

LSFS Container Technical Description

2025

StarWind Documents



Trademarks

“StarWind”, “StarWind Software” and the StarWind and the StarWind Software logos are registered trademarks of StarWind Software. “StarWind LSFS” is a trademark of StarWind Software which may be registered in some jurisdictions. All other trademarks are owned by their respective owners.

Changes

The material in this document is for information only and is subject to change without notice. While reasonable efforts have been made in the preparation of this document to assure its accuracy, StarWind Software assumes no liability resulting from errors or omissions in this document, or from the use of the information contained herein. StarWind Software reserves the right to make changes in the product design without reservation and without notification to its users.

Technical Support and Services

If you have questions about installing or using this software, check this and other documents first - you will find answers to most of your questions on the [Technical Papers](#) webpage or in [StarWind Forum](#). If you need further assistance, please [contact us](#).

About StarWind

StarWind is a pioneer in virtualization and a company that participated in the development of this technology from its earliest days. Now the company is among the leading vendors of software and hardware hyper-converged solutions. The company's core product is the years-proven StarWind Virtual SAN, which allows SMB and ROBO to benefit from cost-efficient hyperconverged IT infrastructure. Having earned a reputation of reliability, StarWind created a hardware product line and is actively tapping into hyperconverged and storage appliances market. In 2016, Gartner named StarWind “Cool Vendor for Compute Platforms” following the success and popularity of StarWind HyperConverged Appliance. StarWind partners with world-known companies: Microsoft, VMware, Veeam, Intel, Dell, Mellanox, Citrix, Western Digital, etc.

Copyright ©2009-2018 StarWind Software Inc.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written consent of StarWind Software.

IMPORTANT NOTE: Starting from 14869 build, the LSFS device feature was excluded and deprecated. Please remove all Asynchronous replicas and LSFS devices before the update. See more information in the release notes:
<https://www.starwindsoftware.com/release-notes-build>

Introduction

StarWind Virtual SAN[®] is a native Windows software-defined VM storage solution. It creates a VM-centric and high performing storage pool purpose-built for virtualization workloads. StarWind Virtual SAN delivers supreme performance compared to any dedicated SAN solution since it runs locally on the hypervisor. In addition, all IO is processed by local RAM/SSD caches and disks and is never bottlenecked by storage fabric. StarWind Virtual SAN includes the Log-Structuring File System technology, which coalesces small random writes, typical for virtualized environment, into the stream of big sequential writes. As a result, the performance is increased and the flash life is prolonged. This guide is intended for experienced Windows system administrators and IT professionals who would like to know more about the StarWind Virtual SAN solution and better understand how StarWind Virtual SAN works. It provides the description of the LSFS container and explains its features, i.e. deduplication, defragmentation etc., which should give the end user the necessary knowledge for its implementation into the system. Full set of up-to-date technical documentation can always be found [here](#), or by pressing the Help button in the StarWind Management Console. For any technical inquiries, please visit our [online community](#), [Frequently Asked Questions](#) page, or use the [support form](#) to contact our technical support department.

Lsfs Features Description

Lsfs In Brief:

LSFS (Log-Structured File System) is a journaling file system that keeps track of the changes that will be made in a journal. This file system keeps no data, only changes. The journal is divided into file segments for convenience. The minimum size of a file segment is 128 MB, maximum – 512 MB. LSFS always keeps one empty file-segment. The initial size of an LSFS device is equal to one empty file-segment irrespective of the device size. When data is written to the device, it grows automatically as changes arrive. Old data is kept on the file system until defragmentation cleans it up. LSFS supports built-in automatic and manual defragmentation, inline deduplication, and snapshots.

How Snapshots Work:

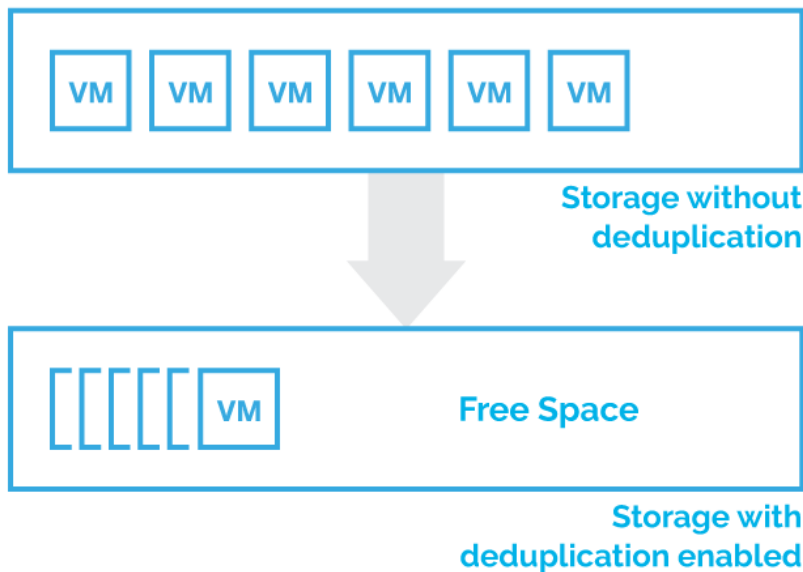
LSFS is a snapshot-based file system. Every snapshot is incremental and occupies additional space which is equal to the changes made since the previous snapshot creation. Snapshots are created every 5 minutes by default and then deleted. A snapshot can be taken manually as well. LSFS also creates restore points during its usual operation. These are the latest consistent journal parts, which will be used in case of failure. Restore points can be viewed via Snapshot Manager in Device Recovery Mode.

How Defragmentation Works:

Defragmentation works continuously in the background. Each file segment will be defragmented when data capacity exceeds the allowed value. Maximum allowed junk rate before defragmentation process is initiated equals 60%. This value can be changed using the context menu of the device. Data from the old fragmented file segment will be moved to another empty file segment and the old file will be deleted. If the available disk space on the physical storage is low, the LSFS uses more aggressive defragmentation policy and slows down access speed for the end user. If there is no space on the physical drive, LSFS enters the “read-only” mode.

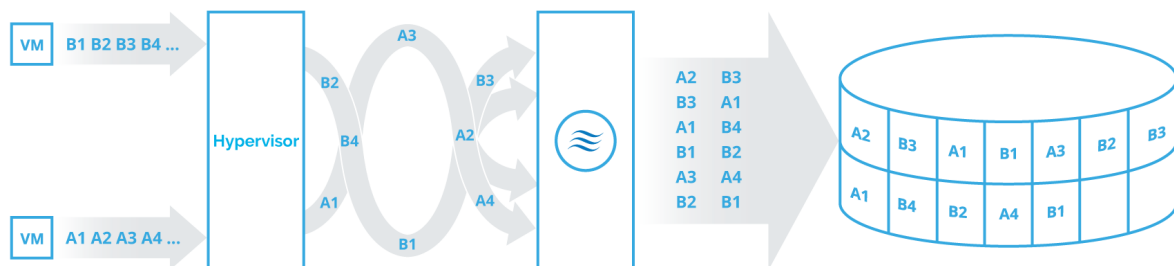
How Deduplication Works:

Unique chunks of data are identified and stored while being analyzed. As the analysis continues, other chunks are compared to the stored copy and whenever a match occurs, the redundant chunk is replaced with a reference shortcut that points to the stored chunk. Given that the same byte pattern may occur dozens, hundreds, or even thousands of times (the matching frequency depends on the chunk size), the amount of data to be stored or transferred can be reduced significantly. For example, similar VMs that are put on top of an LSFS device will be deduplicated as well, but a paging file of every VM will not. The deduplication is not applicable to pagefiles, thus the size of every pagefile will be added as unique data. Therefore, 10 similar VMs 12GB each (2GB for pagefile) occupy $(12-2)+2*10=30\text{GB}$.



Performance Boost:

The log structuring uses Redirect-on-Write for snapshotting, writes, etc. This means that every new data block is written to the next available place on the disk, organizing the data blocks sequentially. Irrespective of the access pattern used, the underlying storage always receives 4 MB blocks. LSFS coalesces multiple smaller random writes into a single sequential big write I/O. It allows achieving up to 90% raw sequential write performance at the file system level, which is by order of magnitude better compared to conventional file systems (i.e NTFS, ZFS, etc.), which is around 10%. Read patterns may vary starting from 4k blocks which eliminates performance negative impact.



Therefore, for the underlying storage, the inexpensive spinning drives can be used. The only side effect of LSFS is possible overprovisioning due to fragmentation and metadata presence.

Overprovisioning:

The junk rate predetermines maximum allowed LSFS growth (overprovisioning) compared to the declared LSFS device size. The default rate is 60%; therefore, LSFS file

segments might use 2.5 times more space than the initial LSFS size. Additionally, metadata occupies up to 20% of the initial LSFS size. Thus, overprovisioning of LSFS devices is 200%. It is possible to run full defragmentation via device context menu. The manually started defragmentation ignores default defragmentation rate and cleans up all junk blocks inside file segments. Metadata occupy additional space as well, but the space consumed by metadata is usually small. Certain patterns cause abnormal metadata growth and it may occupy as much space as the useful data. The most storage-consuming pattern is 4k random write. However, when the entire disk is full, the ratio between metadata and data itself should be stabilized, and 3-times growth is the maximum possible one. Information about the useful data count, metadata, and fragmentation can be found in StarWind Management Console when LSFS device is chosen.

Lsfs-Based Ha Device Synchronization:

An HA device based on LSFS uses snapshots for synchronization purposes. HA synchronizes only the latest changes after any failure: each HA partner has a healthy snapshot before the failure and there is no need to synchronize all the data. Therefore, only the latest changes made after the healthy snapshot creation are synchronized. Full synchronization is performed only after the initial replica creation. Even in case of full synchronization, only useful data is replicated while junk data is skipped.

Current Lsfs Requirements

Required Ram (Not Related To L1 Cache):

4.6 GB of RAM per 1 TB initial LSFS size (deduplication is disabled) 7.6 GB of RAM per 1 TB initial LSFS size (deduplication is enabled)

Overprovisioning Is 200%

LSFS files can occupy 3 times more space compared to initial LSFS size. Snapshots require additional space to store them.

Physical Block Size Of Lsfs Devices Is 4K








This means that write speed will be low if write requests are not aligned and deduplication is not performed. However, this problem is almost completely eliminated since all modern drives use 4k blocks and are compatible with all Windows OS versions

starting from Vista. Hyper-V aligns VHDX, but VHD might not be aligned. ESX isn't compatible with 4k drives, but VMFS can be aligned.

Conclusion

Log-Structured File System is a StarWind Virtual SAN feature that is useful in certain configuration scenarios where deduplication is required. LSFS significantly improves the random writes performance by coalescing them into the stream of sequential writes, additionally extending the durability of flash devices. LSFS device can be created by following the steps provided [here](#).

Contacts

US Headquarters	EMEA and APAC
 +1 617 829 44 95	 +44 2037 691 857 (United Kingdom)
 +1 617 507 58 45	 +49 800 100 68 26 (Germany)
 +1 866 790 26 46	 +34 629 03 07 17 (Spain and Portugal)
	 +33 788 60 30 06 (France)

Customer Support Portal: <https://www.starwind.com/support>

Support Forum: <https://www.starwind.com/forums>

Sales: sales@starwind.com

General Information: info@starwind.com



StarWind Software, Inc. 100 Cummings Center Suite 224-C Beverly MA 01915, USA
www.starwind.com ©2025, StarWind Software Inc. All rights reserved.