td>rg_gssio1_loghome</td>
<td>4WayReplication</td>
<td>DA1</td>
<td>40 GiB</td>
<td>2 MiB</td>
<td>4096</td>
<td>ok</td>
<td>log</td>
</tr>
<tr>
<td>rg_gssio1_Data_8M_2p_1</td>
<td>8+2p</td>
<td>DA1</td>
<td>71 TiB</td>
<td>8 MiB</td>
<td>32 KiB</td>
<td>ok</td>
<td></td>
</tr>
<tr>
<td>rg_gssio1_MetaData_8M_2p_1</td>
<td>3WayReplication</td>
<td>DA1</td>
<td>3672 GiB</td>
<td>1 MiB</td>
<td>32 KiB</td>
<td>ok</td>
<td></td>
</tr>
</tbody>
</table>
Calculating maximum capacity of a DA

In this appendix, we provide an example of how the maximum capacity of a DA can be calculated.

Ignoring overhead for checksum data and some internal VDisks (for example, loghome), the total raw capacity is the mount of PDisks x the capacity per drive. For example, assume 1 TB disks are available. For a fully utilized DA, this following calculation is used:

- total capacity: $174 \times 1 \text{ TB} = 174 \text{ TB}$
- hot spares $6 \times 1 \text{ TB} = 6 \text{ TB}$
- usable for vdisk (raw) $168 \text{ TB}$
- 1 vdisk 8+2p (netto) ~ $134 \text{ TB}$

Depending on the block size of the VDisks, the amount of data per VDisk track varies. Per VDisk track, a fixed check sum trailer of 4 K is added at each PDisk segment. Therefore, the overhead can be estimated by using the following calculations (depending on RAID level and block size):

- With 8+2p and 8 MB block size, it is $40/8192 \text{ [KB]} ~ 0.5 \%$
- With 8+2p and 1 MB block size it is $40/1024 \text{ [KB]} ~ 4 \%$

Therefore, the maximum usable space for a VDisk that is build on 1 TB drives is less than 134 TB. For our example, we continue with 134 TB.

**Data affected by a single disk failure**

Taking a fully utilized DA (174 PDisks in a GL6), with a VDisk 8+2p and a capacity of 134 TB as an example, the data that is affected by a single failure is $10/174 \times 134 \text{ TB} ~ 7.8 \text{ TB}$.

Only 8/10 out of this 7.8 TB is user data; the rest is parity. Also, only one segment per VDisk track is affected and needs to rebuilt. Finally, $1/10 \times 7.8 \text{ TB} = 0.78 \text{ TB}$ must rebuilt. By having a reserved hot spare capacity of six drives, enough room is available to allocate the needed disk space for rebuilding.
Example of file system setup and HANA installation

In this appendix, we shown an example of the file system setup and HANA installation.

Creating the file system is shown in Example D-1.

**Example D-1  Create the file system**

```bash
[root@ems1 vdisk]# cat vdisk.stanza.sharedfs
%vdisk: vdiskName=rg_gssioS1M1 rg=rg_gssio1 da=DA1 blocksize=1m size=5g
    raidCode=4WayReplication diskUsage=metadataOnly
%vdisk: vdiskName=rg_gssioS1D1 rg=rg_gssio1 da=DA1 blocksize=4m size=100g
    raidCode=8+2p
diskUsage=dataOnly pool=datapool

%vdisk: vdiskName=rg_gssioS2M1 rg=rg_gssio2 da=DA1 blocksize=1m size=5g
    raidCode=4WayReplication diskUsage=metadataOnly
%vdisk: vdiskName=rg_gssioS2D1 rg=rg_gssio2 da=DA1 blocksize=4m size=100g
    raidCode=8+2p
diskUsage=dataOnly pool=datapool
[root@ems1 vdisk]# mmcrvdisk -F vdisk.stanza.sharedfs
mmcrvdisk: [I] Processing vdisk rg_gssioS1M1
mmcrvdisk: [I] Processing vdisk rg_gssioS1D1
mmcrvdisk: [I] Processing vdisk rg_gssioS2M1
mmcrvdisk: [I] Processing vdisk rg_gssioS2D1
mmcrvdisk: Propagating the cluster configuration data to all
affected nodes. This is an asynchronous process.
[root@ems1 vdisk]#
[root@ems1 vdisk]# mmcrnsd -F vdisk.stanza.sharedfs
mmcrnsd: Processing disk rg_gssioS1M1
mmcrnsd: Processing disk rg_gssioS1D1
mmcrnsd: Processing disk rg_gssioS2D1
mmcrnsd: Propagating the cluster configuration data to all
affected nodes. This is an asynchronous process.
```
[root@ems1 vdisk]#
[root@ems1 vdisk]# mmcrfs hanashared -F vdisk.stanza.sharedfs -B 4M --metadata-block-size 1M -M 2 -R 2 -m 1 -r 1 -L 256M -T /hana/shared -E no -j scatter -S relatime

The following disks of hanashared will be formatted on node gssio2.spectrum:
  rg_gssioS1M1: size 6088 MB
  rg_gssioS1D1: size 105536 MB
  rg_gssioS2M1: size 6088 MB
  rg_gssioS2D1: size 105536 MB

Formatting file system ...
Disks up to size 415 GB can be added to storage pool system.
Disks up to size 1.6 TB can be added to storage pool datapool.
Creating Inode File
Creating Allocation Maps
Creating Log Files
3 % complete on Fri Oct 21 20:52:34 2016
100 % complete on Fri Oct 21 20:52:37 2016
Clearing Inode Allocation Map
Clearing Block Allocation Map
Formatting Allocation Map for storage pool system
Formatting Allocation Map for storage pool datapool
Completed creation of file system /dev/hanashared.
mmcrfs: Propagating the cluster configuration data to all affected nodes. This is an asynchronous process.
[root@ems1 vdisk]#

After setting up your file systems and after successfully proceeding to the installation of HANA software, you should see output similar to the output that is shown in Example D-2.

Example D-2  Installing SAP HANA software

saphana1:/hana/data # df -h
Filesystem Size Used Avail Use% Mounted on
/dev/sda3 132G 3.1G 123G 3% /
udev 16G 220K 16G 1% /dev
tmpfs 16G 652K 16G 1% /dev/shm
/dev/sda1 132M 13M 120M 10% /boot/efi
/dev/hanadata 414G 3.7G 410G 1% /hana/data
/dev/hanalog 186G 2.3G 184G 2% /hana/log
/dev/hanashared 207G 8.4G 198G 5% /hana/shared

saphana1:/hana/data # find /hana/data

After setting up your file systems and after successfully proceeding to the installation of HANA software, you should see output similar to the output that is shown in Example D-2.
Appendix D. Example of file system setup and HANA installation

```
/hana/data/GER/mnt00001/nameserver.lck

saphana1:/hana/data # find /hana/log
/hana/log
/hana/log/GER
/hana/log/GER/mnt00001
/hana/log/GER/mnt00001/hdb00003
/hana/log/GER/mnt00001/hdb00003/__DO_NOT_TOUCH_FILES_IN_THIS_DIRECTORY__
/hana/log/GER/mnt00001/hdb00003/logsegment_000_00000001.dat
/hana/log/GER/mnt00001/hdb00003/logsegment_000_00000000.dat
/hana/log/GER/mnt00001/hdb00003/logsegment_000_directory.dat

/hana/log/GER/mnt00001/hdb00002
/hana/log/GER/mnt00001/hdb00002/__DO_NOT_TOUCH_FILES_IN_THIS_DIRECTORY__
/hana/log/GER/mnt00001/hdb00002/logsegment_000_00000001.dat
/hana/log/GER/mnt00001/hdb00002/logsegment_000_00000000.dat
/hana/log/GER/mnt00001/hdb00002/logsegment_000_00000002.dat
/hana/log/GER/mnt00001/hdb00002/logsegment_000_directory.dat

/hana/log/GER/mnt00001/hdb00001
/hana/log/GER/mnt00001/hdb00001/__DO_NOT_TOUCH_FILES_IN_THIS_DIRECTORY__
/hana/log/GER/mnt00001/hdb00001/logsegment_000_00000001.dat
/hana/log/GER/mnt00001/hdb00001/logsegment_000_00000000.dat
/hana/log/GER/mnt00001/hdb00001/logsegment_000_directory.dat
/hana/log/GER/mnt00001/hdb00001/landscape.id

saphana1:/hana/data # find /hana/shared | head -15
/hana/shared
/hana/shared/GER
/hana/shared/GER/HDB00
/hana/shared/GER/HDB00/HDBSettings.sh
/hana/shared/GER/HDB00/dev_rfc.trc
/hana/shared/GER/HDB00/backup
/hana/shared/GER/HDB00/backup/log
/hana/shared/GER/HDB00/backup/data
/hana/shared/GER/HDB00/exe
/hana/shared/GER/HDB00/HDBSettings.csh
/hana/shared/GER/HDB00/hdbenv.csh
/hana/shared/GER/HDB00/HDBAdmin.sh
/hana/shared/GER/HDB00/saphana1
/hana/shared/GER/HDB00/saphana1/sapprofile.ini
/hana/shared/GER/HDB00/saphana1/webdispatcher.ini
```
Side aware configuration examples

In this appendix, we show an example of a side aware configuration for a log, data, and shared file system.

An example configuration for a log file system is shown in Example E-1.

Example E-1  Configuration for log file system

[root@is38san1a vdisk]# cat vdisk.stanza.logfs
%vdisk: vdiskName=ESS1hanaLM1 rg=ESS1hanaL da=DA1 blocksize=1m size=50g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=10
%vdisk: vdiskName=ESS1hanaLD1 rg=ESS1hanaL da=DA1 blocksize=1m size=200g raidCode=8+2p diskUsage=dataOnly
  pool=datapool failureGroup=10

%vdisk: vdiskName=ESS1hanaRM1 rg=ESS1hanaR da=DA1 blocksize=1m size=50g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=20
%vdisk: vdiskName=ESS1hanaRD1 rg=ESS1hanaR da=DA1 blocksize=1m size=200g raidCode=8+2p diskUsage=dataOnly
  pool=datapool failureGroup=20

# other site

%vdisk: vdiskName=ESS2hanaLM1 rg=ESS2hanaL da=DA1 blocksize=1m size=50g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=20
%vdisk: vdiskName=ESS2hanaLD1 rg=ESS2hanaL da=DA1 blocksize=1m size=200g raidCode=8+2p diskUsage=dataOnly
  pool=datapool failureGroup=20

%vdisk: vdiskName=ESS2hanaRM1 rg=ESS2hanaR da=DA1 blocksize=1m size=50g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=20
%vdisk: vdiskName=ESS2hanaRD1 rg=ESS2hanaR da=DA1 blocksize=1m size=200g raidCode=8+2p diskUsage=dataOnly
  pool=datapool failureGroup=20

[root@is38san1a vdisk]#
An example configuration for data is shown in Example E-2.

**Example E-2  Configuration for data**

```bash
[root@is38san1a vdisk]# cat vdisk.stanza.datafs
%vdisk: vdiskName=ESS1hanaLDFT2M1 rg=ESS1hanaL da=DA1 blocksize=1m size=200g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=10
%vdisk: vdiskName=ESS1hanaLDFT2D1 rg=ESS1hanaL da=DA1 blocksize=16m size=500g raidCode=8+2p diskUsage=dataOnly
pool=datapool failureGroup=10
#
%vdisk: vdiskName=ESS1hanaRDFT2M1 rg=ESS1hanaR da=DA1 blocksize=1m size=200g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=10
%vdisk: vdiskName=ESS1hanaRDFT2D1 rg=ESS1hanaR da=DA1 blocksize=16m size=500g raidCode=8+2p diskUsage=dataOnly
pool=datapool failureGroup=10
#

#other site
%vdisk: vdiskName=ESS2hanaLDFT2M1 rg=ESS2hanaL da=DA1 blocksize=1m size=200g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=20
%vdisk: vdiskName=ESS2hanaLDFT2D1 rg=ESS2hanaL da=DA1 blocksize=16m size=500g raidCode=8+2p diskUsage=dataOnly
pool=datapool failureGroup=20
#
%vdisk: vdiskName=ESS2hanaRDFT2M1 rg=ESS2hanaR da=DA1 blocksize=1m size=200g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=20
%vdisk: vdiskName=ESS2hanaRDFT2D1 rg=ESS2hanaR da=DA1 blocksize=16m size=500g raidCode=8+2p diskUsage=dataOnly
pool=datapool failureGroup=20
```

An example configuration for shared is shown in Example E-3.

**Example E-3  Configuration for shared**

```bash
%vdisk: vdiskName=ess1L_PROD_S_MD1 rg=ess1io1g da=DA1 blocksize=1m size=100g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=10
%vdisk: vdiskName=ess1L_PROD_S_DO1 rg=ess1io1g da=DA1 blocksize=16m size=2500g raidCode=8+3p
diskUsage=dataOnly pool=datapool failureGroup=10
%vdisk: vdiskName=ess1R_PROD_S_MD1 rg=ess1io2g da=DA1 blocksize=1m size=100g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=10
%vdisk: vdiskName=ess1R_PROD_S_DO1 rg=ess1io2g da=DA1 blocksize=16m size=2500g raidCode=8+3p
diskUsage=dataOnly pool=datapool failureGroup=10
%vdisk: vdiskName=ess2L_PROD_S_MD1 rg=ess2io1g da=DA1 blocksize=1m size=100g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=20
%vdisk: vdiskName=ess2L_PROD_S_DO1 rg=ess2io1g da=DA1 blocksize=16m size=2500g raidCode=8+3p
diskUsage=dataOnly pool=datapool failureGroup=20
%vdisk: vdiskName=ess2R_PROD_S_MD1 rg=ess2io2g da=DA1 blocksize=1m size=100g raidCode=4WayReplication
diskUsage=metadataOnly failureGroup=20
%vdisk: vdiskName=ess2R_PROD_S_DO1 rg=ess2io2g da=DA1 blocksize=16m size=2500g raidCode=8+3p
diskUsage=dataOnly pool=datapool failureGroup=20
```

**Note:** Example E-3 is copied from another example; therefore, you must adjust the names to your needs accordingly.
The commands that are used for creating file systems (after `mmcrvdisk -F filename`, `mmcrnsd -F filename`) are shown in Example E-4.

**Example E-4  Commands for creating file system**

<table>
<thead>
<tr>
<th>Type</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>log</td>
<td><code>mmcrfs hanalog -F vdisk.stanza.logfs -B 1M --metadata-block-size 1M -M 2 -R 2 -m 2 -r 2 -L 256M -T /hana/log -E no -j scatter -S relatime</code></td>
</tr>
<tr>
<td>data</td>
<td><code>mmcrfs hanadata -F vdisk.stanza.datafs -B 16M --metadata-block-size 1M -M 2 -R 2 -m 2 -r 2 -L 256M -T /hana/data -E no -j scatter -S relatime</code></td>
</tr>
<tr>
<td>shared</td>
<td><code>mmcrfs hanashared -F vdisk.stanza.sharedfs -B 4M --metadata-block-size 1M -M 2 -R 2 -m 1 -r 1 -L 256M -T /hana/shared -E no -j scatter -S relatime</code></td>
</tr>
</tbody>
</table>
Related publications

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

IBM Redbooks

The IBM Redbooks publication *Introduction Guide to the IBM Elastic Storage Server*, REDP-5253, provides more information about the topic in this document. Note that this publication might be available in softcopy only.

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

ibm.com/redbooks

Other publications

The following publications are also relevant as further information sources:

- *IBM Spectrum Scale Version 4 Release 2.3 Problem Determination Guide*, GA76-0443
- *IBM Spectrum Scale Version 4 Release 2.3 Command and Programming Reference*, SA23-1456

Online resources

The following websites are also relevant as further information sources:

- IBM Elastic Storage Sever:
  http://www.ibm.com/systems/storage/spectrum/ess
- IBM Elastic Storage Sever Knowledge Center:
- IBM Spectrum Scale:
  http://www.ibm.com/systems/storage/spectrum/scale
- IBM Spectrum Scale Knowledge Center:
  https://ibm.biz/Bdinhb
- IBM Spectrum Scale Wiki:
  https://ibm.biz/BdFymB
- SAP HANA:
Help from IBM

IBM Support and downloads:
ibm.com/support

IBM Global Services
ibm.com/services