

# White Paper

## Fujitsu Server PRIMERGY

## Windows Server 2012 R2 Storage Spaces

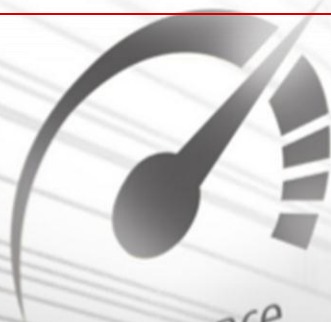
## Performance

Microsoft has introduced the Storage Spaces functionality with Microsoft Windows Server 2012. In Windows Server 2012 R2 existing functions have been optimized and new functions added. This document will give you an overview of these functions and their performance. In addition to the fundamental features and configuration approaches, basis performance values as well as comparative values for HW RAID and Windows Software RAID are mentioned.

### Version

1.0

2014-03-31



## Contents

Document history .....	2
Introduction .....	3
Storage Spaces under Windows Server 2012 and Windows Server 2012 R2 .....	3
Purpose of the document .....	4
Fujitsu products for Storage Spaces under Windows Server 2012 R2 .....	5
Functionality .....	6
Explanations .....	6
Configuration .....	15
Measurement method .....	17
Measurement environment .....	18
Performance .....	19
Storage Spaces vs. Storage Spaces R2 .....	19
Storage Spaces vs. Software-RAID vs. Hardware-RAID .....	19
Storage Spaces with SSD Write-Back Cache .....	25
Storage Spaces with Tiering .....	34
RAID Ctrl SAS 6G 5/6 1GB (D3116C) CacheCade 2.0 .....	37
CPU utilization with Storage Spaces, Software RAID and Hardware RAID .....	38
Summary and conclusion .....	39
Operations and ergonomics .....	39
Functionality .....	39
Performance .....	40
Frequently Asked Questions .....	41
Literature .....	42
Contact .....	42

## Document history

### **Version 1.0**

First version

## Introduction

### Storage Spaces under Windows Server 2012 and Windows Server 2012 R2

As part of the handling of file and storage subsystems Microsoft introduced the functionality of storage pools and Storage Spaces with Microsoft Windows Server 2012 and Microsoft Windows 8. Basically, we are dealing with the following functions here:

- Storage virtualization
- RAID functionality
- Thin provisioning (storage reservation)
- Script-based management via PowerShell
- Efficient file system repair
- Integration with cluster shared volumes (CSV)

Newly added and improved functions for Storage Spaces in Microsoft Windows Server 2012 R2 are:

Functionality	New or improved?	Description
Storage Tiering	New	Automatically stores frequently used data on the faster SSD tier and rarely used data on the slower HDD tier. Parity Spaces are not supported.
Write-back cache	New	An SSD write cache accelerates write operations.
Parity Space support for failover clusters	New	Allows the creation of parity Spaces on failover clusters.
Dual Parity Spaces	New	Double parity allows the simultaneous failure of two hard disks in a parity Space.
Automatically rebuild Storage Spaces from storage pool free Space	New	Allows a faster rebuild of the Space on the remaining memory in the pool after a hard disk failure than would be possible with an individual hot-spare hard disk.
JBOD enclosure awareness	New	Allows the use of a Space across several enclosures and thus ensures that the data can still be accessed if an entire enclosure fails.
File sync between work folders	New	Files that are stored in so-called "Sync Shares" are replicated to the devices of authorized users and are thus also accessible offline for the users.
Datadeduplication	Improved	Reduction in storage Space consumption by avoiding redundantly stored data.

The functionalities that are relevant for understanding the performance of Storage Spaces are explained in more detail in section "[Functionality](#)".

As far as Storage Spaces are concerned, the two Windows Server versions are compatible with each other to the extent that Spaces, which were created with an older version, can be used without any problems in more recent versions, but not the other way around.

## Purpose of the document

Microsoft offers with Storage Spaces a software-based product, which is to provide an extension to the known software RAID and at the same time an alternative to the frequently used hardware RAID controllers. This document is to compare the functionality and performance of the following three different approaches:

- Hardware RAID (management via RAID controller management software)
- Windows Software RAID (management via the disk management console of Windows)
- Windows Storage Spaces (further development of software RAID, implemented via Server Manager and PowerShell)

The following table shows how the known RAID levels of hardware RAID can be compared with software RAID and Windows Storage Spaces.

HW-RAID	Windows SW-RAID	Windows Storage Spaces 2012 R2	RAID description	General condition	Fail-safety
RAID 0	Striping	Simple	Striping without redundancy	None	None
RAID 1E	Not comparable	Not comparable	Striping with simple redundancy	$= 2n + 1$ ( $n > 0$ ) physical disks	1 drive failure
RAID 1 RAID 10	Mirror	Mirror 2-way	Striping with simple redundancy	$= 2$ physical disks $= n \times 2$ disks ( $n > 1$ )	1 drive failure
Not comparable	Not comparable	Mirror 3-way	Striping with double redundancy	$\geq 5$ physical disks	2 drive failures
RAID 5	RAID 5	Single Parity	Striping with simple parity redundancy	$\geq 3$ physical disks	1 drive failure
RAID 6	Not comparable	Dual Parity	Striping with double parity redundancy	$\geq 4$ physical disks	2 drive failures

Reasonable configurations are also considered and compared in order to highlight the advantages of the new functions Storage Tiering und Write-Back Cache.

## Fujitsu products for Storage Spaces under Windows Server 2012 R2

Storage Spaces can currently be used with the following SAS controllers introduced by Fujitsu.

SAS Controller	Released in system
LSI SAS 9200-8e PCIe	PRIMERGY RX100 S8
	PRIMERGY RX200 S8
	PRIMERGY RX2520 M1
	PRIMERGY RX300 S8
PSAS CP200i 6G HBA PCIe	PRIMERGY SX960
	PRIMERGY SX980 (Cib)
	PRIMERGY CX272 S1 (Cib)

In this case, the “LSI SAS 9200-8e PCIe” controller enables the connection of external disk subsystems such as ETERNUS JX40 and the “PSAS CP200i 6G HBA PCIe” controller enables the connection of internal server hard disks and SSDs. The “PSAS CP200i 6G HBA PCIe” controller is also used in the Fujitsu “Cluster-in-a-box” (Cib) solutions on the basis of PRIMERGY BX400 and CX420.

On the assumption of the same number and technology of the connected hard disks, both controllers do not differ when it comes to performance.

Please see the configurators of the systems for the possible combinations of PRIMERGYs and controllers.

Note: Some components may not be available in all countries or sales regions.

Windows Storage Spaces has special requirements of the certification of hard disks, host bus adapters and enclosures. A current list of which Fujitsu products have Storage Spaces certification from Microsoft can be seen in the [“Windows Server Catalog”](#).

# Functionality

## Explanations

### *Storage Pools, Storage Spaces, Virtual Disks*

Storage pools can use USB, SATA and SAS technology to group hard disks (now also of different size) into physical hard disk clusters (but only SAS in failover clustering). These pooled hard disks are then no longer available to classic data media management (Disk Manager).

Virtual disks (Spaces) can now be created in the defined pool by means of Storage Spaces. These virtual disks should then be used like normal drives, e.g. through the allocation of a drive letter.

It is possible to increase and use reserved areas of both a storage pool (through adding a further hard disk) and a virtual disk (through reserved storage). However, the reserved storage feature (thin provisioning) is not supported in failover cluster mode.

### *General*

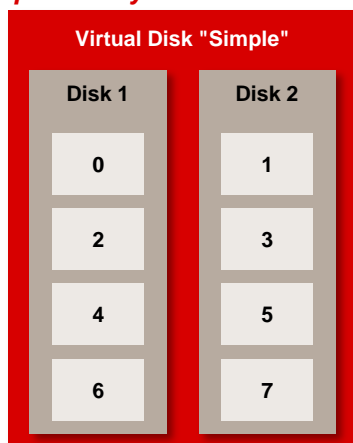
Storage Spaces reserves 256 MB for internal Spaces metadata on every hard disk in the storage pool

Consequently, there is information stored on every hard disk about the entire storage pool, to which the disk belongs. If you took hard disks from an existing pool and put them in another system, the storage pool to which they belong would automatically be displayed as well as an error message if corresponding hard disks were missing.

The following configuration limits apply for Storage Spaces in Windows Server 2012 R2:

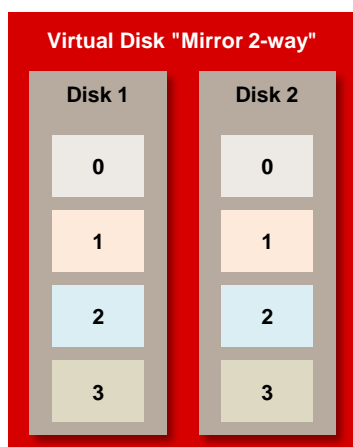
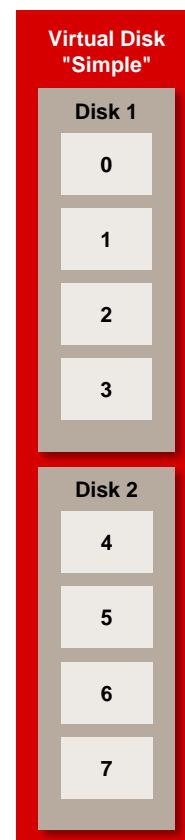
- Up to 240 physical hard disks in a storage pool (and several pools with up to 240 hard disks are possible).
- Up to 480 TB storage capacity in a storage pool.
- Up to 128 Storage Spaces in a storage pool.
- Up to 4 storage pools in a cluster configuration.
- Storage Spaces does not support boot and swap drives.

## Spaces-Layouts



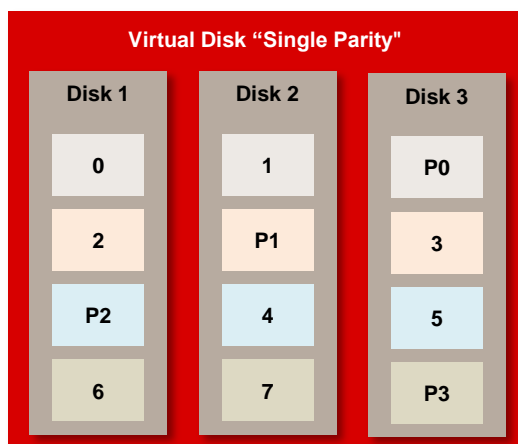
**Simple:** The data blocks (here 0 - 7) are distributed over all the hard disks for accesses via so-called columns (picture left). The more columns (with at least one hard disk) there are, the more parallel accesses are possible. The best performance is achieved with one hard disk per column.

The "same" configuration with only one column would entail losses as regards performance (picture right). The columns can be configured (only PowerShell) and will among other things be considered to some extent a little later.



**Mirror 2-way:** The data blocks (here 0 - 3) are duplicated (picture left), thus of course also reducing the capacity by half.

**Mirror 3-way:** Here the usable capacity is only one third of the available hard disk capacity, but with two data copies, which means that even two drives can fail.



**Single Parity:** The data blocks (here 0 - 7) with parity information (here P1 - P3) are distributed over all the hard disks (picture left). This option offers a good compromise between speed and fail-safety.

**Dual Parity:** Withstands two hard disk failures, because the parity information is double (on two different disks).

A certain minimum number of hard disks is required for the described storage layouts:

Storage Layout	Minimum number of hard disks	Minimum number of hard disks in the cluster
Simple	1	3
Mirror 2-way	2	3
Mirror 3-way	5	6
Single Parity	3	3
Dual Parity	7	6

## PowerShell

Some parameters can only be changed from the default via the PowerShell and cannot be modified with the GUI of the Server Manager.

An overview of the commands that can be used for Storage Spaces is provided by the command:

```
Get-Command -Module Storage
```

A list of commands with appropriate descriptions is available in the TechNet article [Storage Cmdlets in Windows PowerShell](#).

Recommendation: As many options cannot be configured and are not displayed in the graphic interface, PowerShell should be used for the configuration and administration of Storage Spaces.

## Columns

The standard number of the above mentioned columns is one per hard disk for the “Simple” layout. This applies for up to eight hard disks, above that the value remains fixed or would have to be modified via the PowerShell.

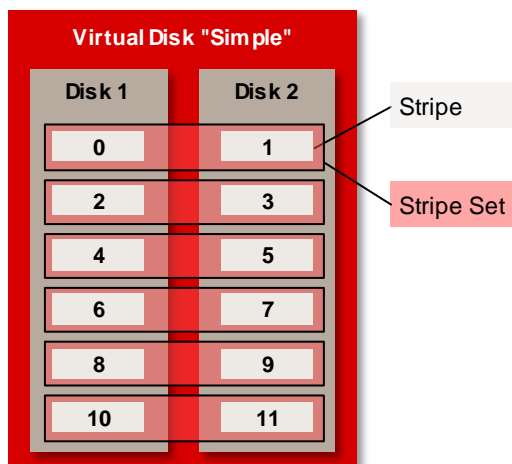
In the case of the “Single Parity” layout three columns are configured as a default setting and the “Mirror” default setting follows from the number of data copies. In other words, Mirror 2-way: *number of disks divided by 2*.

When creating a Space layout, the number of columns can be specified with the PowerShell.

## Stripe Size

The default stripe size (interleave) is 256 kB for all storage layouts.

Stripes (chunks) are sub-units of a stripe set and stripe size refers to the size of a data block that is saved on a disk. A stripe set is composed of one stripe for each data medium of a RAID array.



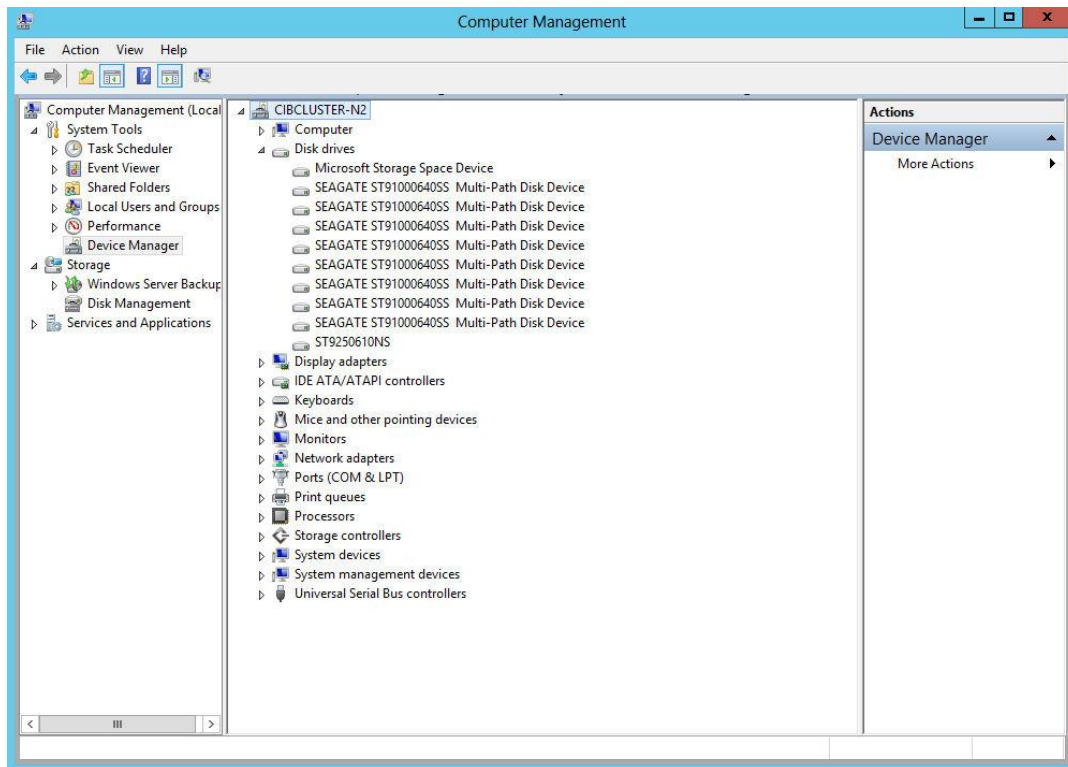
The diagram shows a RAID 0 array or a “Simple” Storage Spaces layout with two hard disks (disk 1 and disk 2) and the corresponding logical distribution of the data blocks.

## Cache settings

There are also a series of cache settings, which are shown here. In addition to the disk cache, which is relevant to the physical hard disk, there are caches for the virtual disk, which is mapped via the Microsoft Storage Space Device.

The impact of the individual caches is explained further in various test scenarios in the white paper [“Windows Storage Spaces Performance”](#) in the sections on the topic performance.

The picture shows how the disk drives are presented in the Device Manager:



The settings can now be made via <Properties> and <Policies>.



Settings for the Microsoft Storage Space Device (“Enable write caching on the device” cannot be disabled here).



Setting for the single hard disk.

For performance reasons it is recommended to enable the hard disk cache for every hard disk. All the measurements in this document were carried out with enabled hard disk caches. In order to minimize the risk of data loss during productive operation in the case of a power failure the use of a UPS (uninterruptible power supply) is strongly recommended. To be able to completely rule out the loss of data in the cache all hard disk caches would have to be disabled.

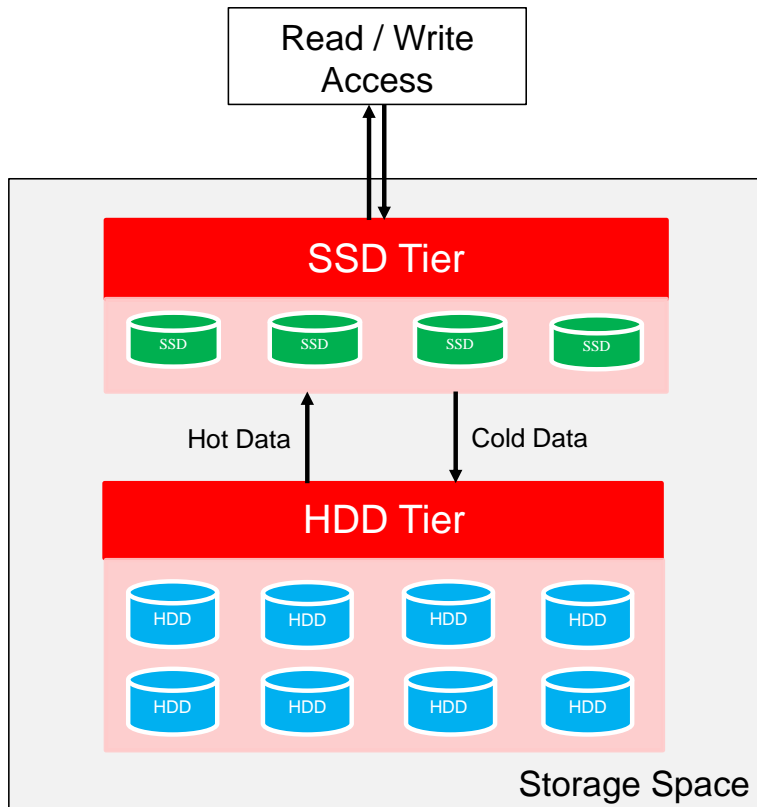
The performance influences which can result from disabling the hard disk caches are described in subsection "Disk cache" of the section "Performance data in comparison" in the white paper entitled "[Single Disk Performance](#)".

## Storage Tiering

Storage tiers are new in Windows Server 2012 R2 and combine the performance advantages of SSDs with the capacity advantages of HDDs. To this end, HDDs are logically consolidated to form a HDD tier and SSDs to form an SSD tier.

In most cases, frequently used data accounts for only 15% - 20% of the entire database and is also called a “hot spot”. As soon as a volume is created with storage tiers, Windows Server 2012 R2 continually analyzes the IO requests and keeps the most frequently used data on the fast SSD tier, while rarely used data is

moved to the HDD tier. Data that is newly added to a tiered volume is first saved on the SSD tier - assuming sufficient storage Space is available - and, if necessary, later moved to the HDD tier. This is illustrated in the diagram opposite.



The advantage of this is that a few fast SSDs can answer most of all the requests, whereas high-capacity and comparably low-priced HDDs store the remaining data in case rarely used data is nevertheless accessed. The minimum number of SSDs for simple tiered Spaces is 1 and at least two SSDs are needed for mirror tiered Spaces. The diagram opposite shows a configuration with four columns, i.e. up to four data media can be written in parallel. Both tiers must always have the identical number of columns.

Prerequisite for a gain in performance through tiering is that the access to the data has a hot spot and the size of the hot spot can be approximately estimated. An appropriate size for the SSD tier can be defined in accordance with the estimation.

Windows Server 2012 R2 decides which data is saved on which storage tier based on a separate algorithm. However, individual files can either be firmly allocated to the HDD tier or the SSD tier with the help of the PowerShell.

In addition to the advantages mentioned, the selection of storage tiers entails the following dependencies:

- Only “Simple” and “Mirror” are available as Spaces layouts, but not “Parity”.
- Only “Fixed” can be selected as the provisioning type. Thin provisioning is not possible.
- The number of columns on the HDD tier always corresponds to the number of SSD tiers and vice versa. If, for example, a “Simple” Spaces layout is created with eight HDDs and two SSDs, the volume will only have two columns. To put it more simply, both tiers have the same Spaces layout. However, a number of only two columns on the eight HDDs reduces the sequential read and write performance, for example four-fold in the “Simple” Spaces layout.
- Frequently used data on the HDD tier is only moved to the SSD tier with a delay (as standard at 1 a.m.). If this data is no longer used on the following day, the tiering would be useless. However, it is possible to define the time and frequency for running the task manually.

Below is a list of the PowerShell commands that can be used to create and manage storage tiers. Terms in pointed brackets are proper names and can be freely selected.

Create tiers in an existing storage pool "Pool1" with HDDs and SSDs:

```
Get-StoragePool <Pool1> | New-StorageTier -FriendlyName <SSDTier> -MediaType  
SSD
```

```
Get-StoragePool <Pool1> | New-StorageTier -FriendlyName <HDDTier> -MediaType  
HDD
```

Show certain fields of the tier configuration:

```
Get-StorageTier | FT FriendlyName, MediaType, Size -AutoSize
```

Create a tiered volume:

```
$SSD = Get-StorageTier -FriendlyName <SSDTier>
```

```
$HDD = Get-StorageTier -FriendlyName <HDDTier>
```

```
Get-StoragePool <Pool1> | New-VirtualDisk -FriendlyName <Space1> -  
ResiliencySettingName Simple -StorageTiers $SSD, $HDD -StorageTierSizes 8GB,  
32GB -WriteCacheSize 1GB
```

```
Get-StoragePool <Pool1> | New-VirtualDisk -FriendlyName <Space2> -  
ResiliencySettingName Mirror -StorageTiers $SSD, $HDD -StorageTierSizes 8GB,  
32GB -WriteCacheSize 1GB
```

Firmly allocate individual files and folders to either the HDD tier or the SSD tier.

The term <Path> must be replaced by a file path in the form "C:\example.txt".

```
Set-FileStorageTier -FilePath <PATH> -DesiredStorageTierFriendlyName  
Space1_SSDTier
```

```
Set-FileStorageTier -FilePath <PATH> -DesiredStorageTierFriendlyName  
Space1_HDDTier
```

The change only becomes effective if the so-called "Storage Tiers Optimization Task" is completed. As standard the task runs in the background every day at 1 a.m. The time and frequency for running the task can be determined individually.

*Control Panel → System and Security → Administrative Tools → Schedule Tasks → Storage Tiers Management*

Furthermore, the "Storage Tiers Optimization Task" can be manually initiated at any time with the following PowerShell command: The term <Drive Letter> must be replaced by the drive letter of the partition concerned.

```
Optimize-Volume -DriveLetter <Drive Letter> -TierOptimize
```

Note: Moving files between the tiers can result in losses in performance during operations. As a rule, the "Storage Tiers Optimization Task" is run with low priority and only influences the remaining performance to a small extent.

The following command can be used to cancel the allocation:

```
Clear-FileStorageTier -FilePath <PATH>
```

Show the allocation of files and folders to the tiers:

```
Get-FileStorageTier -VolumeDriveLetter <Drive Letter>
```

A detailed overview of the commands for storage tiering is provided in the following TechNet blog: ["Step-by-step for Storage Spaces Tiering in Windows Server 2012 R2"](http://blogs.technet.com/tsfujitsu/2014/03/StorageSpacesTiering.aspx)

## SSD Write-Back Cache (WBC)

The WBC is new in Windows Server 2012 R2 and is used to improve write performance by the SSDs located in the storage pool being used as a write cache. This should enable short-term write phases with a high load to be processed more quickly. High-end RAID controllers have a hardware-integrated cache, which pursues the same objective as the WBC with Storage Spaces.

The WBC can be used independently of storage tiering, but is only possible if there are SSDs in the storage pool and sufficient Space ( $\geq 32$  MB) is available on the SSDs. When creating a Space a WBC with the size of 1 GB is automatically created. If there are no SSDs in the storage pool or there is less than 32 MB free storage Space available in the pool, a WBC is not created. One exception, however, is provided by a parity layout, in which a WBC of 32 MB is always automatically created (regardless of whether there are SSDs in the storage pool or not).

The number of SSDs required for the WBC depends on the used storage layout and Microsoft recommends the following configurations:

- |                                    |                     |
|------------------------------------|---------------------|
| ■ Simple:                          | At least one SSD    |
| ■ Mirror or Single Parity:         | At least two SSDs   |
| ■ Three-Way Mirror or Dual Parity: | At least three SSDs |

Apart from one exception, the numbers stated above are also absolutely necessary. However, the WBC can also be created for “Dual Parity” and “Three-Way Mirror” layouts with two instead of three SSDs. On the one hand, this involves differences in performance and on the other hand the WBC is only protected against the failure of one SSD for a WBC with two SSDs, and thus does not provide the fail-safety of the “Dual Parity” or “Three-way Mirror” layout on the HDDs, with which up to two hard disks can fail.

The size of the WBC can only be determined when creating a Space with the PowerShell and may be between 32 MB and 100 GB. A subsequent change in the WBC is not possible. For this purpose, the Space must be deleted and recreated, which results in the data in the Space having to be temporarily saved.

Microsoft recommends a maximum WBC size of 10 GB per Space. A more precise recommendation for the size of the WBC is provided in section [“Storage Spaces with SSD Write-Back Cache”](#).

If you create a Space and have HDDs and SSDs in the storage pool, a WBC is in fact automatically created, but in addition to the WBC the capacity of the SSDs is used together with the HDDs in order to create the virtual disk. In most cases, this is probably not desired. In order to use the SSD storage capacity solely for the WBC it is necessary for the SSDs in the pool concerned to be marked as the “Journal” drive with the following PowerShell command.

```
Get-StoragePool -FriendlyName <Pool1> | Get-PhysicalDisk | ? MediaType -eq SSD | Set-PhysicalDisk -Usage Journal
```

If a virtual disk is now created, the SSDs are solely used for the write-back cache and are not incorporated in the capacity of the virtual disk.

When creating a virtual disk, the following sample PowerShell command can be used to set the size of the WBC when it differs from the standard of 1 GB:

```
New-VirtualDisk -StoragePoolFriendlyName <Pool1> -FriendlyName <Space1>  
-UseMaximumSize -ResiliencySettingName Parity -ProvisioningType Fixed  
-WriteCacheSize 10GB
```

## Journaling

Journaling was used in Windows Server 2012 for the “Parity” storage layout. In Windows Server 2012 R2 the WBC replaces journaling. If you want to use SSDs on a dedicated basis as a WBC, they are marked beforehand as a “journal” drive.

### **Parity Space support for failover clusters**

In Windows Server 2012 R2 it is possible to create single and dual parity Spaces in a failover cluster environment. This enables extended options for failsafe configurations, whereas in Windows Server 2012 only "Mirror" was possible as a failsafe Spaces layout in failover cluster environments.

### **JBOD Enclosure Awareness**

Storage Spaces offers to allocate data copies to specific JBOD enclosures for configurations that require increased fail-safety. This means that the data is mirrored via one or more enclosures. This functionality is known as enclosure awareness and means that if an enclosure fails, the data can still be accessed on one or more of the other enclosures.

If there are several JBOD enclosures available in a configuration and a Space is to be created that is to be distributed across them, thus providing protection against a complete JBOD failure, this can be done with the following parameter when creating a Space:

```
New-VirtualDisk -StoragePoolFriendlyName StoragePool -FriendlyName  
VirtualDisk1 -ResiliencySettingName Mirror -ProvisioningType Fixed -Size 10TB  
-IsEnclosureAware $True
```

### **Hot Spare disks and Pool Storage for hard disk rebuild**

Hard disks can be defined as hot spares to allow them to be used as a replacement hard disk (within an enclosure) in case a hard disk fails. Hot spare hard disks can be defined both when creating the storage pool and at any time during operation.

If a hard disk fails and hot spare hard disks are available in the enclosure concerned, Storage Spaces carries out the following actions as standard:

- Set hard disk concerned to "Failed"
- Replace the faulty hard disk with the hot spare hard disk and repeat the write operation
- If no hot spare hard disk is available, Storage Spaces waits for the hard disk to be replaced
- This behavior can be configured with the help of the PowerShell command  
*Set-StoragePool -RetireMissingPhysicalDisks <value>*

There are three setting options ("Disabled" is the standard value):

- Enabled: Set the hard disk to "Failed", replace the hard disk with hot spare hard disk and repeat the operation
- Disabled: Wait until the hard disk is connected again or an administrator intervenes. (No hot spare hard disks are used.)
- Auto: If a hot spare hard disk is available, proceed as in the Enabled case, otherwise as in the Disabled case.

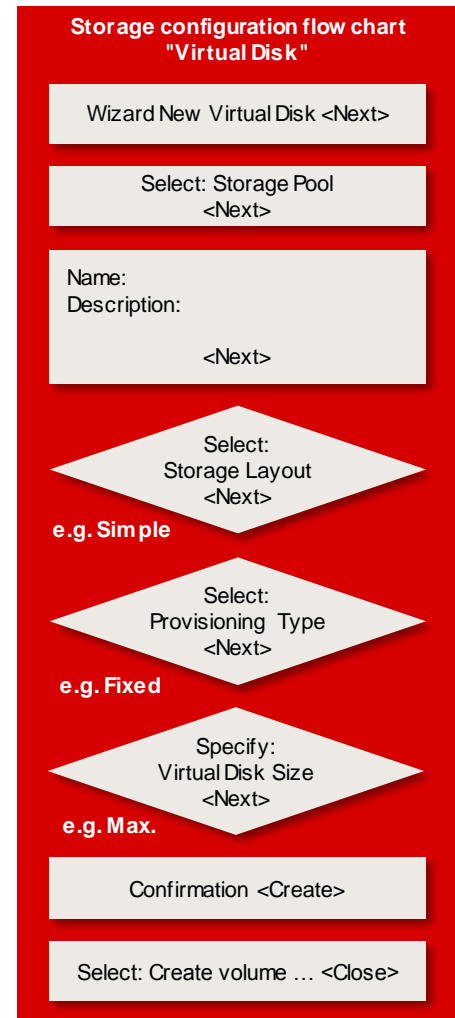
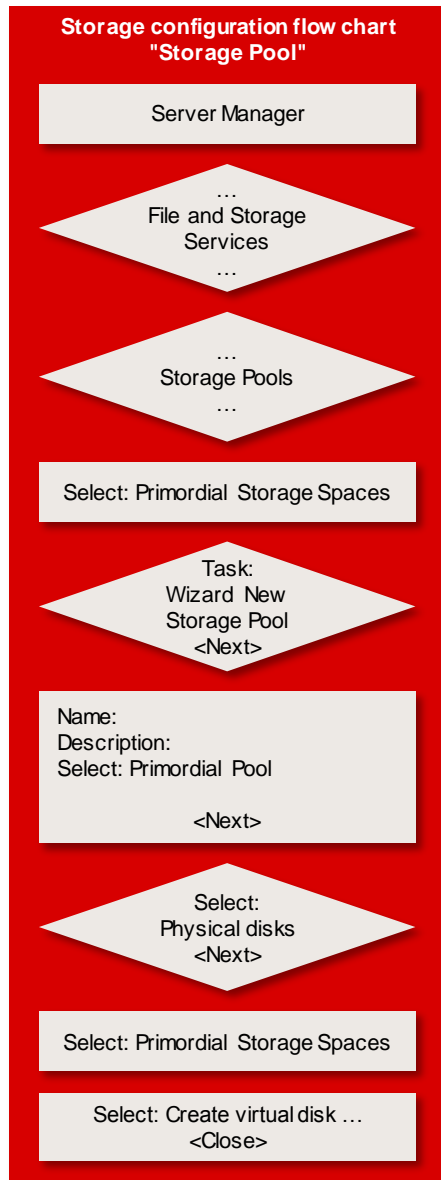
Windows Server 2012 R2 Storage Spaces has the new functionality, and in case of a hard disk failure rebuilds the data in a free storage Space in the storage pool instead of using a hot spare hard disk. Microsoft calls this approach of a significantly faster rebuild an advantage, because it is possible to write simultaneously on several hard disks whereas you can only write on one hard disk during a rebuild onto a hot spare hard disk. However, a recommendation cannot be made for this feature, because no description of how the feature works was available at the time this white paper was written.

## Configuration

By selecting File and Storage Services under Windows Server Manager you can easily configure the hard disk storage.

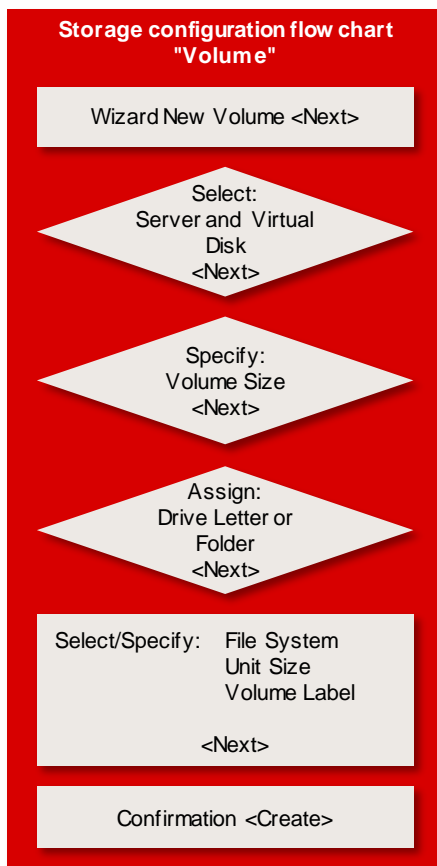
The previously unallocated hard disks are displayed in the so-called Primordial Storage Space in the Storage Pool section. The wizards New Storage Pool, New Virtual Disk and New Volume can be used to configure the data media, data area and storage layouts.

The following sequence schematically depicts the fundamental procedure:



According to this process a configured virtual disk is shown in the Server Manager. We can also see this disk in the Disk Manager, but as the corresponding unallocated basic disk, i.e. not yet operational.

We can now use the New Volume wizard to complete the configuration according to the following process:



The drive is now listed in the Server Manager, in the Disk Manager and also in the File Explorer as an operational hard disk drive.

Note:

In addition to the number of hard disks for a pool, the decisive steps of the selection process are the specifications for the storage layout (Simple, Mirror, Parity) and for the provisioning (Thin or Fixed).

## Measurement method

As standard, performance measurements of disk subsystems in PRIMERGY servers are carried out with a defined measurement method, which models the hard disk accesses of real application scenarios on the basis of specifications.

The essential specifications are:

- Share of random accesses / sequential accesses
- Share of read / write access types
- Block size (kB)
- Number of parallel accesses (# of outstanding I/Os)

A given value combination of these specifications is known as “load profile”. The following five standard load profiles can be allocated to typical application scenarios:

Standard load profile	Access	Type of access		Block size [KB]	Application
		read	write		
File copy	random	50%	50%	64	Copying of files
File server	random	67%	33%	64	File server
Database	random	67%	33%	8	Database (data transfer) Mail server
Streaming	sequential	100%	0%	64	Database (log file), Data backup; Video streaming (partial)
Restore	sequential	0%	100%	64	Restoring of files

In order to model applications that access in parallel with a different load intensity, the “# of Outstanding I/Os” is increased, starting with 1, 3, 8 and going up to 512 (from 8 onwards in increments to the power of two).

The measurements of this document are based on these standard load profiles.

The main results of a measurement are:

- Throughput [MB/s]      Throughput in megabytes per second
- Transactions [I/O/s]      Transaction rate in I/O operations per second
- Latency [ms]      Average response time in ms

The data throughput has established itself as the normal measurement variable for sequential load profiles, whereas the measurement variable “transaction rate” is mostly used for random load profiles with their small block sizes. Data throughput and transaction rate are directly proportional to each other and can be transferred to each other according to the formula

<i>Data throughput [MB/s]</i>	$= \text{Transaction rate [I/O/s]} \times \text{Block size [KB]}$
<i>Transaction rate [I/O/s]</i>	$= \text{Data throughput [MB/s]} / \text{Block size [KB]}$

All the details of the measurement method and the basics of disk I/O performance are described in the white paper [“Basics of Disk I/O Performance”](#).

## Measurement environment

The results presented here are valid for all the PRIMERGY systems in the following configuration:

System Under Test (SUT)	
<b>Hardware</b>	
Processor	2 × Xeon E5-2670 v2
Data medium	8 × HD SAS 6G 900 GB 10K HOT PL 2.5" EP 3 × SSD SAS 12G 400 GB Main 2.5" H-P EP
<b>Software</b>	
Operating system	Microsoft Windows Server 2012 R2 Standard
Controller	RAID Ctrl SAS 6G 5/6 1GB (D3116C) Driver name: megasas2.sys, Driver version: 6.600.21.8 Firmware package: 23.9.0-0031, Firmware version: 3.220.125-2420 Controller cache: 1 GB PSAS CP200i Driver name: lsi_sas2.sys, Driver version: 2.055.84 Firmware version: 14.00.00.00-IT
Cache settings	RAID Ctrl SAS 6G 5/6 1GB (D3116) Controller-Cache: Read-ahead, Write-back, Read Direct Disk Cache: enabled <sup>1)</sup>
Initialization of RAID arrays	Stripe Size: Storage Spaces: 256 kB (Default) SW-RAID: 64 kB (Fixed) HW-RAID: 256 kB
File system	NTFS
Measuring tool	Iometer 2006.07.27
Measurement data	Measurement file of 32 GB

Note: Some components may not be available in all countries or sales regions.

- 1) For performance reasons all hard disk caches are enabled for all the measurements in this document. In order to minimize the risk of data loss during productive operation in the case of a power failure the use of a UPS (uninterruptible power supply) is strongly recommended. To be able to completely rule out the possible loss of data in the cache in such cases all hard disk caches would have to be disabled.

## Performance

In order to create comparable conditions for all measurements the performance of Storage Spaces and the configuration recommendations as listed below were determined based on the hard disks listed under [“Measurement environment”](#).

### Storage Spaces vs. Storage Spaces R2

Performance comparisons of the “Simple” and “Mirror” Spaces layouts between Storage Spaces and Storage Spaces R2 have shown that there are no differences or that they are within the inaccuracy of the measurements.

In the case of the “Single Parity” Spaces layout a considerable increase in performance can be seen due to the new functionality “SSD Write-Back Cache” in Storage Spaces R2, which is why Microsoft only recommends this Spaces layout with SSD write-back cache. The measurement results for this are listed in subsection [“Storage Spaces with SSD Write-Back Cache”](#).

In Storage Spaces R2 the performance for sequential write is between 10% and 40% worse than with the predecessor for “Single Parity” Spaces layout without WBC. On the other hand, an up to 20% improvement in performance can be observed with random access.

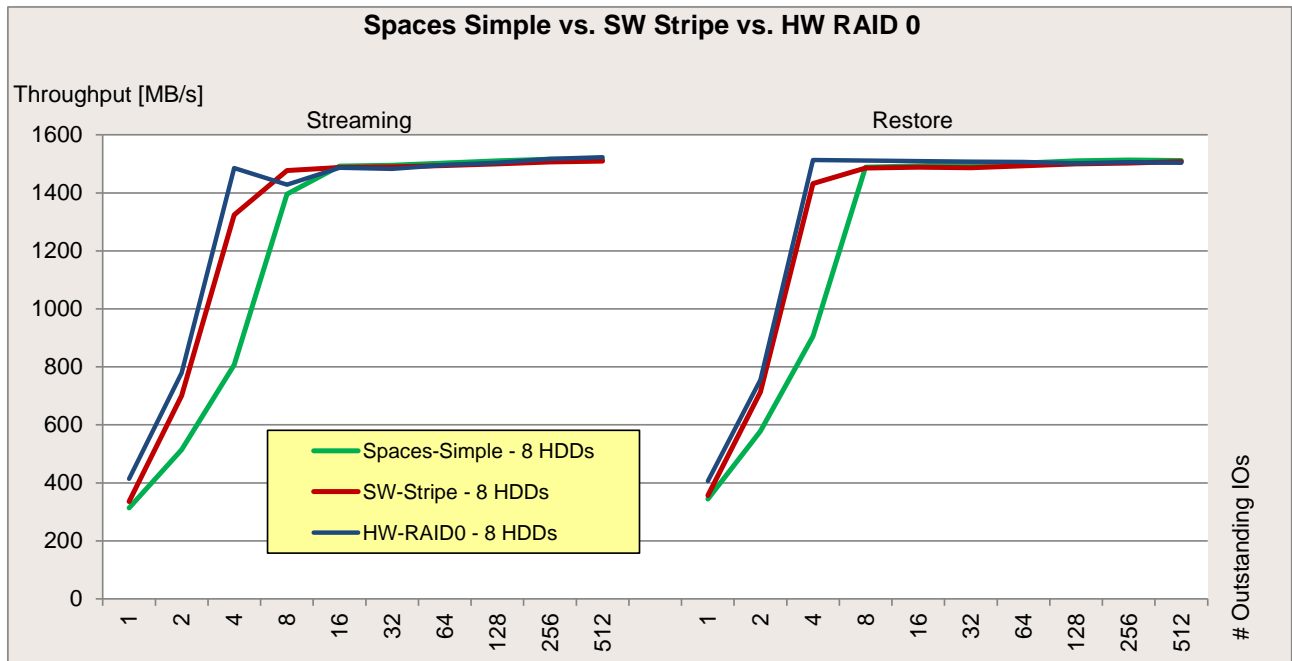
Microsoft clearly recommends you to only use the “Parity” Spaces layout without WBC for archive-like scenarios in which only reading takes place almost exclusively.

### Storage Spaces vs. Software-RAID vs. Hardware-RAID

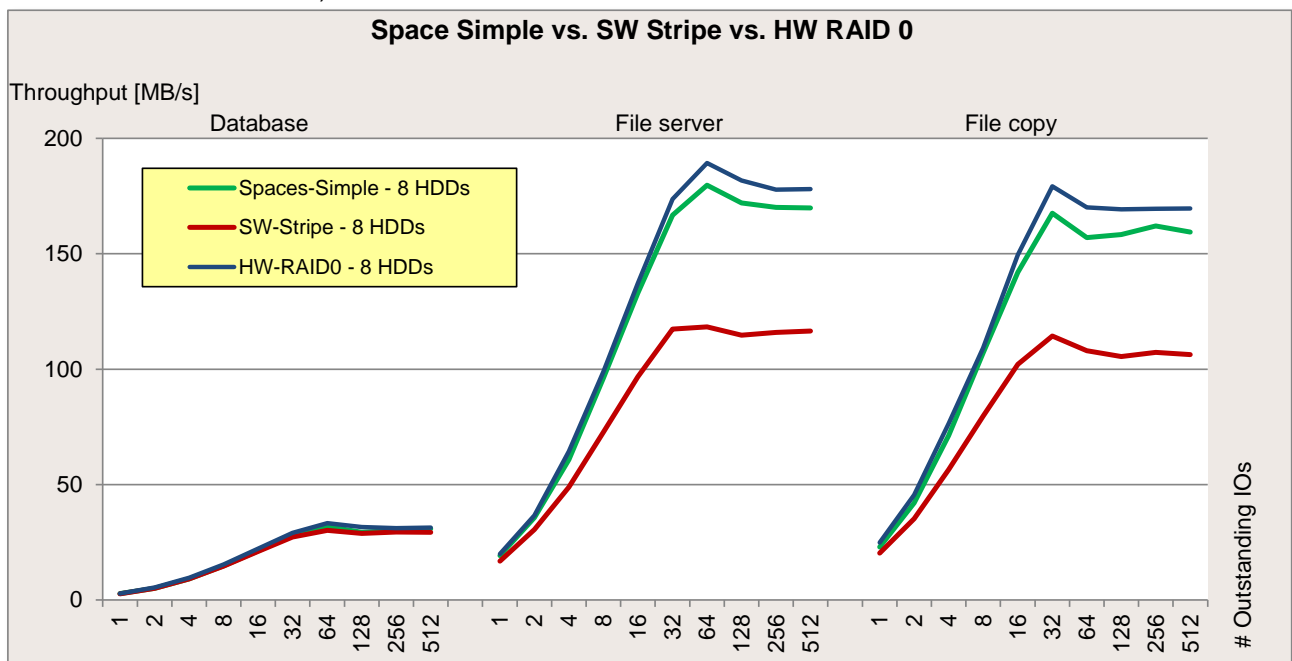
The comparison of a classic hardware RAID with the Storage Spaces function is of particular interest. The Space layouts “Simple/Mirror/Single Parity/Dual Parity” are compared with the appropriate HW RAID levels of a “RAID Ctrl SAS 6G 5/6 1 GB (D3116C)” controller. As already mentioned in the introduction to this document, a comparison with Windows software RAID will also be carried out at this juncture. The individual configurations were consistently measured with enabled disk cache and, apart from software RAID, with a stripe size of 256 kB. Software RAID is an exception, because the stripe size is unalterably set at 64 kB.

### Spaces Simple vs. SW Stripe vs. HW RAID 0

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



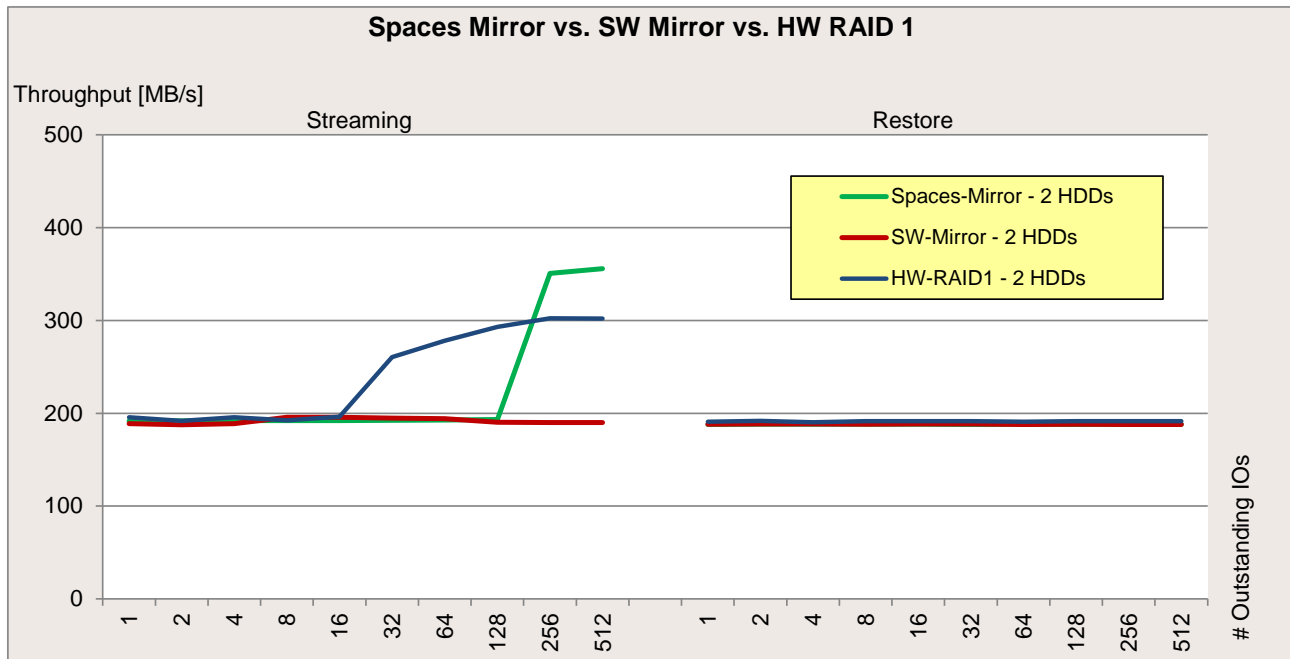
The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).



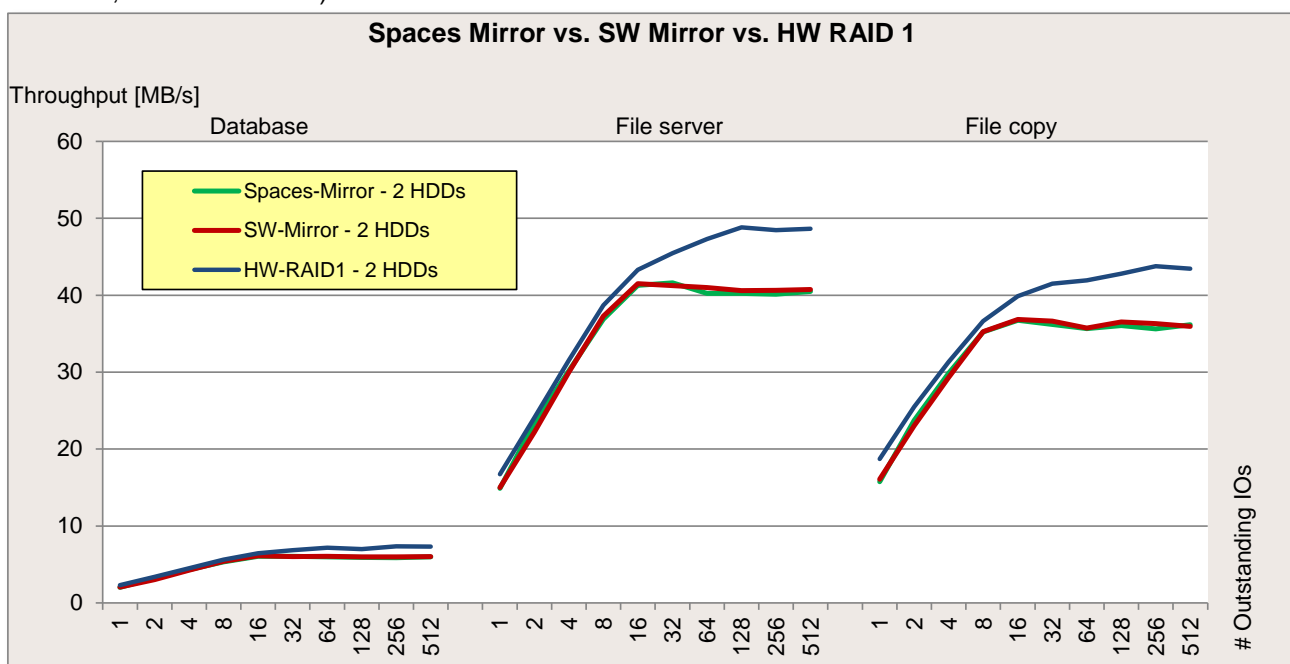
In the case of the load profiles “Streaming”, “Restore” and “Database” the performance is very similar for all three versions. It is merely evident that Storage Spaces can only close the gap on the performance of software RAID and hardware RAID for the load profiles “Streaming” and “Restore” with a higher number of outstanding I/Os. Storage Spaces achieves approximately the same speed as hardware RAID for the load profiles “File server” and “File copy”. Software RAID performs significantly worse with these two load profiles.

### Spaces Mirror vs. SW Mirror vs. HW RAID 1

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



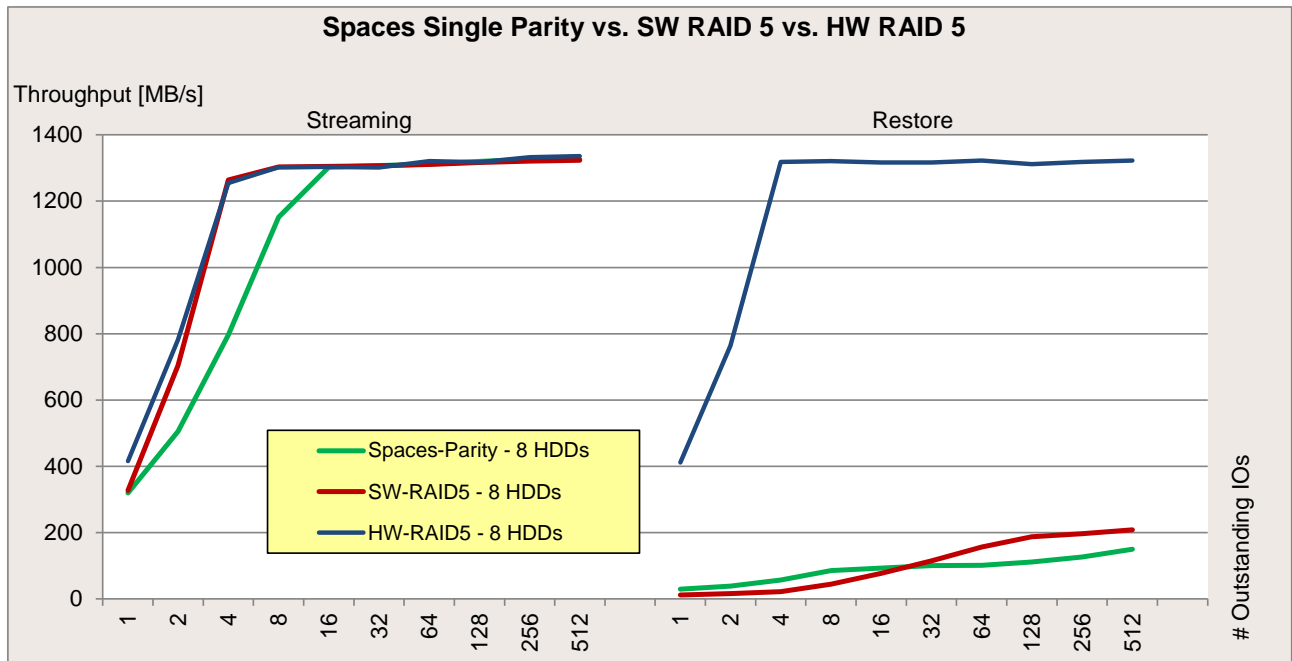
The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).



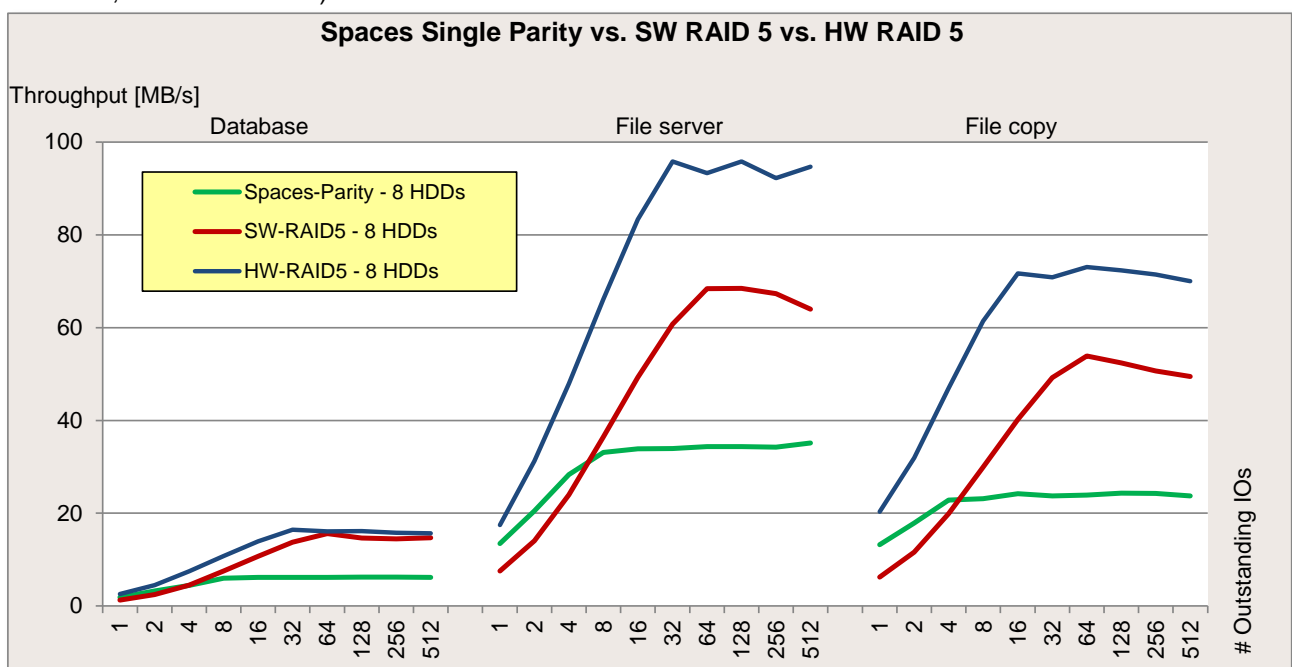
Whereas Storage Spaces and software RAID have approximately the same performance – apart from the load profile “Streaming” with high load intensities (Outstanding IOs), where Storage Spaces is clearly faster – hardware RAID is up to approx. 20% (“Database”, “File server”, “File copy”) and 40% (“Streaming”) faster with higher load intensities (Outstanding IOs). There are no differences in performance to be seen with the load profile “Restore”.

### Spaces Single Parity vs. SW RAID 5 vs. HW RAID 5

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).

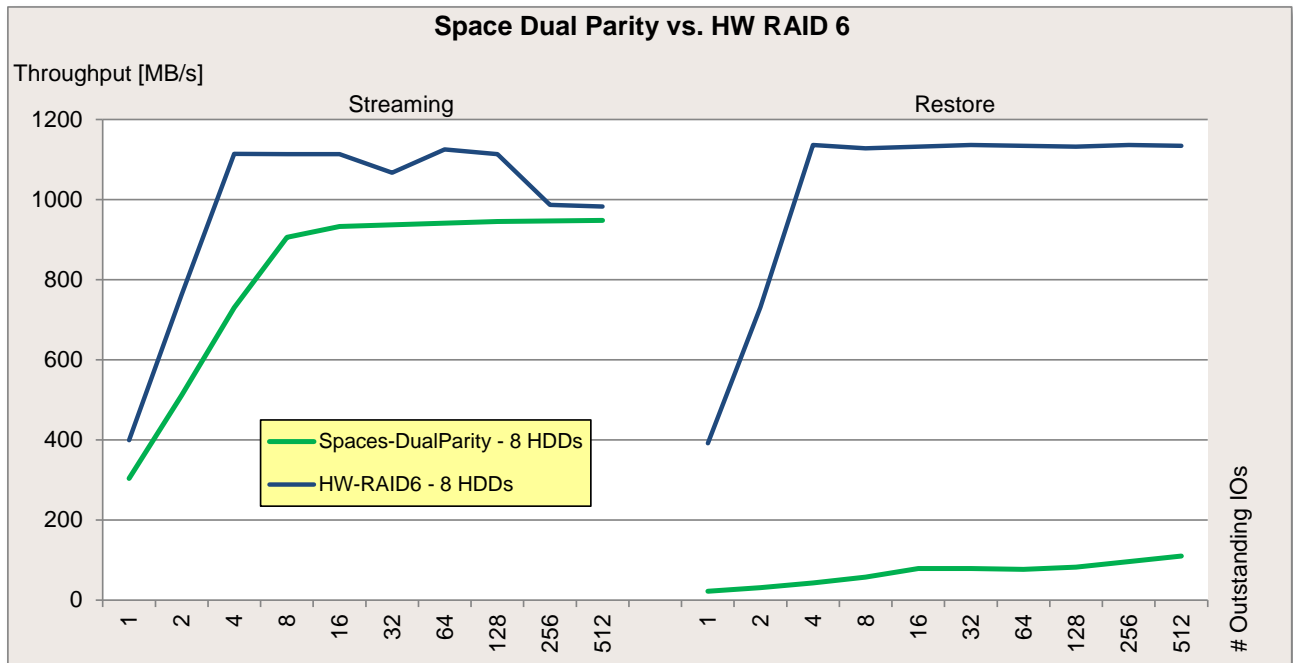


As with the other Spaces layouts, it is evident that Storage Spaces only closes the gap on the performance of software RAID and hardware RAID for the load profile “Streaming” with a higher number of outstanding IOs.

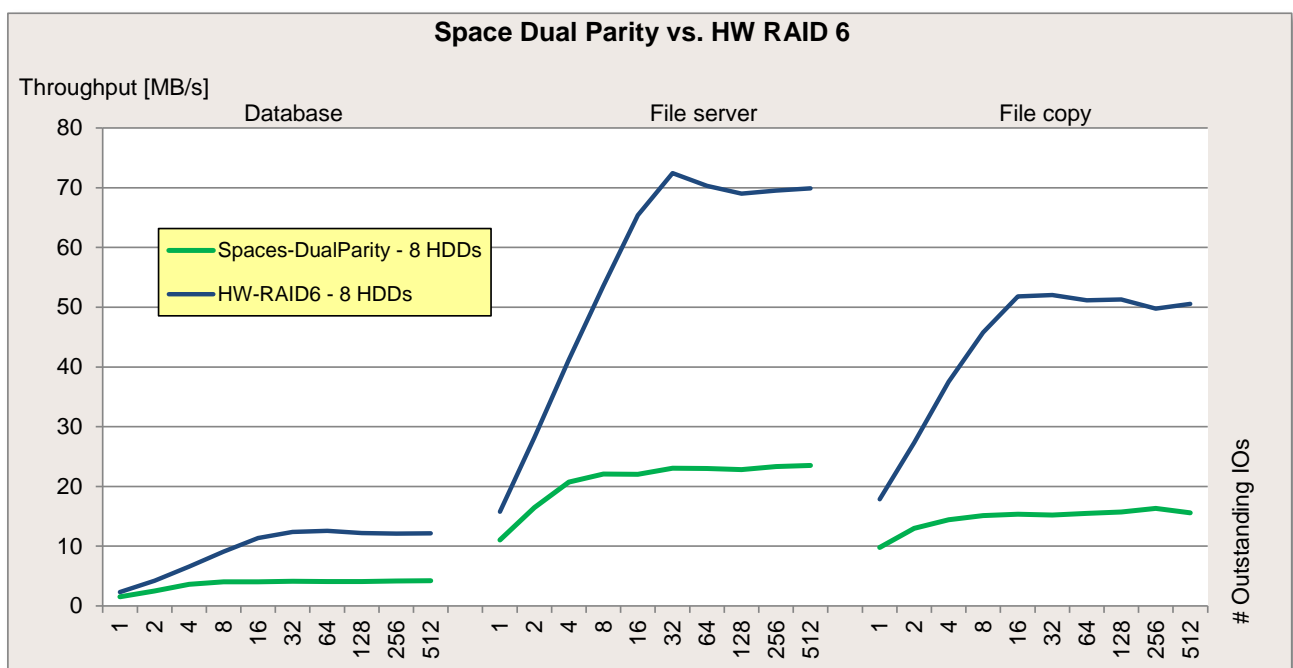
In all the other load profiles, particularly with higher load intensities (Outstanding IOs), the performance of hardware RAID is on account of the controller cache between three and ten times superior to that of Storage Spaces. Whereas software RAID provides approximately the same performance for the load profiles “Streaming” and “Database” as hardware RAID, it is 20%-30% inferior for the load profiles “File server” and “File copy”.

## Spaces Dual Parity vs. HW RAID 6

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).



The overall picture resembles that of the “Parity” Spaces layout. The performance of hardware RAID is considerably better than that of Storage Spaces. Whereas the performance of Storage Spaces in the “Parity” Spaces layout was still comparable with that of hardware RAID 5 for the load profile “Streaming”, the performance in the “Dual Parity” Spaces layout is in part clearly worse than with RAID 6.

***Conclusion: Storage Spaces vs. Software-RAID vs. Hardware-RAID***

In summary it can be said that Storage Spaces can by all means be an alternative to hardware RAID 0 and 1 in the Spaces layouts "Simple" and "Mirror" and to software RAID in "Stripe" and "Mirror".

Compared with hardware RAID 5 and 6 it is not possible from a performance view to make a recommendation - except for read-intensive loads - for the "Parity" and "Dual Parity" Spaces layouts. For this reason Microsoft introduced the WBC in Storage Spaces R2, which we will look at in more detail in the next section.

Whether the use of software RAID 5 compare with hardware RAID 5 makes sense depends on the load profile and must be tested for each individual case.

## Storage Spaces with SSD Write-Back Cache

The following configurations are compared in this section:

- Storage Spaces (no WBC)
- Storage Spaces with 1 GB WBC on two SSDs (only for WBC)
- Software RAID
- Hardware RAID

Data that is written to a Storage Space with a configured WBC is for the time being only saved on the WBC, i.e. on the SSDs. As of a specific filling degree (usually between 25% and 50%) Storage Spaces starts to empty the cache by moving the data to the hard disks.

However, as long as the data is in the WBC read operations, which affect this data, are served from the WBC, i.e. from the SSDs. This makes reproducible and meaningful measurements difficult, because read performance largely depends on how much of the relevant data is currently in the WBC. This means that a larger WBC can under circumstances improve the read performance.

Tests have shown that the WBC size only has a limited influence on write performance. As long as capacity is still free in the WBC, write operations, such as those to be seen in subsection [“Spaces WBC Random Write Performance”](#) achieve approximately the native SSD speed. However, as soon as the WBC is written faster than it can be emptied and therefore fills up, there is no improvement to be seen in write performance from a WBC size of 1 GB.

If sufficient SSD capacity is available, a WBC size of 10 GB should be selected in order to compensate potential write phases with a high load. Microsoft recommends a WBC size of up to 10 GB. It is not possible to make a general recommendation as regards the size of the WBC, because it essentially depends on how much data can be expected during peak write phases. If sufficient SSD capacity is available, a WBC size of 10 GB should be selected in order to cater as effectively as possible for any peak write phases.

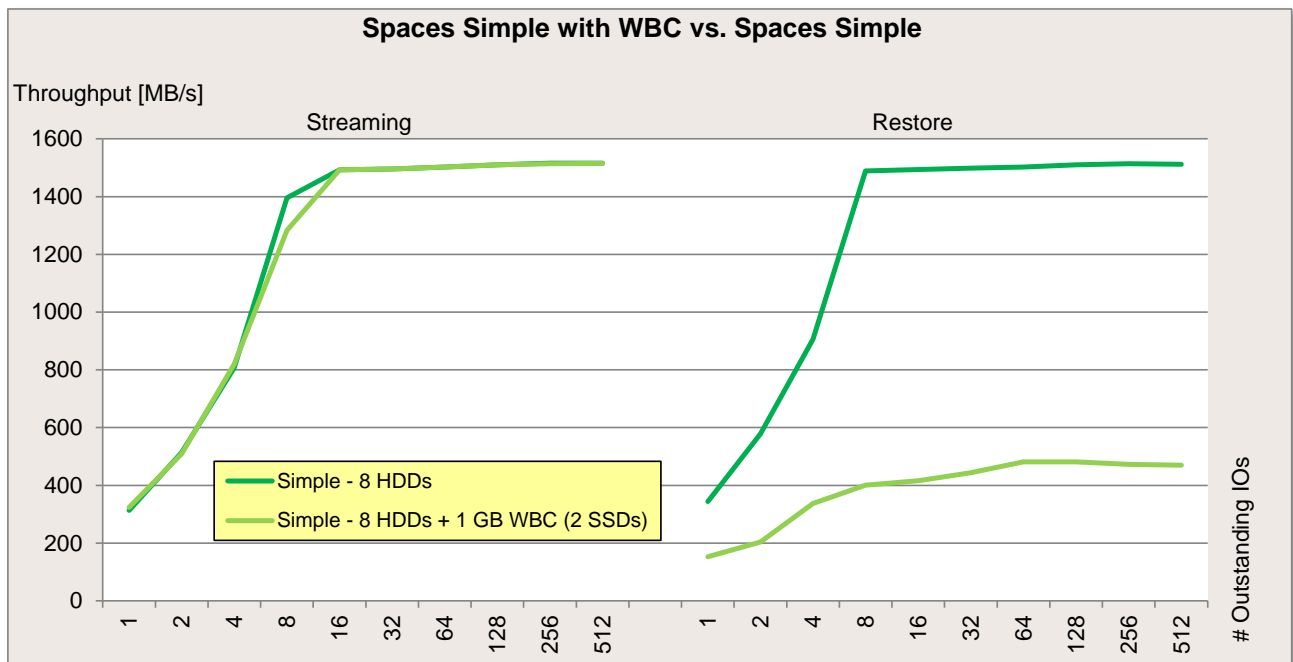
The following diagrams show that Storage Spaces with WBC in the load profile “Restore” have an altogether significantly lower performance level than Storage Spaces without WBC.

The reason for this is the very good sequential write performance of hard disks, which are able to keep up with the performance of SSDs. As there are only two SSDs in a configured WBC and all the data must first be written to the WBC, the sequential write performance for Storage Spaces with WBC is accordingly low. However, this only applies for applications that use block sizes smaller than 256 kB and for synthetic load simulations, such as in this case, the measurement program “IOMeter”.

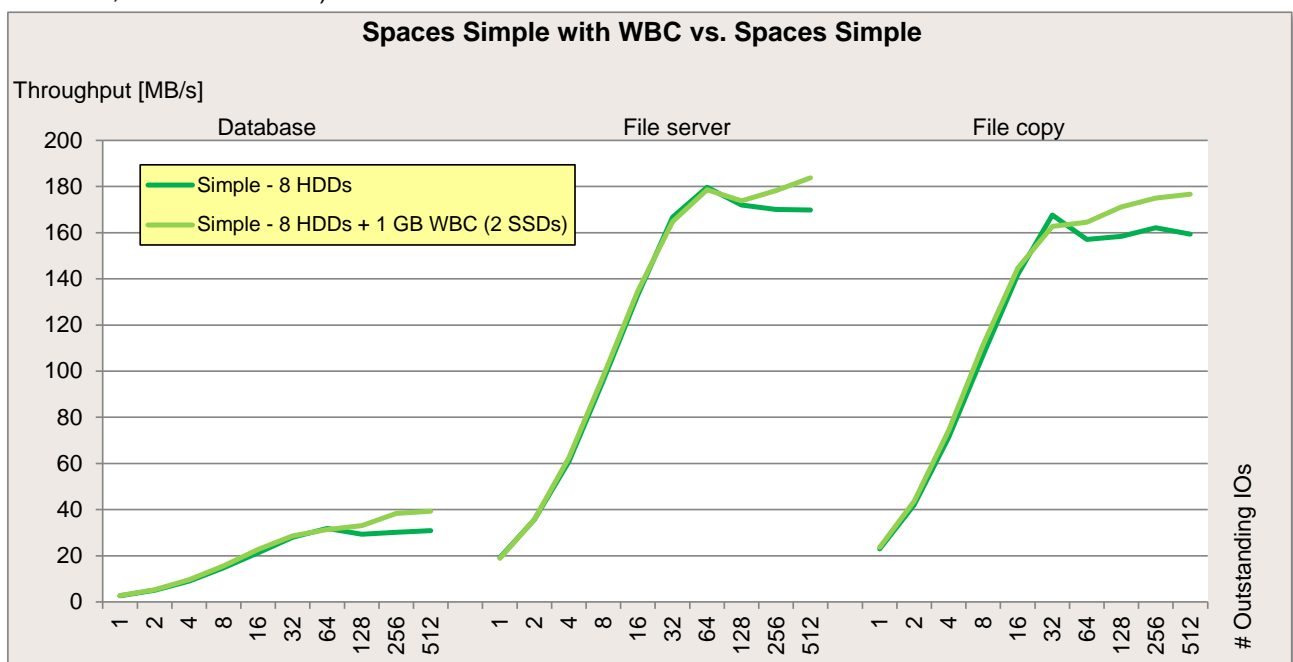
If the data blocks written by the application are 256 kB and greater, Storage Spaces recognizes this and writes the data directly to the hard disks. In so doing, the speed shown in the section [“Storage Spaces vs. Software RAID vs. Hardware RAID”](#) is achieved. However, this only applies for the “Simple” and “Mirror” Spaces layouts. “Parity” Spaces layouts do not have this function and thus possibly have the poorer sequential write performance of SSDs.

### Spaces Simple with WBC vs. Spaces Simple

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



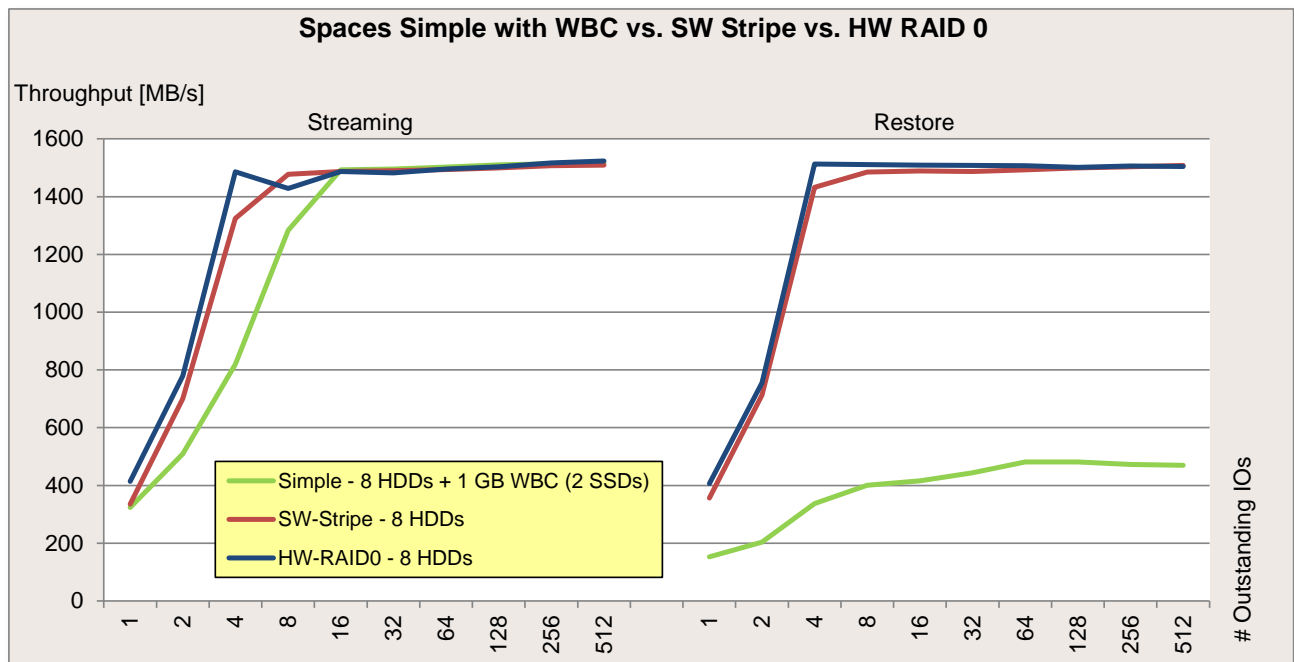
The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).



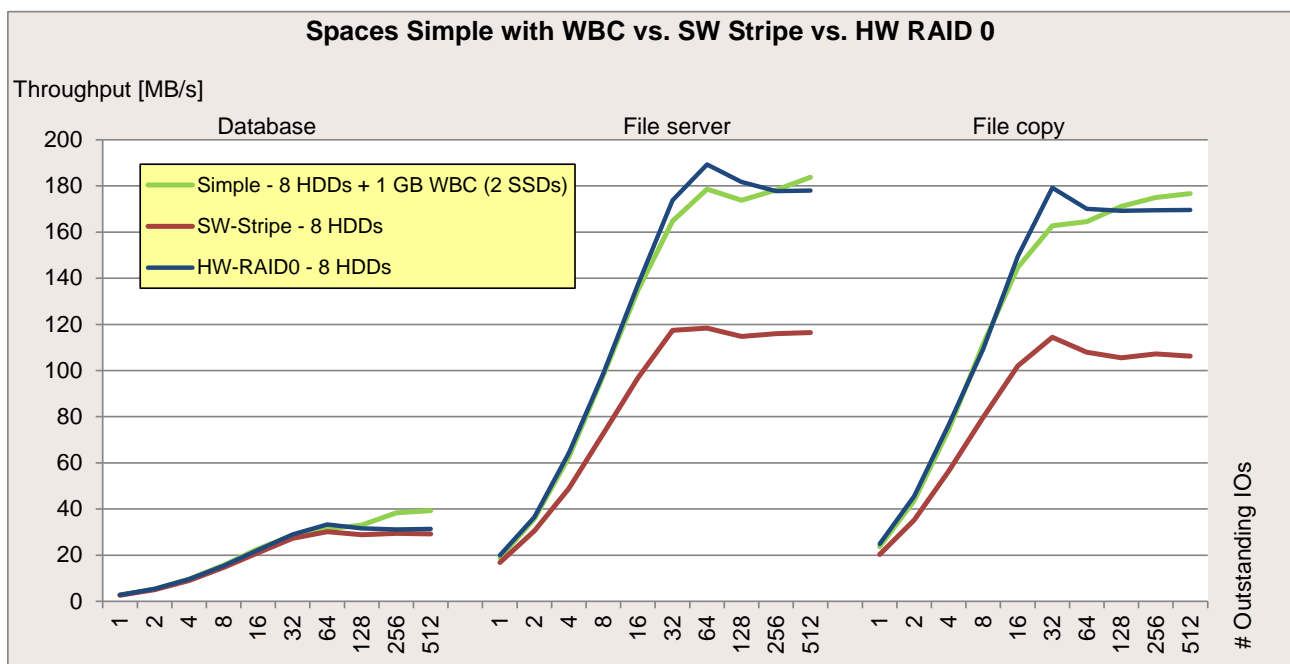
Based on the measurement results you can see that in the case of long-lasting write loads the WBC only offers a slightly improved performance with a higher number of outstanding IOs. The WBC has no impact on the read performance. As already described at the [start of the section](#), the measurement results in the load profile “Restore” do not necessarily correspond to practice. If the application uses a block size of 256 kB or greater, it can be assumed that the same performance will be achieved as without a configured WBC.

### Spaces Simple with WBC vs. SW Stripe vs. HW RAID 0

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).

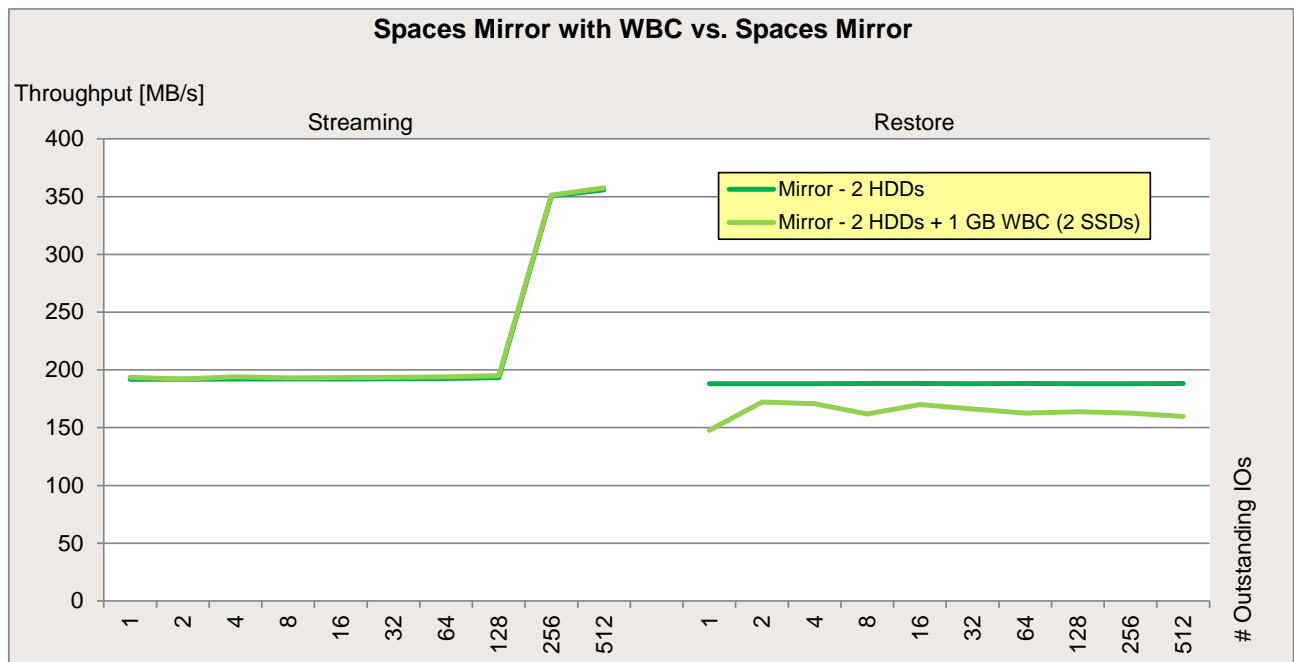


As the use of a WBC in the Spaces layout “Simple”, as described in the previous subsection, only has a very small influence, the comparison to software RAID and hardware RAID is correspondingly similar to subsection [“Spaces Simple vs. SW Stripe vs. HW RAID 0”](#).

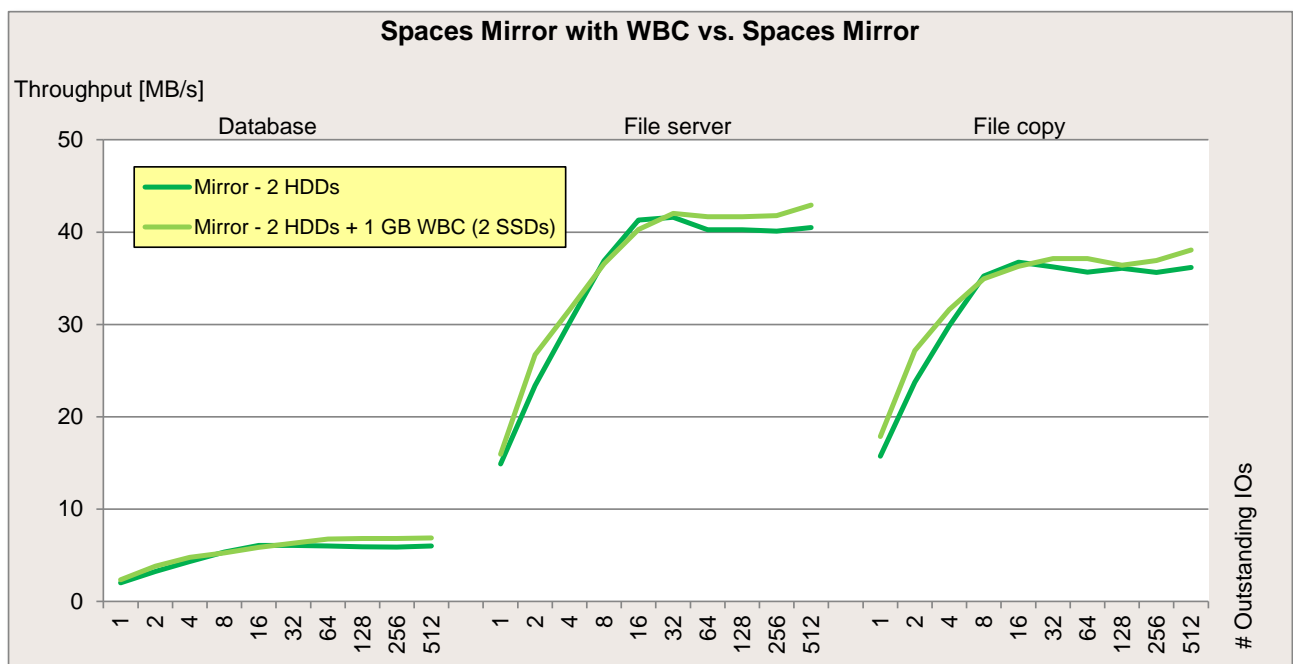
As already described at the [start of the section](#), the measurement results in the load profile “Restore” do not necessarily correspond to practice. If the application uses a block size of 256 kB or greater, it can be assumed that the same performance will be achieved as without a configured WBC.

### Spaces Mirror with WBC vs. Spaces Mirror

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).

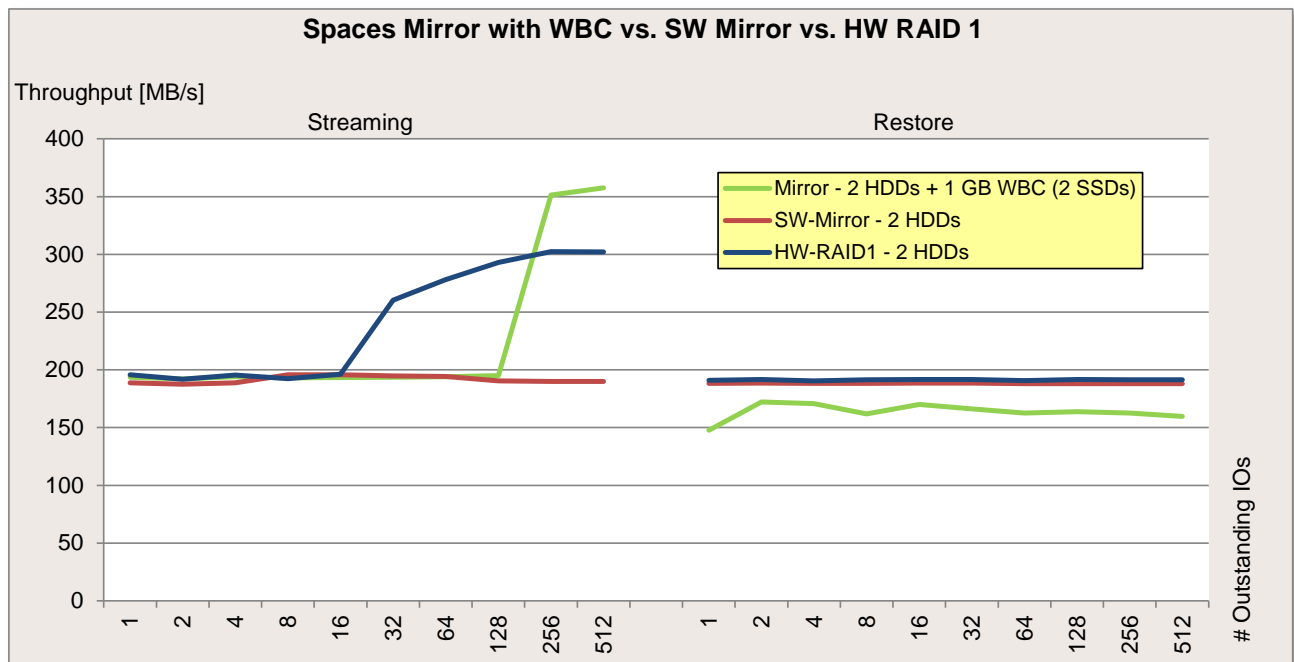


Based on the measurement results you can see that in the case of long-lasting write loads the WBC only provides a slight improvement. The performance is even absolutely identical for read accesses.

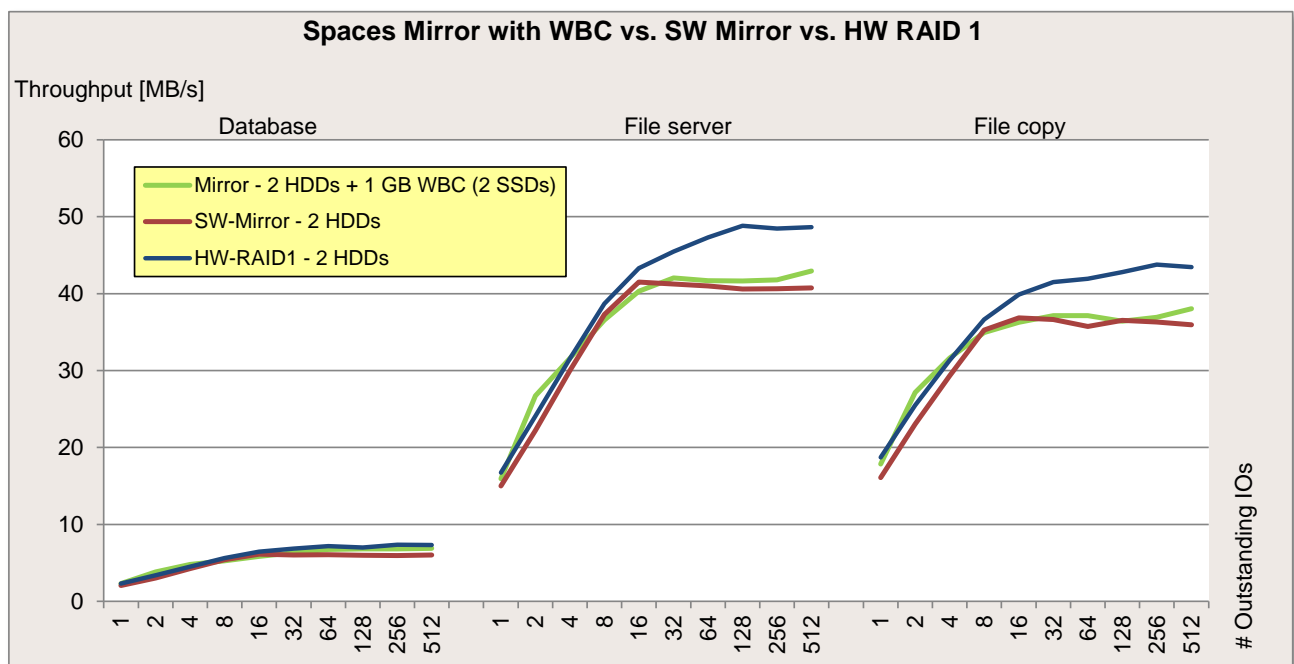
As already described at the [start of the section](#), the measurement results in the load profile “Restore” do not necessarily correspond to practice. If the application uses a block size of 256 kB or greater, it can be assumed that the same performance will be achieved as without a configured WBC.

### Spaces Mirror with WBC vs. SW Mirror vs. HW RAID 1

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).

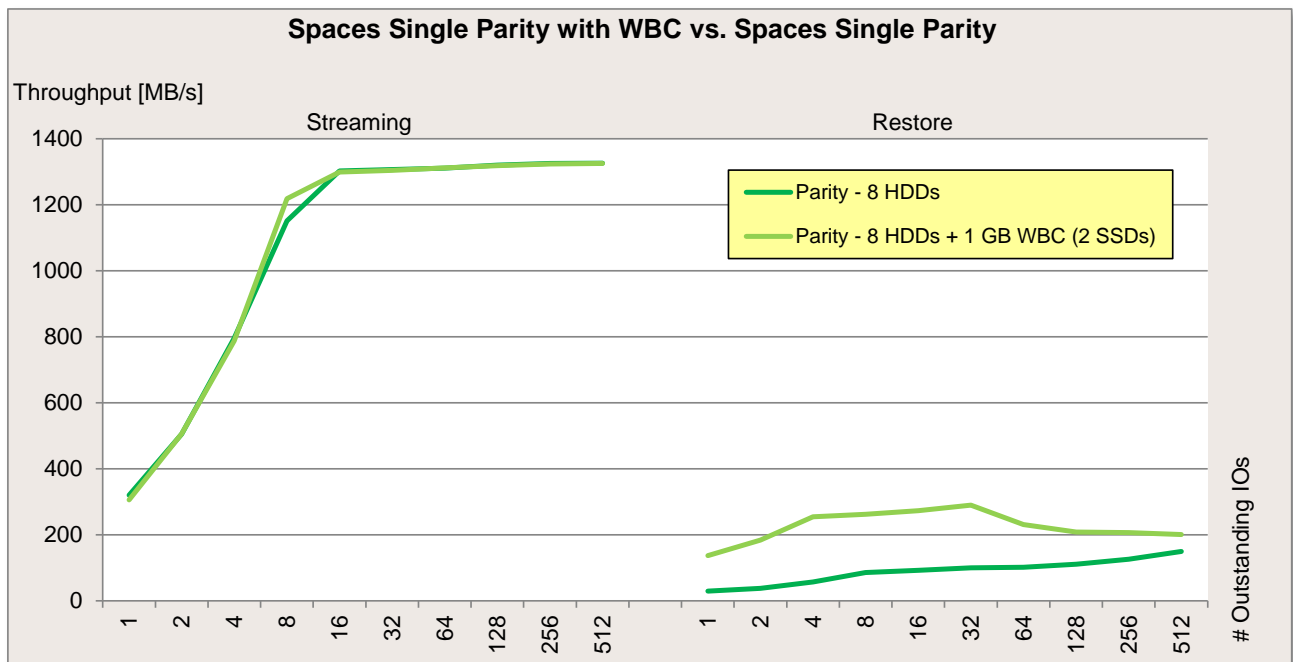


As the use of a WBC in the Spaces layout “Simple”, as described in the previous subsection, only has a very small influence, the comparison to software RAID and hardware RAID is correspondingly similar to subsection [“Spaces Simple vs. SW Mirror vs. HW RAID 1”](#).

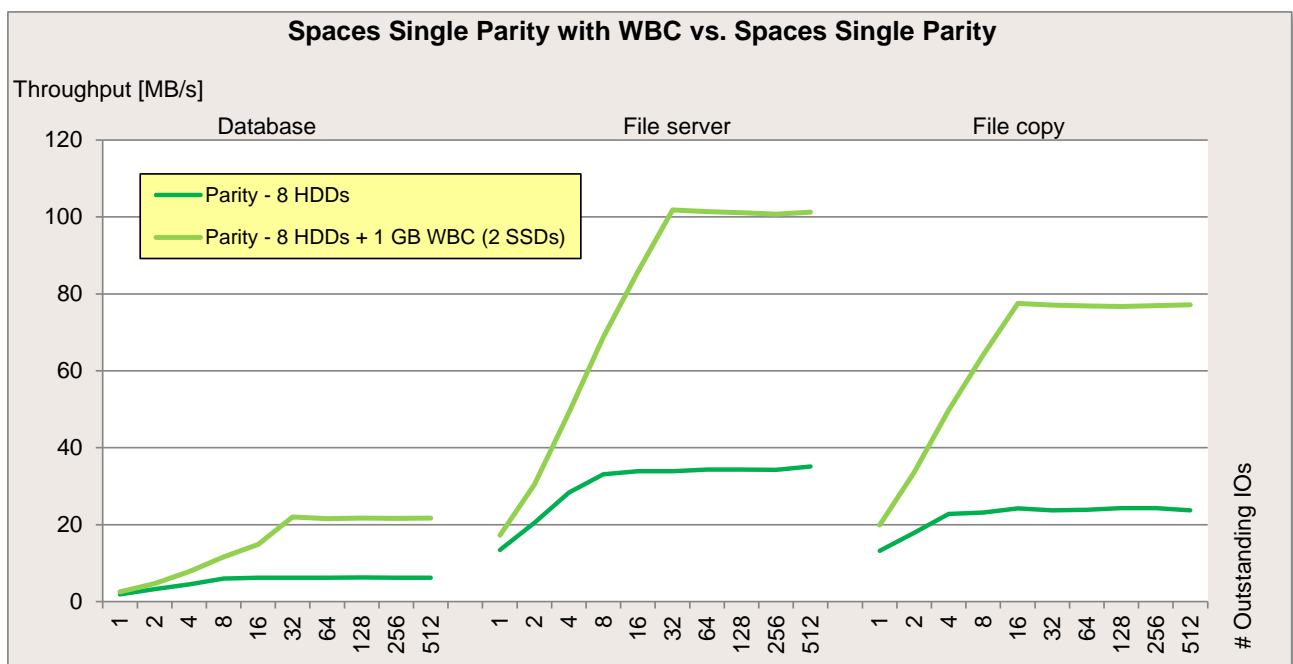
As already described at the [start of the section](#), the measurement results in the load profile “Restore” do not necessarily correspond to practice. If the application uses a block size of 256 kB or greater, it can be assumed that the same performance will be achieved as without a configured WBC.

### Spaces Single Parity with WBC vs. Spaces Single Parity

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).

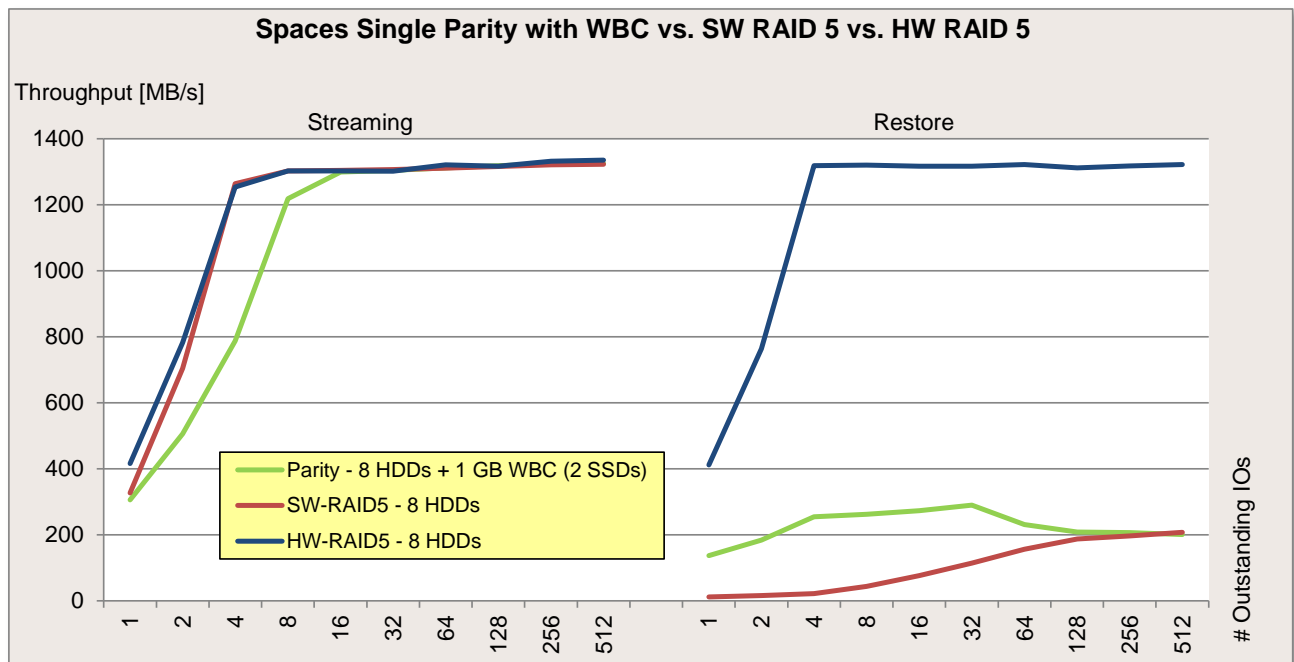


The WBC shows significant improvements in all load profiles in the “Parity” Spaces layout, apart from the load profile “Streaming”, over which the WBC has no substantial influence.

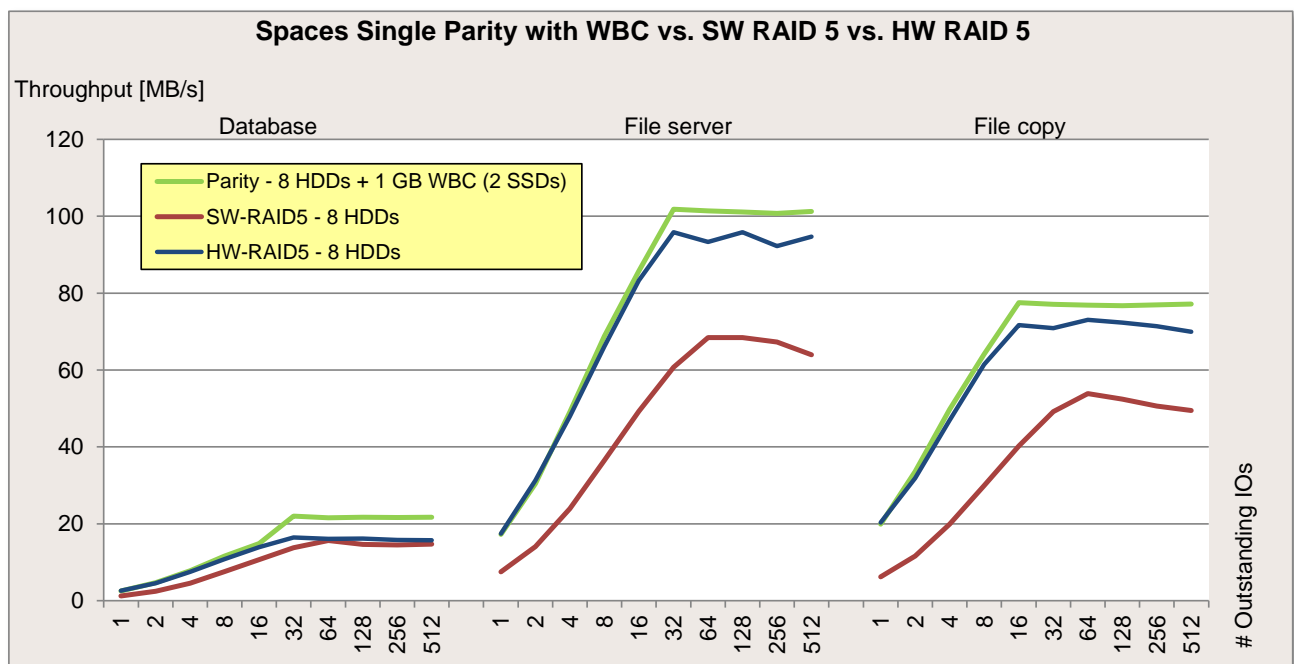
A clear increase in performance can also be seen with sequential write.

### Spaces Single Parity with WBC vs. SW RAID 5 vs. HW RAID 5

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).

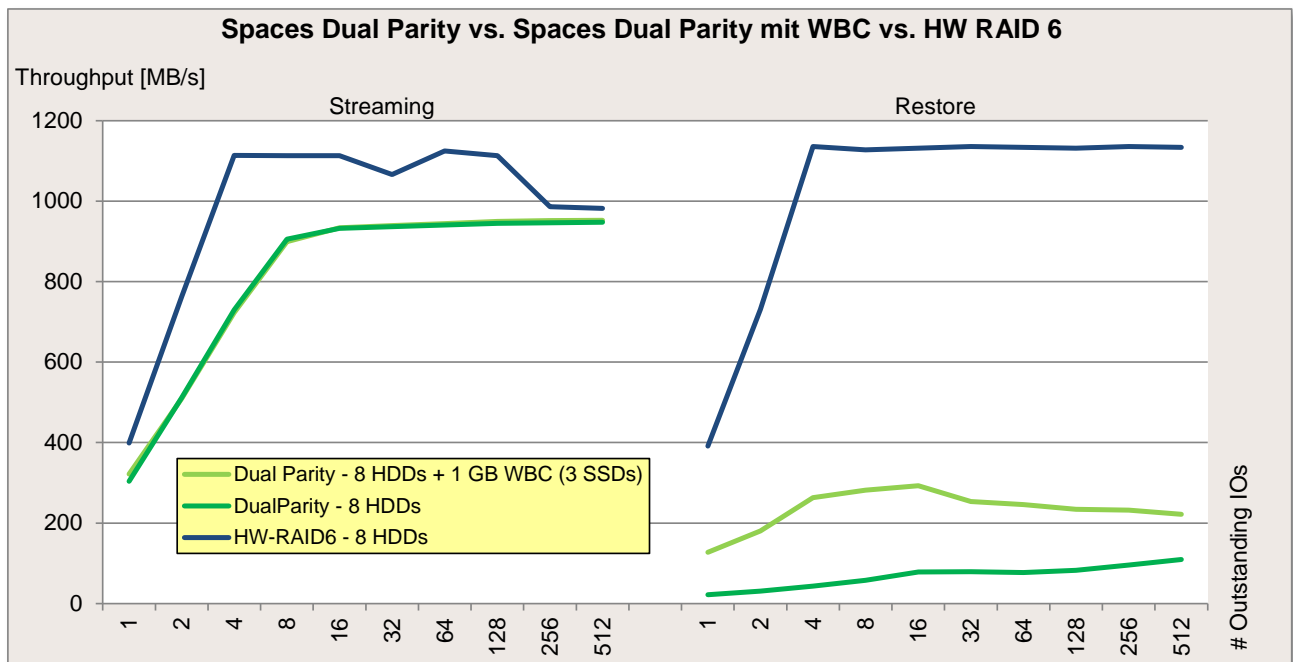


As a result of the WBC, it is possible for Storage Spaces to provide comparable performance values for the “Parity” Spaces layout in load profiles with mixed random access (second diagram) and even slightly better performance values with a higher load intensity (Outstanding IOs), such as hardware RAID 5.

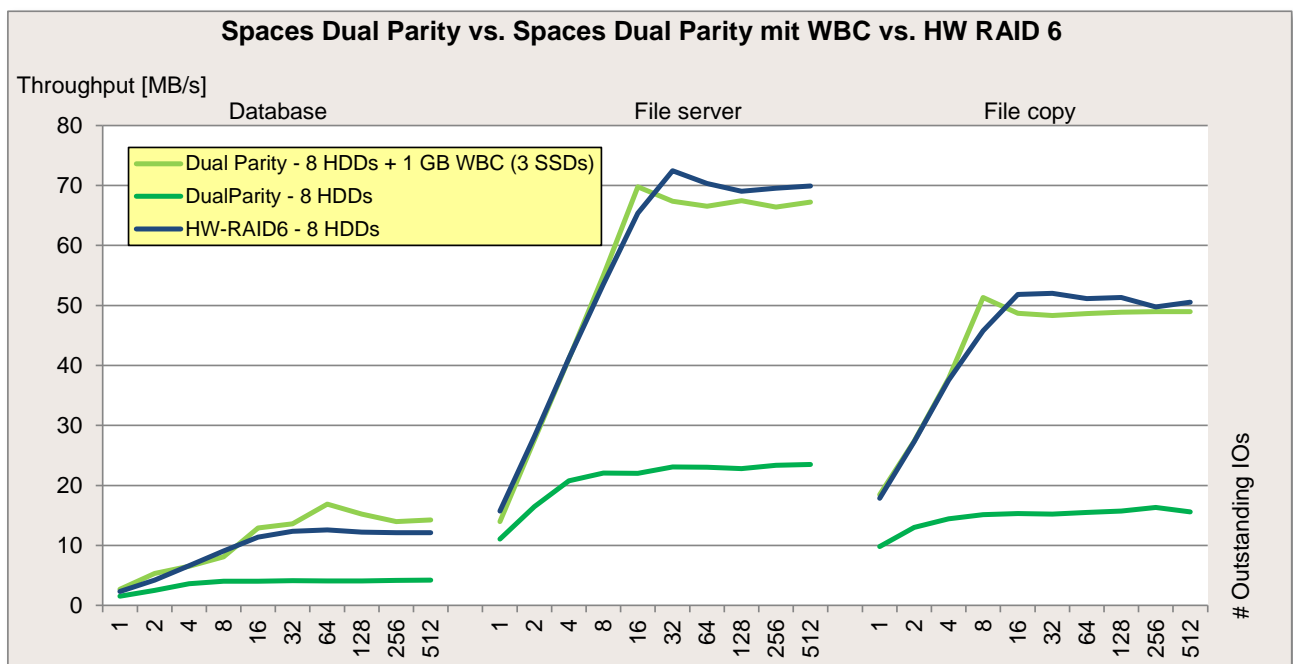
In the case of “Single Parity” Spaces layout, the performance in practice for sequential write also corresponds to the performance of the values measured here. Even if the use of WBC has notably improved the performance in this load profile, a recommendation for Storage Spaces cannot be made for this case.

### Spaces Dual Parity vs. Spaces Dual Parity with WBC vs. HW RAID 6

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).



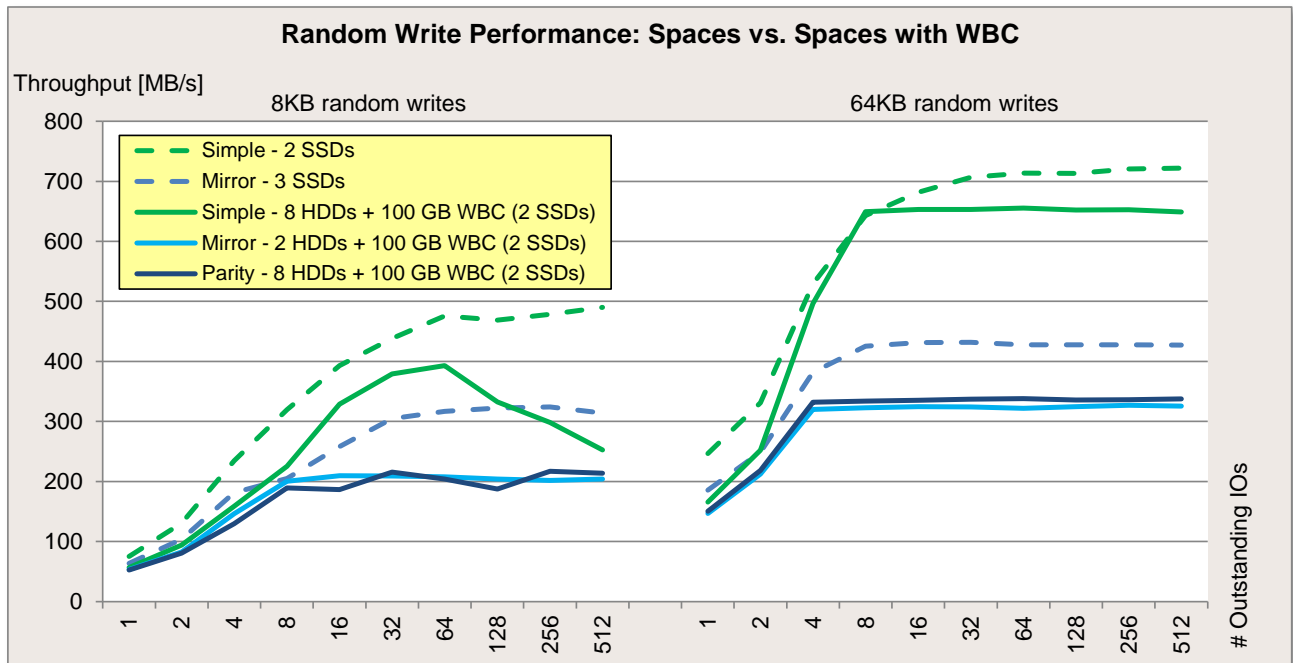
WBC is used to clearly improve the performance of Storage Spaces for the “Dual Parity” Spaces layout to the effect that a similar performance can be observed in the load profiles with mixed random access (second diagram) in comparison with hardware RAID.

In the case of “Dual Parity” Spaces layout the performance in practice for sequential write also corresponds to the performance of the values measured here. Even if the use of WBC has slightly improved the performance in this load profile, a recommendation for Storage Spaces cannot be made for this case.

## Spaces WBC Random Write Performance

The measurement results presented in the diagram below should indicate the maximum WBC write speed in the various Spaces layouts. This rather theoretical view only applies if the WBC is not full assigned and Storage Spaces is not currently moving the data from the WBC to the hard disks.

The following diagram shows the measurement results for two especially write-intensive load profiles (random access, 100% write, 8 kB block size and random access, 100% write, 64 kB block size).



The broken curves represent the native SSD speed in the respective Spaces layouts. In the case of “Simple” Spaces layouts the WBC is built as a “Simple” structure. In the case of “Mirror” and “Parity” Spaces layouts the WBC is built as a “Mirror” structure.

The native SSD speed is not achieved by the WBC in all cases, but offers in comparison with the performance of hard disks a considerable improvement. This increase in performance is to be expected in practice for special applications with write-intensive random write accesses.

## Conclusion: Storage Spaces with SSD Write-Back Cache

Even if the WBC does not offer any appreciable advantage in write phases of long duration in the “Simple” and “Mirror” Space layouts, it can significantly accelerate brief write phases with a high load. The performance of the WBC can be further enhanced by increasing the number of SSDs. However, it should be remembered that performance during a sequential write phase can be significantly worse when a WBC is used and the block size is less than 256 kB because SSDs do not offer better performance than HDDs in this load profile.

The WBC has a major influence on performance in the case of “Single Parity” and “Dual Parity” Spaces layouts. The extreme increase in performance achieved with mixed random load profiles “Database”, “File server” and “File copy” also offers performance comparable to Hardware RAID “5” and “6” even in write phases of long duration.

The WBC has no appreciable influence over performance in read accesses with “Single Parity” and “Dual Parity” Spaces layouts and thus remains slightly slower than with Hardware RAID “5” and “6”. In the case of sequential writes, a significant increase in performance is achieved with the “Single Parity” and “Dual Parity” Spaces layouts. Despite this fact, performance in this load profile is several times worse than with Hardware RAID “5” and “6”.

## Storage Spaces with Tiering

The following configurations are compared in this section:

- Storage Spaces (no WBC)
- Storage Spaces with 1 GB WBC on two SSDs (only for WBC)
- Storage Spaces with 1 GB WBC and SSD tier on two SSDs (only for WBC and tiering)

The measurements using tiering functionality are intended to indicate the performance of the SSD tier and thus represent the maximum values expected for the data on the SSD tier. This also corresponds to performance in practice when new data is written and the SSD tier still has sufficient capacity. If already existing data is modified, the performance will equal that of the relevant tier on which the data is located at this moment. The HDD tier benefits from a configured WBC.

The use of tiering in Storage Spaces is particularly effective if there is a significant difference in performance between the HDD tier and the SSD tier because on the one hand a benefit exists from the fact that seldom used data is stored on hard disks with the best possible price per GB ratio and, on the other hand, frequently used data is provided on particularly fast SSDs.

Measurement results for the “Simple” Spaces layout with tiering functionality are presented below with an SSD tier consisting of two SSDs and an SSD tier consisting of eight SSDs.

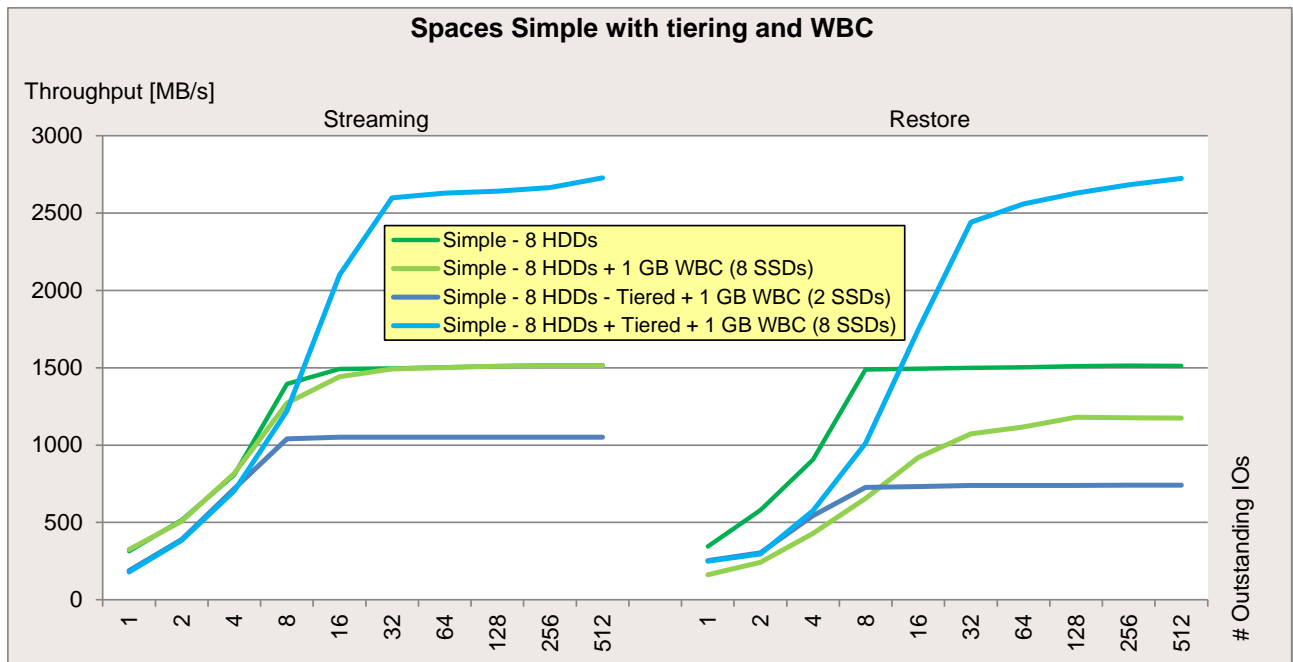
The reason for this is the restriction for tiering functionality described in the “[Storage Tiering](#)” subsection in the “[Functionality](#)” section. In a configuration of eight HDDs and two SSDs using tiering functionality, the Spaces layout would only have two columns. This reduces the sequential read and write performance of the HDDs by a factor of 4 and causes a 30% drop in random read and write performance. The number of SSDs must match the number of HDDs if this loss of performance is to be avoided.

In the case of the “Mirror” Spaces layout, the number of columns is already at the bare minimum of “one” because only two HDDs are used. Thus, the performance of the HDD tier is not affected by the use of tiering.

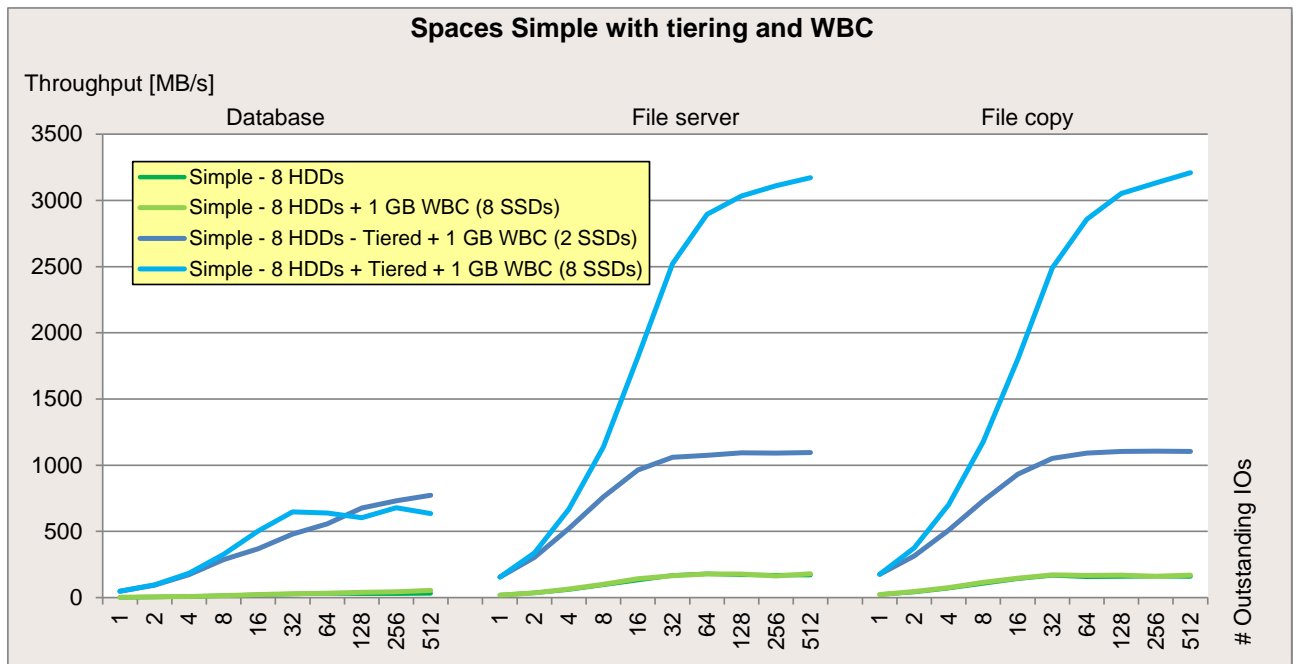
At this point it should be pointed out that a storage Space can only be configured as a tiered storage with or without WBC during initial configuration. It is only possible to extend the capacity of a HDD tier or SSD tier at a later point if the number of columns is retained.

### Spaces Simple

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).



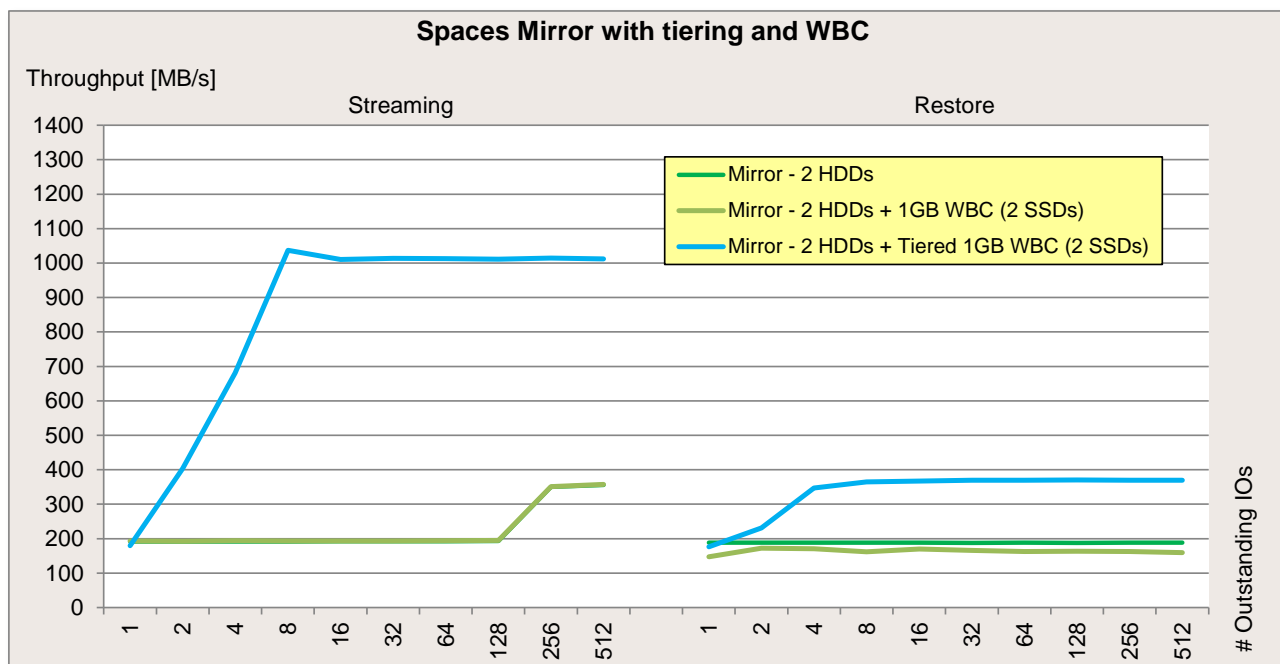
As can be seen in the first diagram, the sequential read and write performance of the SSD tier with two SSDs is much poorer than of eight HDDs without the tiering function. As explained at the start of this [section](#), the sequential read and write performance of the HDDs in this configuration is poorer by a factor of four because of the smaller number of columns. As a consequence, a significantly poorer read and write performance is to be expected in the “Simple” Spaces layout with tiering than in a configuration without tiering, irrespective of whether the data is on the HDD or SSD tier.

As already described in the “[Storage Spaces with SSD Write-Back Cache](#)” section, the measurement results for the purely WBC measurement in the “Restore” load profile do not necessarily reflect everyday practice. If the application uses a block size of 256 kB or greater, it can be assumed that the same performance will be achieved as without a configured WBC.

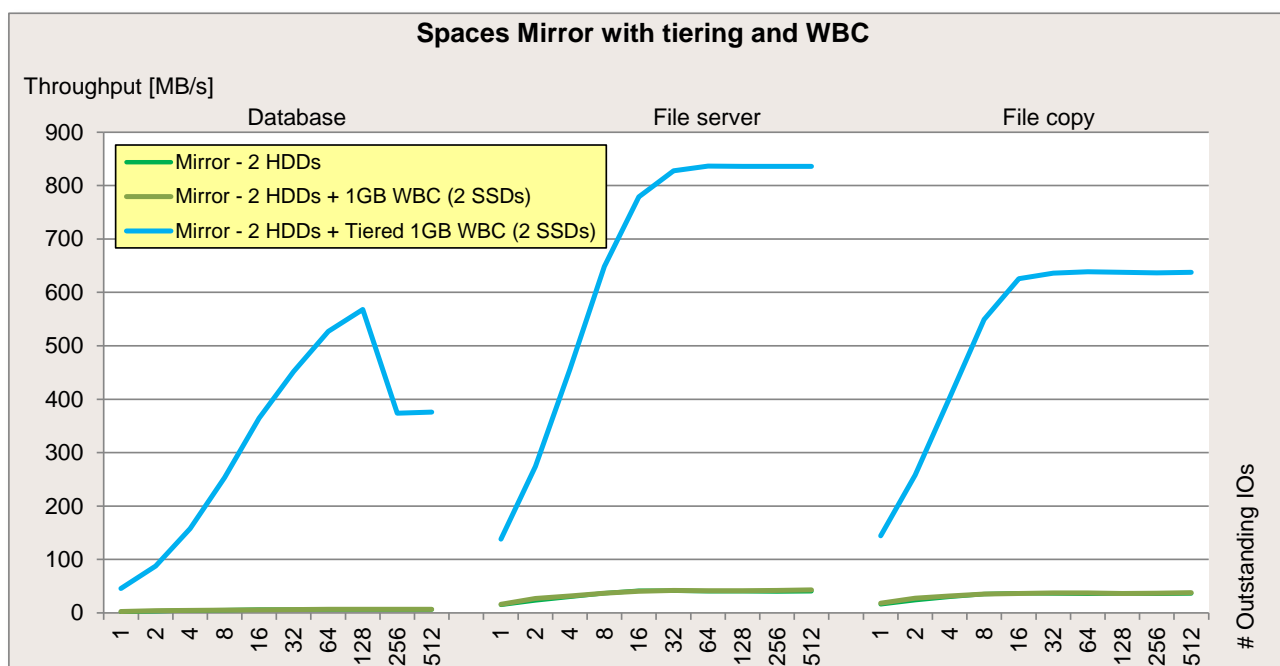
The second graphic shows a clear gain in performance with random accesses through tiering. It should be remembered that this only applies to the data located on the SSD tier. The accesses to the data on the HDDs perform up to 30% slower with two SSDs for the SSD tier and the resulting number of two columns.

## Spaces Mirror

The first diagram shows the measurement results of the standard load profiles “Streaming” (sequential access, 100% read, 64 kB block size) and “Restore” (sequential access, 100% write, 64 kB block size).



The next diagram shows the course of the standard load profiles “Database” (random access, 67% read, 8 kB block size), “File server” (random access, 67% read, 64 kB block size) and “File copy” (random access, 50% read, 64 kB block size).



The first diagram shows that the sequential read and write performance of the SSD tier is significantly better than that of two HDDs without the tiering function.

As already described in [“Storage Spaces with SSD Write-Back Cache”](#), the measurement results for the purely WBC measurement in the “Restore” load profile do not necessarily reflect everyday practice. If the application uses a block size of 256 kB or greater, it can be assumed that the same performance as without a configured WBC will be achieved.

In the case of random accesses, the SSD tier offers significantly better performance.

## **Conclusion: Storage Spaces with Tiering**

The use of tiering can significantly improve the performance of the “Simple” and “Mirror” Spaces layouts. However this only fully applies to random accesses. In the case of the “Simple” and “Mirror” Spaces layouts, a smaller number of SSDs in comparison with the number of HDDs will lead to a smaller number of columns (number of hard disks on which the individual stripes are distributed), causing the sequential read and write performance to decline in line with the number of columns. Tiering is only recommended for sequential accesses if the number of SSDs in the SSD tier at least matches the number of HDDs in the HDD tier. However, such a high volume of SSDs is not cost effective in all cases. A possible configuration would be a large number of hard disks (e.g. 40) in combination with eight SSDs in one enclosure where several Spaces (with up to eight HDDs) share a part of the SSD capacity with eight SSDs.

In addition to this restriction, it should be noted that the data stock should include a specific hot spot, in other words a certain volume of data that is accessed particularly frequently and a difference between the HDD tier and the SSD tier in terms of performance. After all, the greater the performance difference, the more effective the use of tiering will be.

As already described in the restrictions in the “[Storage Tiering](#)” subsection of the “[Functionality](#)” section, seldom used data located on the HDD tier will be moved to the SSDs again as a result of sudden more frequent use when the “Storage Tiers Optimization” task is run (usually at 1 a.m.). This should be considered because the anticipated performance improvement only occurs later. A hot spot that largely changes on a daily basis cannot be accelerated efficiently through auto tiering unless the “Storage Tiers Optimization” task is run several times each day, which, as pointed out in the description of storage tiering, can lead to a loss of performance during operation. In addition, files can be assigned to a specific tier.

If it is decided that the tiering function is to be used, so that there are SSDs in the storage pool, it is best to configure a WBC of 1-10 GB per Space. After all, even though it may seem that WBC has no performance benefits to offer because all data must first be written to the SSD tier, so that performance is the same as that of the WBC, the WBC can actually improve write performance if there is no longer any free capacity on the SSD tier. Also, in the event that already existing data is modified on the HDD tier, the performance of the HDD tier will benefit from the existence of a WBC.

Because Storage Spaces clearly benefits for the high SSD speed with the tiering functionality, it can not be compared fairly with hardware RAID at this point. Hardware RAID controller “RAID Ctrl SAS 6G 5/6 1GB (D3116C)” offers optional so-called “MegaRAID advanced software options” that can also bring performance gains by using SSDs. These are described below.

## **RAID Ctrl SAS 6G 5/6 1GB (D3116C) CacheCade 2.0**

Like Tiering in Storage Spaces, CacheCade in the hardware RAID controller “RAID Ctrl SAS 6G 5/6 1GB (D3116C)” ensures that frequently used data is stored on the SSDs, thus speeding up access to this data. However, the principle behind CacheCade is different from tiering in Storage Spaces. In the case of CacheCade, data is always written first on the HDDs and, if used more frequently, moved to the SSDs. Hot and cold data are assessed at any time. The maximum size of a CacheCade cache is 512 GB.

Another difference is that CacheCade can also only be configured as a read-only cache. In addition, unlike tiering in Storage Spaces, CacheCade does not allow the user to choose whether certain data is to be stored on the SSDs or HDDs.

Because CacheCade works in a fundamentally different way to storage tiering, a direct comparison can not be made without further ado. Ideally, CacheCade enables the “RAID Ctrl SAS 6G 5/6 1GB (D3116C)” hardware RAID controller to equal the performance of storage Spaces with tiering.

The “[RAID Controller Performance](#)” White Paper explains the performance enhancements that can be expected with CacheCade.

## CPU utilization with Storage Spaces, Software RAID and Hardware RAID

Because the intelligence and computing capacity must be handled by the CPU in the case of Storage Spaces and software RAID, while the RAID controller provides this function in the case of hardware RAID, higher CPU utilization is to be expected.

Measurements in this context based on the hardware described in the "[Measurement environment](#)" section have shown that the entire CPU utilization is 2% to 10% with Storage Spaces and software RAID, while only up to 3% CPU utilization was measured with hardware RAID.

This aspect should be taken into account if the utilization of the CPU by other applications is already high (> 70%). If CPU capacity is not sufficient, performance in Storage Spaces and software RAID can suffer.

Modern CPUs already offer a large number of processor cores for the parallel execution of computing operations, so that even a system with just one CPU is perfectly capable of running Storage Spaces.

## Summary and conclusion

### Operations and ergonomics

Storage Spaces from Microsoft is an extension of the standard Software RAID (which is administered in the disk management console in Windows). Storage Spaces can be configured and managed using the Server Manager graphic interface specially developed for this purpose or with the PowerShell. Thanks to the graphic interface and the simple handling procedures, it is easier for users who previously had little contact with RAID configurations to get started in Storage Spaces. However, it is necessary to take a closer look at Storage Spaces in order to be able to configure and use all functionalities on an individual basis and to be aware of the dependencies. After all, anyone who considers the topic of Storage Spaces more intensely will soon realize that the graphic interface does not display all information and does not allow some functions to be configured. For this reason you are strongly advised to use the Power Shell for productive operations.

In addition, standard configuration of Storage Spaces takes some of the decision-making away from the administrator, irrespective of whether or not they are managed by means of the graphic interface or the PowerShell, so that the result is not always immediately understandable. For example, it is not obvious which hard disks in the storage pool are actually to be used for a particular Space or where the WBC is actually created.

### Functionality

The outstanding functions of Storage Spaces include the following:

- It is possible to add additional storage capacity in the form of hard disks to a Storage Pool, Space, HDD tier or SSD tier at any time.
- The manual assignment of files to a HDD tier or SSD tier, in addition to the internal automatic distribution algorithm of Storage Spaces, enables tiering to be optimized.
- Storage Spaces can be used on any hardware that supplies hard disks to the operating system as JBOD device, irrespective of the manufacturer.

Storage Spaces also imposes the following limits / restrictions:

- The size of a Write-Back cache cannot be changed in an already existing Space.
- Tiering cannot be used at a later point for an already existing Space.
- Tiering is not supported for the "Parity" Spaces layout.
- Booting from Spaces is not supported, so that the boot hard disks need to be run on a different hard disk controller.
- It is not always obvious, how many columns are created for a Space, which can have unpleasant consequences later on. For example, the number of hard disks added must correspond to the number or a multiple of the columns in the relevant Space.)
- When using the WBC and tiering functionality, the location of data is not always clear. When a Space or tier is subsequently extended by additional hard disks, it is no longer possible to determine the location of the data.
- When using tiering only files can be pinned to a tier, but not folders.
- Even if Storage Spaces allows a WBC with just two SSDs when a "Dual Parity" and "Three-way-mirror" is created, three SSDs should be used as recommended by Microsoft, as the WBC otherwise will not be able to cope with the same number of hard disk failures as the HDD tier.

## Performance

In order to be able to assess performance and decide whether the use of Storage Spaces makes sense, a distinction should be made between two fundamental scenarios.

### *Storage Spaces without the use of SSDs*

Without the use of SSDs, Storage Spaces, as described in subsection [“Storage Spaces vs. Software-RAID vs. Hardware-RAID”](#) only can be recommended with the Spaces layouts “Simple” and “Mirror” even if the performance in some cases does not quite match hardware RAID 0 and 1 with the RAID controller “RAID Ctrl SAS 6G 5/6 1GB (D3116C)”. Storage Spaces only makes sense for pure read accesses for the Spaces layouts “Single Parity” and “Dual Parity” and offers approximately the performance of Hardware RAID 5 and 6 in these load profiles. Hardware RAID offers ten times higher performance for both sequential and random write accesses and should therefore be favored for these two load profiles.

Although Software RAID is the long-standing predecessor of Storage Spaces, it performs equally well or even slightly better than Storage Spaces in most cases. One exception to this is the mixed read and write access for the Spaces layout “Simple” and Software RAID “Stripe”, where Storage Spaces has marked advantages in part.

### *Storage Spaces with the use of SSDs*

The Write-Back Cache and tiering functionalities presume the use of SSDs, whereby the WBC can buffer write phases with a high load and thus increase the write speed for a certain amount of time and tiering accelerates the access to frequently used data. Prerequisite for a marked increase in performance due to tiering is the existence of a hot spot within the data area, which does not change to a large extent on a daily basis.

If the decision in favor of using SSDs is made, the WBC and tiering functionality should be used jointly. In the case of “Parity” Spaces layouts, SSDs can only be used for the use of the WBC, because the tiering functionality is not supported there.

The WBC should be between 1 GB and 10 GB large. The capacity of the SSD tier should as a rule of thumb account for between 1% and 8% of the capacity of the HDD tier.

The use of WBC is only recommended to a restricted extent for predominantly sequential write accesses. More explanations are available in section [“Storage Spaces with SSD Write-Back Cache”](#).

The “Parity” Spaces layouts only offer an acceptable performance with an SSD Write-Back Cache. Even with the minimum number of two SSDs for the WBC, the performance of random accesses can be increased by a factor of three and Storage Spaces can thus almost equal the performance of hardware RAID. Performance is still only insufficient for sequential write accesses, even though a clear improvement is achieved through the use of the WBC.

Tiering is especially recommended for random accesses because significant increases in performance can be achieved there. Tiering is only recommended for sequential accesses if the number of SSDs in the SSD tier matches the number of HDDs in the HDD tier. However, such a high volume of SSDs is not cost effective in all cases. A possible configuration would be a large number of hard disks (e.g. 40) in combination with eight SSDs in one enclosure where several Spaces (with up to eight HDDs) share a part of the SSD capacity with eight SSDs.

If the hot spot can be covered by the SSD tier to optimum effect, so that the majority of accesses can be serviced by the faster SSD tier and only a small part of the accesses are handled by the HDD tier, it is best to use hard disks with the best possible price/GB ratio for the HDD tier. Examples are BC-SATA or BC-SAS hard disks.

It would only be possible to compare storage Spaces with tiering and RAID hardware at this point using the “CacheCade” functionality of the “RAID Ctrl SAS 6G 5/6 1GB (D3116C)” controller. Because the functionality of these two technologies is fundamentally different, it is not possible to compare them directly. Ideally, CacheCade enables the “RAID Ctrl SAS 6G 5/6 1GB (D3116C)” hardware RAID controller to equal the performance of storage Spaces with tiering.

## Frequently Asked Questions

### Is thin provisioning a feature of a storage pool or of a Space?

Thin provisioning refers to the Space. A storage pool can contain Spaces both with thin provisioning and with fixed provisioning.

### Is it possible to change the provisioning setting after creating a Space?

This is unfortunately not possible. For this purpose, it would be necessary for a Space to be deleted and newly created.

### Is it possible to add storage Space in the form of further hard disks to an existing storage pool?

This is possible at any time, but the number of added hard disks must correspond to the number or a multiplicity of the columns in the respective Space.

### Is it possible to change the Space layout of an existing Space?

This is unfortunately not possible. For this purpose, it would be necessary for the Space to be deleted and newly created.

### Is it possible to combine different hard disks (manufacturer, performance, capacity) in a pool?

In principle, it is possible but not recommended, because when using hard disks with different speeds in a Space the expected performance can only be estimated with great difficulty. If hard disks and SSDs are combined in a pool for the use of SSD Write-Back Cache or tiering, you should ensure that the SSDs are only used for the WBC or the SSD tier and not for the actual Space. A detailed description is available in the subitem "[SSD Write-Back Cache \(WBC\)](#)" of subsection "[Explanations](#)" of the section entitled "[Functionality](#)".

### Which throughput can the controllers destined for Storage Spaces "PSAS CP200i" or "LSI SAS 9200-8e PCIe" provide?

The named controllers fully exploit the bandwidth of PCIe Gen2 and provide a maximum throughput of 2800 MB/s for sequential access and a maximum of 3200 MB/s for random access.


## Literature


### PRIMERGY Systems

<http://primergy.com/>

### PRIMERGY Performance White Papers

This White Paper:

 <http://docs.ts.fujitsu.com/dl.aspx?id=65bfeca0-c51a-44bb-acc3-5e92a0e38b98>

 <http://docs.ts.fujitsu.com/dl.aspx?id=820a11e1-73ae-49ce-bf53-86be33adbc72>

Basics of Disk I/O Performance

<http://docs.ts.fujitsu.com/dl.aspx?id=65781a00-556f-4a98-90a7-7022feacc602>

Solid State Drives – FAQ

<http://docs.ts.fujitsu.com/dl.aspx?id=78858d6c-4c0f-479a-8ceb-705fe1938f4e>

Windows Storage Spaces Performance

<http://docs.ts.fujitsu.com/dl.aspx?id=83fe3711-d361-4ca8-aa00-2ec69c8d0e4f>

RAID Controller Performance

<http://docs.ts.fujitsu.com/dl.aspx?id=e2489893-cab7-44f6-bff2-7aeea97c5aef>

Single Disk Performance

<http://docs.ts.fujitsu.com/dl.aspx?id=0e30cb69-44db-4cd5-92a7-d38bacec6a99>

### PRIMERGY Performance

<http://www.fujitsu.com/fts/products/computing/servers/primergy/benchmarks/>

### Microsoft Storage Spaces

Website:

<http://technet.microsoft.com/en-us/library/dn387076.aspx>

Windows Server Catalog

<http://www.windowsservercatalog.com/default.aspx>

### Information about Iometer

<http://www.iometer.org>

## Contact

### FUJITSU

Website: <http://www.fujitsu.com/>

### PRIMERGY Product Marketing

<mailto:Primergy-PM@ts.fujitsu.com>

### PRIMERGY Performance and Benchmarks

<mailto:primergy.benchmark@ts.fujitsu.com>